



A MULTI-AGENT FRAMEWORK FOR LOAD CONSOLIDATION IN LOGISTICS

Adil Baykasoglu¹, Vahit Kaplanoglu², Rizvan Erol³, Cenk Sahin⁴

¹ Dept of Industrial Engineering, Dokuz Eylul University, 35160 Buca/Izmir, Turkey

² Dept of Industrial Engineering, University of Gaziantep, 27310 Şehitkamil/Gaziantep, Turkey

^{3,4} Dept of Industrial Engineering, Cukurova University, 01330 Balcalı/Adana, Turkey

E-mails: ¹baykasoglu@gmail.com (corresponding author); ²kaplanoglu@gantep.edu.tr;
³rerol@cu.edu.tr; ⁴cenksahin@cu.edu.tr

Received 23 July 2010; accepted 10 August 2011

Abstract. Logistics companies mainly provide land transportation services facing with difficulties in making effective operational decisions. This is especially the case of making load/capacity/route planning and *load consolidation* where customer orders are generally unpredictable and subject to sudden changes. Classical modelling and decision support systems are mostly insufficient for providing satisfactory solutions in a reasonable time solving such dynamic problems. Agent-based approaches, especially multi-agent paradigms that can be considered as relatively new members of system science and software engineering, are providing effective mechanisms for modelling dynamic systems generally operating under unpredictable environments and having a high degree of complex interactions. It seems that multi-agent paradigms have big potential for handling complex problems in land transportation logistics. Based on this motivation, the paper proposes a multi-agent based framework for load consolidation problems of third-party logistics companies.

Keywords: third part logistics, load consolidation, agent based load consolidation.

1. Introduction

Worldwide dependence of goods flow is rapidly growing. As a result of such condition, the logistics and planning of load transportation has become vital for both economical and ecological reasons (Baykasoglu, Kaplanoglu, 2011; Daunoras *et al.* 2008; Fischer *et al.* 1996).

During transportation operations, logistics companies select an optimal transportation alternative concerning only the criteria related to the economic effectiveness of the transport task (Simongáti 2010). Although the abundance of transportation types reduces transportation costs, the business complexity of third party logistics (3PL) proportionally increases.

Freight or *load consolidation* is a transportation option that combines different items produced and used at different locations and different times with single vehicle loads in order to minimize transportation costs and maximize vehicle utilization. When more than one item is put to the same container, it means they are consolidated into a single vehicle unit (Hall 1987). The more items are put inside the container the lower is transportation cost per order and per unit. Although the practice of load consolidation is widely used in road, rail, sea and

air transportation (Tyan *et al.* 2003), the interdependence of organizations and volatile business environments make the problem domain very hard to model and solve.

In general, load consolidation problems are handled in a similar manner with the economic order quantity models (EOQ) of operations research (OR). Economic shipment weight (ESW) models are used in order to consolidate loads into a point (at the capacity or time level) before dispatching under an assumption that all loads have the same route. However, practically, orders having different routes might be consolidated within the same container. Therefore, in a practical manner, the problem is much more complex than the ones handled applying ESW models. Load consolidation decisions inherently comprise the problem of order-scheduling where efficient algorithms for solving static scheduling problems exist; however, classical computer science, OR and classical centralized artificial intelligence (AI) have so far failed to provide adequate methodologies and algorithms for coping with open dynamic scheduling problems (Fischer *et al.* 1996). Unfortunately, virtually, any logistics problem of practical interest is not static. Thus, calculating the optimum of a complex logistics problem

is a difficult task. It is not possible for similar problems to calculate the best solution in a straightforward manner. Within complex inter-organizational relations, the performance of a logistics company is dependent on the effective coordination and collaboration of system resources and partners (Wong, Fang, 2008).

Earlier studies on this subject reveal that agent-based systems seem very suitable for the transportation domain. However, this subject needs to be verified employing a more deployed system (Dastani *et al.* 2004; Davidsson *et al.* 2005; Fox *et al.* 2000; Ying, Dayong 2005).

There are numerous studies in literature where an intelligent agent-based solution and co-operative systems are proposed for some distributed, adaptive and dynamic problem domains (Cil, Mala 2010; Jarašūniene 2007; Szűcs 2009). Nevertheless, the application of multi-agent technology in supply chain management and logistics, especially the coordination of transportation orders, has become a strongly emerging research area (Lu, Wang 2008).

In this paper, load consolidation problems are handled within a multi-agent framework so as to adapt the decisions made in this domain to the complex and changing business environment of logistics. The agent term in this paper denotes software elements representing real transportation resources or entities. The present paper is organized as follows. We begin with an overview of load consolidation problems encountered within a 3PL company. Then, the indication of a multi-agent system environment and fundamental agent architecture are presented. Next, the multi-agent framework for load consolidation decisions is described. We also provide an auction-based method for load assignment to transportation resources in order to consolidate transportation orders adaptively/effectively using the Prometheus design methodology. Finally, conclusions are drawn at the end of the paper.

2. Overview of Load Consolidation Problems

Managers must make a number of decisions on planning load consolidation:

- time of order release;
- triggering events to dispatch consolidated loads;
- distinction between the loads to be consolidated and shipped alone;
- the place of consolidation (road, factory, vehicle, warehouse or terminal)
- a method used for consolidation (Higginson, Bookbinder 1994).

