Dynamic Decision Making on Embedded Platforms in Transport Logistics - A case study

Reiner Jedermann[1], Luis Javier Antúnez Congil[1], Martin Lorenz[2], Jan D. Gehrke[2], Walter Lang[1], and Otthein Herzog[2]

[1] Institute for Microsensors, -Actuators and -Systems IMSAS {rjedermann|ljacongil|wlang}@imsas.uni-bremen.de
[2] Center for Computing Technologies—TZI {mlo|jgehrke|herzog}@tzi.de

Summary. Autonomous logistic processes aim at coping with logistic dynamics and complexity by local decision making to gain flexibility and robustness. This paper discusses resource-bounded logistics decision making using software agents and task decomposition. Simulations show the feasibility of dynamic vehicle routing and quality monitoring on embedded systems for the transport of perishable goods.

1 Introduction

Within the last years logistics has become a key success factor in globally distributed production because of its cross-sectional function. But enhanced product life cycles, rapid changes in company structures and information flows change the requirements for logistic processes. On the one hand the rising complexity of inter-organizational structures and also a relative shortage of logistic infrastructure lead to increasing utilization of the existing logistic processes. On the other hand a specialization of the ways of transportation and the carriers, which are connected to the transported goods can be observed. These factors combined with changing customer market conditions have considerable effects on planning and controlling logistic processes in such a dynamic environment. A possible approach to face these challenges is the dezentralization of logistic control and coordination by entities with the ability for autonomous decision making—in other words: autonomous logistic processes.

Especially the management of supply chains for agricultural products has to cope with multiple dynamic factors. Increasing and varying consumer expectations meet hardly predictable harvest quality and amount, which in turn puts high demands on supply chain management in the face of uncertainty [1]. Large inventories within the chain are generally to be avoided.

The quality of perishable products like fruits, meat, fish, flowers and pharmaceuticals is a dynamic factor by itself, depending on harvest / production conditions, time and stress factors during transport and storage.

In the future "quality oriented tracking and tracing system[s]" [1] will offer new approaches to cool chain management. Stock rotation could be based on current quality instead of fixed 'sell by' dates. Warehouse planning will be organized by 'First *expires* first out' (FEFO) instead of 'First in first out' [2]. Cool chain management will not only reduce unnecessary buffers in inventory but also in shelf life. The dynamic shelf life expresses the remaining time span until the product quality falls below an acceptance limit. Individual items with high reserves in shelf life will be sent exactly *where* they are needed, e.g., international deliveries or retail shops with lower turn over.

The main methodology used throughout our work is based on autonomous logistic processes modelled by software agents. Timm [3] defines four levels of autonomy and investigates the possibilities of their respective realization in software agents. Whereas "strong regulation" is associated with classical software engineering the autonomy on levels one and two, "operational autonomy" and "tactical autonomy", can be realized with known decision making methods especially the *Belief – Desire – Intention* (BDI) approach [4]. A concept for realizing "strategic autonomy" within a software agent is sketched in [3].

In the context of resource-bounded computation another aspect has to be added to the question of autonomy. An agent's autonomy is limited by the computational resources that are available for its deliberation. In other words, while *operational autonomy* can be realized by a relatively simple model-based reflex agent [5, chap. 2] *tactical autonomy* already needs planning and deliberation skills for which current resource-bounded environments are not suitable.

Therefore this paper will show the tradeoff between computational needs of autonomous logistic processes on the one hand and capabilities of current embedded systems on the other hand. Furthermore we show the current state of a prototype implementation using off-the-shelf embedded hardware to demonstrate the opportunities of intelligent local decision making for transportation of perishable goods.

2 Autonomous Decision Making in Transport Logistics

Experiences regarding perishable goods show that loss of quality depends on its location inside the container and on the environment in general [6], without forgetting unexpected problems. Therefore a prediction of actual quality losses in this dynamic environment is a hard problem. In order to enable real-time reaction quality tracing must happen locally and individually in the container.

The intelligent container we have presented in [7] provides a tracing system that consists of four components: wireless sensor network (WSN), RFID system, CPU and communication system. The WSN provides information about the environment, e.g., temperature and humidity. The RFID system is responsible for the identification of goods. The next step is the extension of this tracing system towards an autonomous decision making system, thus goods will be able to change routing decisions of the carrier if necessary for their goal achievement.

This monitoring and autonomous decision making system (MADMS) should support the following requirements. The MADMS must trace the goods in an autonomous way. It has to react to changes in the environment and the goods' shelf life. The MADMS should be able to communicate with a route planner and must be able, considering these aspects, to make autonomous decisions to adapt the route to the current shelf life state.