Fig. 1 summarizes the decision problem when a transportation order arrives to a shipper.

There are mainly three commonly used load consolidation policies in literature: time, quantity and time-quantity. Considering quantity based load consolidation, if the order arrival rate is known, the ESW model is used for calculating the weight that should be accumulated in order to make shipment economical. ESW plays the same role for economic order quantity (EOQ) in inven-

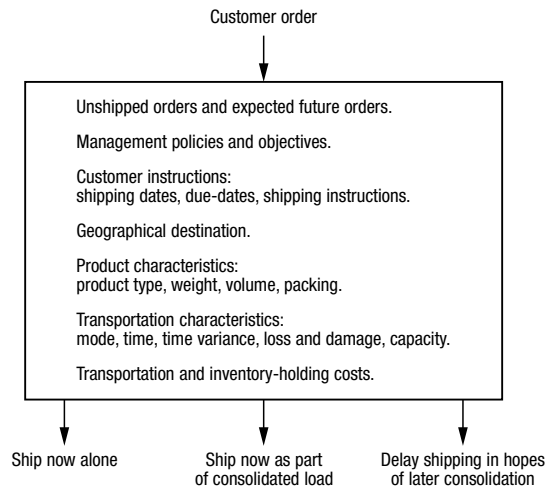


Fig. 1. A decision problem for a newly arrived customer order (Higginson, Bookbinder 1994)

tory models (Higginson, Bookbinder 1994). Equation represents the ESW model:

$$ESW = \sqrt{\frac{2 \cdot \hat{a} \cdot F_L \cdot E[W]}{r_w}}$$

where: \hat{a} is the order arrival rate; F_L is the sum of all fixed costs associated with vehicle load; $E[W]$ is the expected weight per customer order; r_w is the variable cost of carrying inventory per unit weight per time period.

The analytic models given in such studies try to find the best shipment consolidation policy under some conditions. However, real life consolidation/dispatching operations do not fit exactly to the models given in similar studies due to the dynamic nature of a real business environment. For example, a dispatch officer might change the content of the consolidated batch during the process of consolidation because of some changes in order attributes or resource conditions. A dispatch officer might shift one load from one batch to another for the sake of route optimization and cost reduction after a change in order attribute. Thus, new technologies are required to keep up with the complexity and dynamics of the domain (Fischer *et al.* 1996).

The domain of this problem has common properties with vehicle routing problems (VRC). A typical vehicle routing problem can be described as the problem of designing least cost routes from one depot to a set of geographically scattered points (cities, stores, warehouses, schools, customers etc). The routes must be designed in such a way that each point is visited only once by exactly one vehicle, all routes start and end at the depot and the total demands of all points on one particular route must not exceed the capacity of the vehicle. In practice, the load consolidation problem also considers the vehicle routing problem of pickup and delivery (VRPPD) and vehicle routing problem of time windows (VRPTW). Still, pick-up and delivery points and time windows of *all orders* are unknown for a static point of time. Load consolidation decisions are simultaneously conducted by considering all available order attributes and system parameters.

3. Multi-Agent Systems

As is expected from a fairly young area of research, there is not yet a universal consensus on the definition of an agent (Padgham, Winikoff 2004). However, the definition provided by Wooldridge and Jennings (1995) is increasingly adopted in the above discussed field; 'An agent is a computer system *situated* in some *environment* and capable of an *autonomous action* in this environment in order to meet its design objectives' (Wooldridge, Jennings 1995). When an agent is instantiated, it will wait until given a goal to achieve or experiences an event it must respond to (Hahn *et al.* 2009). Agents are *social*, because they cooperate with humans or other agent types in order to achieve its tasks. Agents are *reactive*, because they perceive their environment and respond in a timely fashion to changes that occur in the environment.

4. Multi-Agent System for Load Consolidation Decisions

The reason for applying a multi-agent approach to the load consolidation problem is the complexity of the problem domain of 3PL. The model of load consolidation discussed in literature is inadequate for solving the problems of open-dynamic load consolidation. ESW models assume that, consolidated orders are taken from a single source and shipped to a single destination. However, in real business, this is not the fact. The orders from different sources might be consolidated and shipped to different destinations within the same container under their respective time constraints. In this study, system resources such as trucks maintain their own load consolidation initiatives. Thus, the decision of a very complex problem of load consolidation is made by some system resources and some system entities (orders) collaboratively. This construct enables the system to response some unexpected events without global re-planning and re-consolidation.

The domain structure of load consolidation problems in 3PL is also eligible for decentralized decision making due to the natural distributed structure of 3PL operations. Resources and entities are naturally distributed on a transportation network. In addition, system resources and entities within this domain react proactively within such business environment in order to reach their respective goals. Agents are often situated in dynamic environments that change rapidly. The business environment of the 3PL domain is also exposed to sudden changes. Therefore, the system resources represented as an agent within the domain of the faced problem fits to the agent definition. The agent types within the 3PL domain have the property of flexibility and sociality due to the fact that resource elements can change their plans in order to achieve their goals (flexibility) and interact with other system resources (trucks, drivers etc.) in a social environment. What is different in our proposed approach is that the 3PL companies themselves do not have to centrally give load consolidation decisions. Thus, one very complex central load consolidation decision is replaced with several (depends on the number of trucks, drivers and orders) smaller consolidation auctions allowing for quick reactions without global re-consolidation.

5. Design of a Multi-Agent System

In this study, a multi-agent based load consolidation system is designed by utilizing the methodology of Prometheus, which is a general purpose methodology for the development of software agent systems not tied to any specific model of the agency of the software platform (Padgham, Winikoff 2004). The Prometheus design methodology comprises three distinct but interconnected stages. Fig. 2 represents the stages of modelling the Prometheus agent methodology:

- **System specification stage:** at this stage, the system is specified by focusing on the goals and basic functionalities of the system.

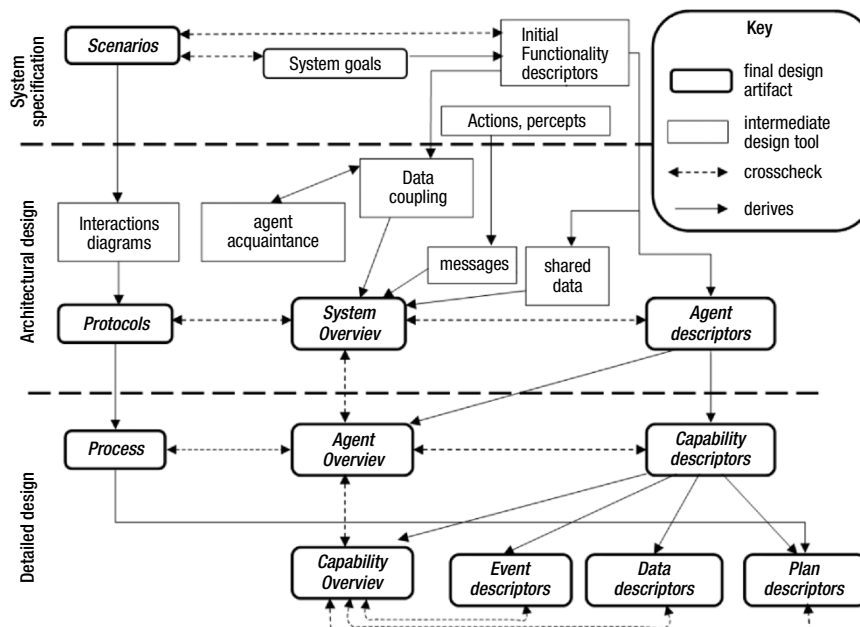


Fig. 2. The stages of the Prometheus methodology (Padgham, Winikoff 2004)

- **Architectural design stage:** at this stage, the agent types are determined according to the system specification stage. It is similar to building class diagrams from the requirements for documents in the unified modelling language (UML).
- **Detailed design stage:** this stage includes the plan definition of the agent types.

The further section of this paper proposes a multi-agent load consolidation approach that is formally designed using the Prometheus methodology.

5.1. System Specification

The Prometheus methodology used in this study focuses particularly on the specification of goals and scenario descriptions. In addition, it requires the description of *functionalities* related to the identified goals. At this stage of modelling, environmental interfaces having suited agent types are determined. The interfaces are defined in terms of *percepts* arriving from the environment. Also, actions having an impact on the environment are taken at this stage of modelling.

5.1.1. Goal Specification

System goals are defined as:

- consolidating transportation orders while satisfying system constraints (e.g. pick-up and delivery times, volume and weight capacity of containers etc.) in real time;
- providing support for dispatchers during their decision process

As regards the Prometheus methodology, the system goals are refined addressing the question ‘how’ to each goal of the system. The answer to this question is the sub-goal of each goal. Thereby, the refined goal list is obtained:

- Consolidating transportation orders while satisfying system constraints (e.g. pick-up and delivery times, volume and weight capacity of containers etc.) in real time:

Sub-goals:

- get information on the orders from customers in real time;
- monitor each truck (geographic position, availability condition, capacity level etc.)
- Select a proper truck for assigning the order on hand (e.g. hazard category of dangerous goods, some special equipment might be necessary to handle the order);
- check truck routes in order to change/exchange orders between trucks;
- get changes in truck attributes from the truck monitoring system;
- get changes in order attributes from the user interfaces;
- change/exchange transportation orders between trucks when necessary;
- provide order delivery schedule to the user interfaces.
- Provide support for dispatchers during their decision process:

Sub-goals:

- get load consolidation decisions from the system;
- obtain data on the real time system ;
- monitor trucks during their operation in real time;
- monitor orders during their operation in real time;
- provide the user interface to make the system users to monitor system changes;
- provide the user interface to make the system users to be aware of transportation proposals (when orders have not definite pick-up and delivery time)

In the further section of the system specification stage, the system goals are presented and coalesced into four different groups:

Order Management:

- get order information from customers in real time;
- get order attribute changes from the user interface;
- monitor orders during their operation in real time;

Truck Management:

- monitor each truck (geographic position, availability condition, capacity level etc.);
- check truck routes in order to exchange orders between trucks;
- get truck attribute changes from the truck monitoring system;
- monitor trucks during their operation in time;

Load Consolidation:

- select a proper truck to assign the order on hand (e.g. hazard category of dangerous goods, some special equipment is necessary to handle the order)
- change/exchange transportation orders between trucks when necessary;
- get data on the real time system;

General System Management:

- provide an order delivery schedule to the user interface (when orders have not definite pick-up and delivery time);
- get load consolidation decisions from the system;
- provide the user interface to make the system users to monitor changes in the system;
- provide the user interfaces to make the system users to be aware of the system transportation proposal (when orders have not definite pick-up and delivery time).