The MADMS can be modelled as a multiagent system, using the deliberative agent technology, i.e., the BDI architecture. The BDI architecture is based on practical reasoning and consists of two processes: deliberation and means-ends reasoning. A BDI agent has information about its environment, its goals and ways to achieve them [8].

We consider one BDI agent that is responsible for the goods in a shipping container. The agent's desire is to deliver all goods without losses. The agent's beliefs are: the goods' respective shelf-life, destination and delivery time as well as information about the local environment provided by the WSN and other information systems, e.g., the route planner. The intentions determine the agent‘s planned actions to achieve its current goals.

A BDI agent is able to make autonomous decisions, for example if a truck cannot connect with the route planner system and it does not have information about the next destination. An embedded system has bounded resources, e.g., memory or CPU, compared to desktop PCs. The challenge is the further development of a MADMS for embedded systems using the BDI architecture, where the agents must deliberate with only scarce resources.

3 Implementation in Embedded Systems

Quality-oriented tracking and tracing (t&t) for chilled transport and storage is primarily based on permanent temperature supervision. Literature reports [9, 10, 11, 12] and preliminary tests [13] revealed spatial deviations of 5 °C or more over the length of a container or temperature zone in trucks with separated compartments. Simulations with shelf life models showed that these temperature differences could lead to variances in the remaining quality of more than 50 % at the end of transport [13]. In cool down to deep freezer mode, deviations of up to 10 °C were still present after 5 hours.

Evaluation of sensor data by current t&t systems is centralized. But due to limited communication bandwidth and the cost of mobile or satellite communication it is not feasible to transfer the vast information created by spatial temperature supervision to a remote server for evaluation. Therefore we shift part of the logistic decision system into the means of transport. Equipped with an embedded processing unit, the truck or container can act as an autonomous entity. In section 4.1 we show by example how an embedded unit can also handle parts of the route planning. As test platform we selected a credit card sized ARM XScale processor module with a clock rate of 400 MHz and 32 MBytes SDRAM memory (www.dilnetpc.com).

3.1 Representation of Logistical Objects by Software Agents

In the context of our work the implementation paradigm for autonomous processes is based on software agents. Agents communicate by asynchronous messages independently of their current location to achieve their goals [14]. Currently the most common environment to test and implement agents is the Java Agent DEvelopment framework (JADE) [15].

A special version of JADE, the Light Extensible Agent Platform (LEAP), was developed for devices with limited computing power and memory, e.g., mobile phones [16]. The programming language Java is required by the agent framework. Although Java is not a common language to operate microcontrollers, it offers useful advantages for our setting like platform independency and the execution of dynamic code. New programming environments like the Jamaica Virtual Machine (www.aicas.com) make it feasible to efficiently run complex Java programs on embedded systems [17].

As example process we investigated the transfer of a software agent representing the freight-specific supervision instructions. After the loading of a freight item to a new means of transport is detected by a RFID reader, the vehicle sends a request for the corresponding agent. After being transferred to the vehicle the agent is re-started on the local platform provided by the truck or container. Finally a confirmation message is sent to the previous location of the freight.

By optimizations of the agent framework the execution time for the above described process has been reduced from 15 to 6 seconds [18]. Major bottlenecks were the translation of messages into the FIPA-ACL format and JADE internal services.

3.2 Interpretation of Sensor Data and Quality Assessment

A wireless sensor system for spatial temperature supervision is provided by the means of transport. The complexity of fruit ripening processes or quality decay of other products can not be reduced to a simple temperature threshold checking. Based on the Arrhenius law on reaction kinetics the freight supervision agent calculates the current loss in shelf per time unit, depending on the measured temperature [7]. The feasibility of this concept for automated quality evaluation by mobile freight agents was demonstrated by our reduced scale prototype of an intelligent container [19].

4 Distributed Solution of Route Planning Problems

Planning for a logistic domain is a real-time problem with relaxed timing constraints of hours or even days rather than seconds. Although deliberating can be done concurrently to acting it is constrained by onboard computational resources, which may be very limited regarding computing power and memory (see section 3).

Classical planning based on theorem proving dates back to the 1960's. The widely known STRIPS system was introduced in [20]. STRIPS-like planners can cope with "fully observable, deterministic, finite, static, and discrete" environments [21]. Even though numerous approaches exist that address one or the other restriction, none of them can remove all of them.

Decision-theoretic approaches, especially Markov Decision Processes, add the ability of handling uncertainty, observations, the concept of utility, and also partial observability [22, 23]. Several approaches and systems have been proposed and implemented to address the problem of timing constraints but still include major shortcomings. A discussion of these approaches and a potential solution for the logistics domain can be found in [24].