5.1.2. Functionalities

Functionality is the term used for a stack of behaviour that consists of grouping related goals, percepts, actions and data relevant to behaviour (Padgham, Winikoff 2004). The actions, triggering events, information used and information produced are the elements stated within the functionality descriptor at a system specification

stage. Functionality descriptors for each cluster of the system goals are as follows:

- **Order management functionality.** The goals of the order management group include 'getting the order data from a customer via the Internet or a telephone call', 'getting changes in the order attribute from the user interface via the Internet or a telephone call' and 'monitoring orders during their operation in real time' actions. The system is triggered by a new order arrival or an attribute change in the order.
- **Truck management functionality.** The goals of the truck management group embrace 'monitoring trucks', 'checking truck routes in order to exchange orders between trucks', 'getting changes in truck attributes from the truck monitoring system' and 'monitoring trucks during their operation in real time' actions. The system is triggered by changes in truck attributes. Data on a geographical position and a container status of trucks is used. When transportation operations are completed, information on operation is generated.
- **Load consolidation management functionality.** The goals of load consolidation management are consolidation orders while "changing/exchanging transportation orders between trucks" and "getting data on the real time system" actions. The system is triggered by truck management and order management functionality actions. Data on a geographical position of trucks as well as data on the capacity utilization of trucks and changes in order attributes are used. The system is also triggered by changes in order attributes. The orders might be re-consolidated with other trucks after they are assigned to a particular truck. When referring to this functionality, operation data are generated.
- **General system management functionality.** The goals of general system management cover 'providing order delivery schedule', 'getting load consolidation decisions from the system' and 'providing the user interface' actions. The actions of order and truck management functionality as well as load consolidation management functionality trigger the actions of general system management functionality. The data produced from load consolidation management functionality are used as input in this functionality.

5.1.3. Scenario Development

Scenarios are complementary to goals showing the sequences of steps taking place within the system. The scenarios in the agent based load consolidation system shows the sequence of operations required for changes in the system. In this study, the scenarios are stated as follows:

- order arrival to the system;
- changes in truck attributes;
- changes in order attributes after the plan is prepared;

- selecting proper trucks from the available trucks within the system;
- change/exchange transportation orders between trucks;
- order move (transportation);
- delay in transportation (due to a traffic jam etc.).

5.1.4. Interface Description

Interface description is also complementary to scenario development that appears during the created scenarios:

- **Percepts:** information on a new order of arrival, information on changes in the position of the truck network, information on loading/unloading, changes in order attributes, traffic jam, truck breakdowns, driver breaks, delays in customs, search for new consolidation alternatives.
- **Actions:** change/exchange of orders after reassignment or changes in reconsolidation, delivery time, changes in route order, reassignment, reconsolidation, changes in truck routes, changes in container elements.
- **Data:** a database of a transportation management system (TMS): the database system contains information about operations stored for retrieval when necessary. The database used within this system is the one of the management system for 3PL logistics operations.

5.2. Architectural Design: Specifying Agent Types and Interactions

During system specification stage described in section 5.1, system goals, scenarios, functionality and interface descriptions are modelled. The above section covers an architectural design stage of the Prometheus methodology for the 3PL agent system. First, agent types in the 3PL system are determined. Then, the interactions between the agent types are modelled (by utilizing *interaction diagrams*). Finally, the overall system structure is modelled by identifying the boundaries of the 3PL logistics system.

The architectural design used in the Prometheus methodology defines the agents that are a part of the system and the way these agents interact with each other to meet the predefined functionalities of the system. The Prometheus methodology employs the agent types constituted by utilizing the functionalities of the system. Determining the type of an agent is mostly based on the functionalities and scenarios of the system specification stage. Each agent must be cohesive and have a description with a short name. The designed agents also should be loosely *coupled*. After the test on coupling and cohesion, the agents should be described from the view point of the scenario and system interfaces (percepts and actions).

5.2.1. Specifying Agent Types

Fig. 3 depicts a diagram of data coupling and refers to load consolidation decisions made within the 3PL domain. The diagram of data coupling consists of functionalities and all identified system data. Then, the links are connected between functionalities and system data.

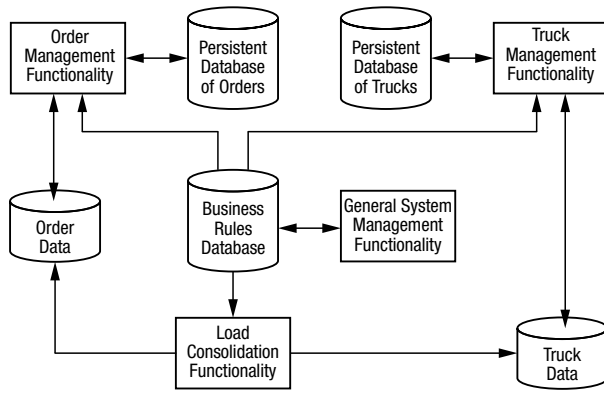


Fig. 3. The diagram of data coupling concerning decisions on load consolidation in 3PL

Three clusters of the diagram of data coupling are obtained: *order agent*, *truck agent* and *regional load consolidation agent* respectively.

Order agents are generated when a new order request enters the system. Pick-up and delivery time attributes of the order agents can be summarized in Fig. 4:

- τ_i^{ARV} denotes the arrival time of order i ;
- τ_i^{AVL} is the earliest pick-up time of the order;
- T_i^{ADV} measures the time between the arrival of order i and its earliest pick-up time;
- W_i is the corresponding duration between the earliest pick-up time and the earliest delivery time;
- T_i^{SLK} is the slack time available between the earliest possible and latest allowed delivery;
- τ_i^{DLN} is the time of the latest delivery;
- T_i^{RES} is the time when the company needs to respond to a job.

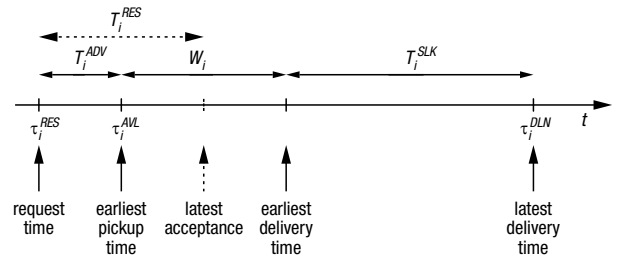


Fig. 4. Pick-up and delivery time attributes (Yang et al. 2004)

Critical time for the order agent is T_i^{RES} . The agent suits in the system for the time period of T_i^{RES} . In case it cannot find any truck agents within this period, it leaves the system without being assigned or consolidated to a proper truck agent.

Truck agents have the attributes of geographical coordination, a list of consolidated order agents and vehicle status (transporting, loading, unloading or idle).

Regional load consolidation agent: as represented in Fig. 5, when a new order enters the system, it directly requests a regional load consolidation agent to be directed to respective (or available) truck agents to be assigned or consolidated.

The mediator interface (central interface) provides a global information base in order to support regional load consolidation agents (see Fig. 5) and works like a central switch to route new arriving orders considering related regional load consolidation agents.

5.2.2. Specifying Interactions

After determining agent types, inter-agent interactions are held. In the diagrams displaying interaction, time increases as one move down the diagram. Each agent has a lifeline graphed with a vertical line with the agent name.

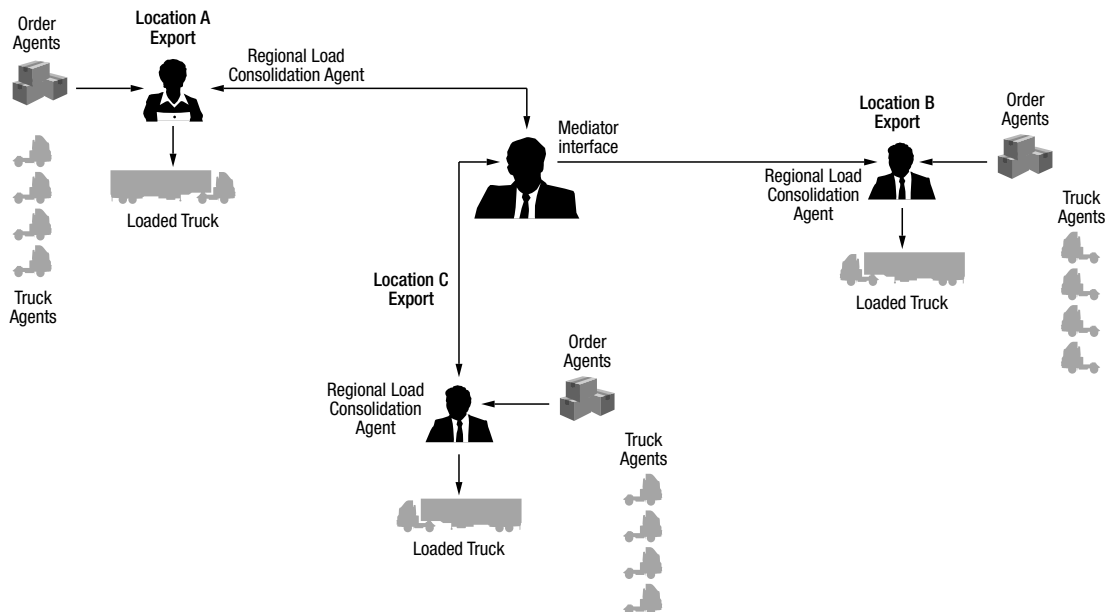


Fig. 5. Agent-based load consolidation environment

Messages are depicted as horizontal arrows between lifelines. Sample interaction diagrams selected for some system scenarios are presented in the further section of this paper.