For evaluation purposes in this paper the planning problem is reduced to a simple route planning algorithm which is guaranteed to find the optimal solution at the cost of hyperexponential computation time.

4.1 Distributed Planning by Truck Agents

In order to illustrate how distributed problem solving could shift part of the route planning into the local processing platform of the vehicle we use the following example:

> A truck is loaded with a number of perishable products that should be delivered to multiple customers in different destinations. The products have different initial shelf lives, which are reduced by journey times. The task is to optimize the route in a way that product losses are avoided due to zero shelf life at delivery. The sum of the remaining shelf lives at time of delivery should be maximized. In an extended scenario unexpected shelf life losses by temperature changes force re-planning of the route. A solution for this problem has to consider two sources of information: First the current shelf life state of the loaded products; these data are directly available inside the vehicle. And secondly information about travel distances and traffic situation. These data would be provided by a remote traffic data base.

The above example describes a special case of a Traveling Salesman Problem. To solve this kind of problem we apply a heuristic approach by splitting the route search between remote traffic data base and local vehicle. This distributed problem solving results in increased robustness and autonomy. A remote Route Planning Agent (RPA) searches for routes with low total driving time to the delivery sites that have not been visited so far.

The proposals are fetched by the Local Vehicle Agent (LVA), which selects one of the possible routes based on an evaluation of the achievable grade of goal fulfilment, given as the sum of remaining shelf lives at delivery and avoidance of zero shelf life. After arriving at the next customer the LVA requests new route proposals, if the current grade of goal fulfilment is not satisfactory or unexpected quality losses make re-planning necessary.

By this approach the delivery service can quickly react to sudden changes in freight quality. Even if the RPA could not be reached due to communication failure, the vehicle can continue its planning based on the proposals that are already known.

As an additional win this approach also secures information about shelf life and especially about quality problems as internal state because the actual decision parameters remain local to the vehicle.

4.2 Experimental Evaluation

An important requirement regarding the RPA is that the route proposals must be substantially different from each other. If, e.g., only a list of the routes with shortest driving distance is provided, these routes often resample each other and deprive the LVA of the freedom it needs to find optimal paths in terms of shelf life.

The following approach was tested as example: For each step with N customers still to be visited, the LVA requests a proposal for a short, but not necessarily the shortest round trip. The truck could drive from its current position to one of the N destinations as starting point for the round trip and continue in clockwise or counter clockwise direction. The resulting $2 * N$ options are evaluated according to their grade of goal achievement. After arriving at the first customer the procedure is repeated by sending a new request to the RPA. This approach was compared to a full optimization by a software simulation. For a fixed map of twelve destinations, freight items with a shelf life set by random values had to be delivered to the customers.

In two thirds of the simulation runs, the systems finds a route that delivers as much items before expiration as theoretical possible (see Table 1 - row A). The higher flexibility of the system and lower planning effort entails a small reduction of the remaining shelf life at delivery to 92% in average of the possible value. To avoid package losses for the remaining third (see Table 1 - row B) of experiments, further improvements of the local planning process are necessary. Such a system should detect cases in which the described simple heuristic is not sufficient and switch to a more costly strategy, e.g., requesting additional route proposals.

Table 1. Summary of 600 simulation runs. The points give a measure for the remaining shelf life at delivery. Late deliveries with zero shelf life were punished with -1000 points (row B)

	Runs	Local planning	Optimal	Ratio
A (no losses)	402	252,73 points	272,02 points	92,62% ± 7,37
B (with losses)	198	-813,64 points	218,93 points	—

5 Conclusion

In this paper we have shown that it is indeed possible to create autonomous logistic decision making systems based on state-of-the-art deliberation methods of distributed artificial intelligence. By skilled division of planning tasks, as shown by our software simulations, it is feasible to implement decision instances onto off-the-shelf embedded hardware. These autonomous systems are capable of on-line monitoring of certain parameters that a logistic service provider would not want to reveal to third parties and base decision on these data by considering information collected from third party providers.

The solution we propose is autonomous on a level we consider clearly above *operational*. Whether it is jusitified to claim *tactical* autonomy has to be shown by further experiments and refinement of the approach. To increase their capabilities the implementation of the JADE agent platform should be extended to a deliberation on a BDI basis.

Acknowledgement. This research was supported by the German Research Foundation (DFG) as part of the Collaborative Research Centre 637 "Autonomous Cooperating Logistic Processes".