5.2.2.1. Order Arrival (Percept) – Order Agent: The Interaction Diagram

Scenario: order arrival to the system

The diagram of such interaction consists of an arrival of a new order from any system resources as the percept (such as a telephone call or the Internet). The order agent dynamically builds itself from data provided by the customer order form. Fig. 6 depicts such interaction in the diagram. The time that passed from the point of response to the percept is for finding a suitable truck to be assigned or consolidated. During this process, the order agent builds necessary connections with a regional load consolidation agent to find suitable truck agents to be assigned or consolidated.

The order agent uses data on the percept to make itself to be assigned or consolidated. Time to be consolidated to a proper truck depends on the order attribute. If the order has an attribute of *quick response by the system* (short T_i^{RES}), then the order agent suits in the system up to the predetermined time. Within this period, if the order agent finds a suitable truck agent, it proposes a schedule for the environment (customers); in case the order agent cannot find a suitable truck agent, it leaves the system without being processed.

5.2.2.2. A Diagram of the Interaction between the Order Agent and the Truck

Scenario: selecting proper trucks from the available trucks within the system

Fig. 7 illustrates the interaction between any particular order agent and the truck agents provided to order the agent by its respective regional load consolidation agent. When order agent o enters the system and the available truck agents are determined by the regional load consolidation agent, it has to bid for truck agents to execute it. It is assumed that there is m number of truck agents within the truck agent pool (truck agents may be from the same region the order agent is). The order agent calls for proposals to the agents in the truck agent pool. The order agent has the goal of *being transported* from its origin to destination within the period of τ_i^{AVL} and τ_i^{DLN} with a predetermined freight rate.

To determine costs, truck agent a computes a bid for the order agent as $(a, \text{cost}(C_a + o) - \text{cost}(C_a))$ where C_a is the current content of truck a . $\text{Cost}(C_a + o)$ denotes additional costs for a when executing o given C_a . Order agent o receives a set of bids from truck agents. $B = [(a_1, c_1), \dots, (a_n, c_n)]$, $n \in \mathbb{N}$, where c_i specifies costs that truck a_i will produce when executing order o . Order agent o selects (a_{min}, c_{min}) . Finally, agent o sends a grant to truck agent a_{min} notifying it will be granted to be assigned with the amounts of w_o and v_o , where w_o is weight and v_o is the volume of the order agent. Truck agent a_{min} informs agent o about the assignment results. The order agent is informed when a problem occurs. Af-

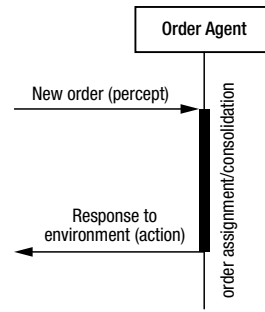


Fig. 6. A diagram of the interaction between the environment and the order agent

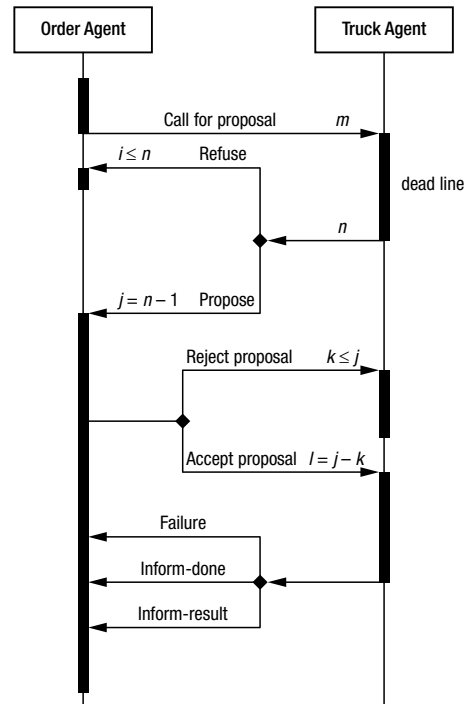


Fig. 7. A diagram of the interaction between order agents and trucks

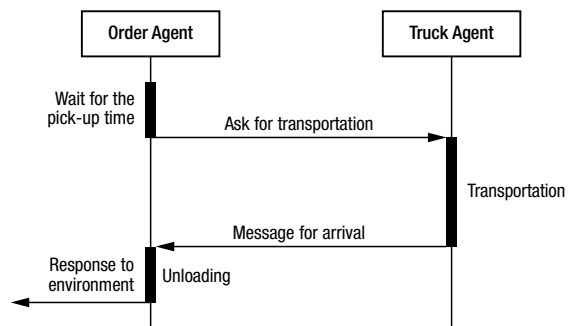


Fig. 8. Diagram of the interaction between the order agent and the truck

ter the consolidation or assignment of the order agent on a proper truck agent, the order agent starts waiting for the time of being transported. When the order agent reaches a time point where it has to be picked-up, it sends a message to the respective truck agent it had previously consolidated or assigned (see Fig. 8).

5.2.2.3. A Diagram of the Interaction between Regional Load Consolidation and the Truck Agent
Scenario: change/exchange transportation orders between trucks

This diagram represents the interaction between the regional load consolidation agent and truck agents within the system. The regional load consolidation agent interacts with truck agents by using its initiatives in order to search a better consolidation alternative. If the regional load consolidation agent finds any better load consolidation alternative, it responds to order agents and matches them with the available truck agents. As a result of this process, some order agents might be changed or exchanged between truck agents for the sake of better load consolidation (see Fig. 9).

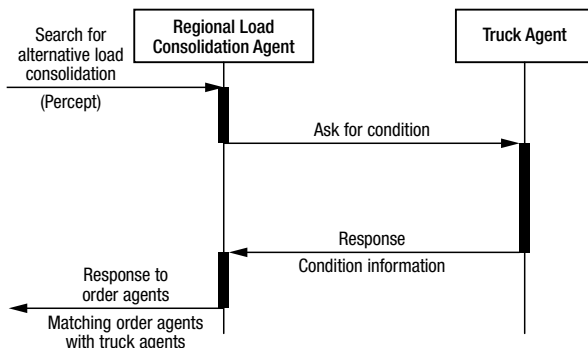


Fig. 9. Diagram of the interaction between the regional load consolidation agent and the truck agent

5.2.3. Boundaries of the Agent-Based Load Consolidation System

Agent systems are suited to an environment; thus, the environment that the agent system represents should be specified. In the real system of a 3PL company, the environment is made of the things other than the system of 3PL. Therefore, a boundary is necessary for the agent-based system. Considering the agent based design of 3PL, traffic that the trucks are involved is the same environment where truck agents are employed. Customers and customer requests with a dynamic change in their attributes are the environment the order agents are involved in. Customs and governmental regulations that expose to change are the environment of regional load consolidation agents that are directly exposed to the business rules during their operations while consolidating order agents to truck agents.

5.3. Detailed Design

The third stage of the Prometheus methodology is a detailed design of the proposed framework. At this stage, the plans of the agent type are determined. The plans provide details on how to reach the required situations and goals.

The plans are triggered by events. The percepts received from the environment, messages received from other agents and messages internal to an agent are three types of events that might be received by any particular plan. If there are several plans suitable to be triggered after receiving an event, then, it is important to speci-

fy conditions or situations under which various plans are applicable. The Prometheus methodology refers to similar circumstances as to *context condition* because it builds the main mechanism for reasoning. Context condition holds information about both the environment and agent types.

The percepts and messages between agent types are given in section 5.2.2. For illustration purposes, the following two sections represent order acceptance/rejection and load consolidation plans of truck agents.

5.3.1. Order Acceptance/Rejection Plan

The most critical point in the management of transportation logistics systems is making a decision on order acceptance/rejection regarding the current system. This is because the decision on acceptance/rejection necessitates all data of the system. Truck agents use the plan to make a decision on order acceptance/rejection. While making the acceptance/rejection decision, the truck agent must consider its operation schedule. Fig. 10 presents a sample of the truck agent schedule that includes the pickup and delivery of some specific orders ($o_1, o_2, o_3, \dots, o_n$).

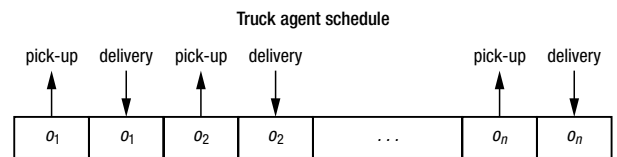


Fig. 10. Truck schedule representation

While making the decision on acceptance/rejection, truck agents should consider all the previously accepted orders considerations for which show the domain of this problem to be very complex. This is due to the fact that truck agents should consider the vehicle routing problem with time windows while making this decision. Dispatch officers also encounter the problem of order insertion considering the available truck schedules during 3PL operations. In the above proposed framework, truck agents make personal order insertion with the help of negotiation.

5.3.2. Load Consolidation Plan

After accepting respective order agents for their operation schedules, the truck agent makes the reasoning of their dispatch time. Within the process, they try to achieve a better load consolidation alternative. Truck agents have an initiative for their dispatch time. Figs 11 and 12 represent truck agents selecting their plans during operations.

Fig. 11 represents the plan selection of truck agents before a delivery operation. They can adjust their dispatch time according to their plan by considering potential load consolidation alternatives at the current position and future destination.

Fig. 12 represents the plan selection of a particular truck agent before a pickup operation. Truck agents can choose one of the plans in its plan pool. They can either wait for their current position or can go to the destination point as soon as possible.

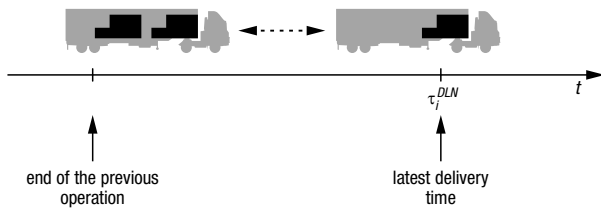


Fig. 11. Load consolidation reasoning of the truck agent before a delivery operation

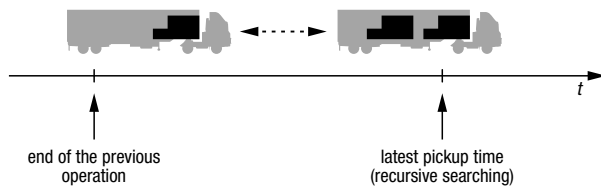


Fig. 12. Load consolidation reasoning of the truck agent before a pickup operation

The effectiveness of the operation depends on selecting the plan of the agent. Therefore, possible plans should be defined for each software agent during the implementation of the proposed framework.