References

1. P. Scheer. Optimising supply chains using traceability systems. In A. Furness, editor, *Improving traceability in food processing and distribution*, pages 52–64. Woodhead, Cambridge, UK, 2006.
2. J. P. Emond and M. Nicometo. Shelf-life prediction and FEFO inventory management with RFID. In *Cool Chain Association Workshop*, November 2006.
3. I. J. Timm. Strategic management of autonomous software systems. TZI-Bericht 35, Center for Computing Technologies, University of Bremen, Bremen, 2006.
4. A. S. Rao and M. P. Georgeff. BDI agents: From theory to practice. In *Proceedings of the First Intl. Conference on Multiagent Systems*, pages 312–319, 1995.
5. S. J. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, New Jersey, 2 edition, 2003.
6. D. J. Tanner and N. D. Amos. Modelling product quality changes as a result of temperature variability in shipping systems. In *Proceedings of the International Congress of Refrigeration (ICR 2003)*, Paris, 2003. International Institute of Refrigeration. Published on CD-ROM.
7. R. Jedermann, R. Schouten, A. Sklorz, W. Lang, and O. van Kooten. Linking keeping quality models and sensor systems to an autonomous transport supervision system. In J. Kreyenschmidt and B. Petersen, editors, *Cold Chain Management – 2nd international Workshop, Bonn, 8.–9. May 2006*, pages 3–18. University Bonn, 2006.
8. M. Wooldridge. Intelligent agents. In G. Weiss, editor, *Multiagent Systems. A Modern Approach to Distributed Artificial Intelligence*, pages 27–77. The MIT Press, April 1999.

9. J. Rodríguez-Bernejo, P. Barreiro, J. I. Robla, and L. Ruíz-García. Thermal study of a transport container. *Journal of Food Enginnering*, 80(2):517–527, 2006.
10. H. Punt and M. Huysamer. Temperature variances in a 12 m integral reefer container carrying plums under a dual temperature shipping regime. In *International Conference Postharvest Unlimited Downunder 2004*, volume 687 of *Acta Horticulturae*, pages 289–296. International Society for Horticultural Science, 2005.
11. D. J. Tanner and N. D. Amos. Temperature variability during shipment of fresh produce. In *International Conference: Postharvest Unlimited*, volume 599 of *Acta Horticulturae*, pages 193–204. International Society for Horticultural Science, 2003.
12. J. Moureh and D. Flick. Airflow pattern and temperature distribution in a typical refrigerated truck configuration loaded with pallets. *International Journal of Refrigeration*, 27(5):464–474, 2004.
13. R. Jedermann and W. Lang. Semi-passive RFID and beyond - steps towards automated quality tracing in the food chain. *International Journal of Radio Frequency Identification Technology and Applications (IJRFITA)*, 2007. In press.
14. M. Wooldridge and N. R. Jennings. Intelligent agents: Theory and practice. *Knowledge Engineering Review*, 10(2):115–152, 1995.
15. F. Bellifemine, A. Poggi, and G. Rimassa. Developing multi-agent systems with a FIPA-compliant agent framework. *Software – Practice & Experience*, 31(2):103–128, February 2001.
16. A. Moreno, A. Valls, and A. Viejo. Using JADE-LEAP to implement agents in mobile devices, 2007. `http://jade.tilab.com/papers/EXP/02Moreno.pdf`, last visit: 2007-03-05.
17. F. Siebert, editor. *Hard Realtime Garbage Collection*. aicas GmbH, 2002.
18. R. Jedermann and W. Lang. Mobile java code for embedded transport monitoring systems. In R. Ester, editor, *Proceedings of the Embedded World Conference 2006*, pages 771–777. Franzis Verlag, 2006.
19. R. Jedermann, J. D. Gehrke, M. Becker, C. Behrens, E. Morales-Kluge, O. Herzog, and W. Lang. Transport scenario for the intelligent container. In M. Hülsmann and K. Windt, editors, *Understanding Autonomous Cooperation & Control in Logistics. The impact on Management, Information and Communication and Material Flow*, pages 365–404. Springer, Berlin, 2007.
20. R. E. Fikes and N. J. Nilsson. STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 2(3–4):189–208, 1971.
21. M. E. Pollack. The uses of plans. *Artificial Intelligence*, 57(1):43–68, 1992.
22. C. Boutilier, T. Dean, and S. Hanks. Decision-theoretic planning: Structural assumptions and computational leverage. *Journal of Artificial Intelligence Research (JAIR)*, 11:1–94, 1999.
23. T. Wagner, A. Raja, and V. Lesser. Modeling uncertainty and its implications to sophisticated control in TAEMS agents. *Autonomous Agents and Multi-Agent Systems*, 13(3):235–292, April 2006.
24. M. Lorenz, C. Ober-Blöbaum, and O. Herzog. Planning for autonomous decision-making in a logistic scenario. In *Proceedings of the 21st European Conference on Modelling and Simulation (ECMS 2007)*, pages 140–145, June 2007.