6. Conclusions

3PL companies are facing difficulties in making effective decisions on load consolidation.

Classical modelling and decision support systems are not adequate to provide on-line solutions to the problem.

The present paper proposes a multi-agent framework for complex load consolidation decisions regarding 3PL companies.

The implementation of the suggested framework towards a software system for a real company is still under progress.

Acknowledgments

Prof. Dr Adil Baykasoglu is grateful to Turkish Academy of Sciences (TÜBA) for supporting his scientific studies. Vahit Kaplanoglu thanks to The Scientific and Technological Research Council of Turkey (TÜBİTAK) for supporting his Ph.D. studies.

References

- Baykasoglu, A.; Kaplanoglu, V. 2011. Evaluating the basic load consolidation strategies for a transportation company through logistics process modelling and simulation, *International Journal of Data Analysis Techniques and Strategies* 3(3): 241–260. doi:10.1504/IJDATS.2011.041333
- Cil, I.; Mala, M. 2010. A multi-agent architecture for modelling and simulation of small military unit combat in asymmetric warfare, *Expert Systems with Applications* 37(2): 1331–1343. doi:10.1016/j.eswa.2009.06.024
- Dastani, M.; Dix, J.; Fallah-Seghrouchni, A. E. 2004. *Programming Multi-Agent Systems: First International Workshop, PROMAS 2003, Melbourne, Australia, July 15, 2003, Selected Revised and Invited Papers*. 1st edition. Springer. 231 p.

- Daunoras, J.; Bagdonas, V.; Gargasas, V. 2008. City transport monitoring and routes optimal management system, *Transport* 23(2): 144–149. doi:10.3846/1648-4142.2008.23.144-149
- Davidsson, P.; Henesey, L.; Ramstedt, L.; Törnquist, J.; Wernstedt, F. 2005. An analysis of agent-based approaches to transport logistics, *Transportation Research Part C: Emerging Technologies* 13(4): 255–271. doi:10.1016/j.trc.2005.07.002
- Fischer, K.; Müller, J. P.; Pischel, M. 1996. Cooperative transportation scheduling: an application domain for DAI, *Applied Artificial Intelligence* 10(1): 1–34. doi:10.1080/088395196118669
- Fox, M. S.; Barbuceanu, M.; Teigen, R. 2000. Agent-oriented supply-chain management, *International Journal of Flexible Manufacturing Systems* 12(2–3): 165–188. doi:10.1023/A:1008195614074
- Hahn, C.; Madrigal-Mora, C.; Fischer, K. 2009. A platform-independent metamodel for multiagent systems, *Autonomous Agents and Multi-Agent Systems* 18(2): 239–266. doi:10.1007/s10458-008-9042-0
- Hall, R. W. 1987. Consolidation strategy: inventory, vehicles and terminals, *Journal of Business Logistics* 8(2): 57–73.
- Higginson, J. K.; Bookbinder, J. H. 1994. Policy recommendations for a shipment-consolidation program, *Journal of Business Logistics* 15(1): 87–112.
- Jarašūnienė, A. 2007. Research into intelligent transport systems (ITS) technologies and efficiency, *Transport* 22(2): 61–67.
- Lu, L.; Wang, G. 2008. A study on multi-agent supply chain framework based on network economy, *Computers and Industrial Engineering* 54(2): 288–300. doi:10.1016/j.cie.2007.07.010
- Padgham, L.; Winikoff, M. 2004. *Developing Intelligent Agent Systems: A Practical Guide*. 1st edition. Wiley. 240 p.
- Simongáti, G. 2010. Multi-criteria decision making support tool for freight integrators: selecting the most sustainable alternative, *Transport* 25(1): 89–97. doi:10.3846/transport.2010.12
- Szűcs, G. 2009. Developing co-operative transport system and route planning, *Transport* 24(1): 21–25. doi:10.3846/1648-4142.2009.24.21-25
- Tyan, J. C.; Wang, F.-K.; Du, T. C. 2003. An evaluation of freight consolidation policies in global third party logistics, *Omega* 31(1): 55–62. doi:10.1016/S0305-0483(02)00094-4
- Wong, T. N.; Fang, F. 2008. A multi-agent protocol for multilateral negotiations in supply chain management, *International Journal of Production Research* 48(1): 271–299. doi:10.1080/00207540802425393
- Wooldridge, M.; Jennings, N. R. 1995. Intelligent agents: theory and practice, *The Knowledge Engineering Review* 10(2): 115–152. doi:10.1017/S0269888900008122
- Yang, J.; Jaillet, P.; Mahmassani, H. 2004. Real-time multivehicle truckload pickup and delivery problems, *Transportation Science* 38(2): 135–148. doi:10.1287/trsc.1030.0068
- Ying, W.; Dayong, S. 2005. Multi-agent framework for third party logistics in e-commerce, *Expert Systems with Applications* 29(2): 431–436. doi:10.1016/j.eswa.2005.04.039