An Adaptive Solution to Dynamic Transport Optimization

Klaus Dorer and Monique Calisti
Whitestein Technologies AG
Pestalozzistrasse 24
8032 Zürich, Switzerland
{kdo,mca}@whitestein.com

ABSTRACT

This paper describes LS/ATN, Living Systems®Adaptive Transportation Networks, an agent-based solution we have developed to solve transportation problems in the charter business logistics. LS/ATN provides automatic optimization and execution capabilities that extend the existing planning systems accordingly. To describe our solution and analyse its performance, we report on a real case scenario in which transportation requests of a big logistics provider were optimized. Besides describing the agent approach and the LS/ATN features we stress the necessity to integrate such agent system into a real-world IT architecture. Finally, we show that our adaptive solution produces significantly better results in real case scenarios than what achieved with manual optimization of professional dispatchers.

Categories and Subject Descriptors

I.2.11 [**Distributed Artificial Intelligence**]: [Intelligent Agents]; H.4.2 [**Information Systems Applications**]: [Logistics]

General Terms

Algorithms, Economics

Keywords

transportation logistics, agent system, LS/ATN, dynamic optimization

1. INTRODUCTION

As with many industries and markets, the logistics sector faces the extensive and fundamental challenges associated with globalization. With shrinking margins, and in many cases being barely able to cope with immense cost pressure, companies are having to substantially revise their product and service offerings, business processes, and operational systems. Being aware that especially in the transportation business, the most optimal utilization of available capacity has proven to be the most important critical success factor, many logistics companies have implemented computer- based systems for strategic network planning and short-term route optimization planning. While a number of such solutions allow the automatic creation of the dispatching plans, they usually do not, or only marginally, support the case of plan deviations resulting from unexpected, or previously unknown events.

Recent experience has shown that conventional systems for network planning and transportation optimization are limited in their ability to cope with the increasing complexity, and especially with the dynamics, of a noticeably globalized transportation business. These systems were developed for relatively stable, not overly complex, and mostly repetitive transportation processes, for which it is straightforward to create optimized plans using established analytical algorithms. Conventional planning systems, however, usually fail in turbulent environments, where plans have to be adjusted to new, changing conditions within shortest time frames (at runtime, in real-time). Examples are new transportation orders or last minute changes of orders, unexpected shortages of resources due to traffic jams, breakdowns, or accidents. In such cases, the Execution Control, or the respective Event Management, has to be performed more or less manually by the responsible human dispatchers, a Sisyphean task!

In cooperation with one of the worldwide leading logistics providers, we have developed the agent-based solution LS/ATN that:

- Provides broad support to the dispatchers to cope with the challenges of Execution Control in complex real-world environments.
- Significantly reduces the transportation costs by making use of innovative conceptual and technological measures.
- Increases the service quality logistic providers can offer to their customers.

In particular, cost reduction is achieved mainly by:

- Automatic dispatching support (including handling of unexpected events).
- Dynamic transportation optimization in real-time.
- Increased capacity utilization through optimized allocation of shipments to trucks.
- Comprehensive integration of telematics services.
- Synergy effects through combination of different logistics network types.
- Increased transparency and visibility throughout the network.
- Combination of different traffic types (round trip, linehaul meetings, multi-modal traffic).

AAMAS'05, July 25-29, 2005, Utrecht, Netherlands.

LS/ATN supports, optimizes, and to a large extent automates the central challenge of a logistics provider, the *dispatching*. Used as stand-alone solution LS/ATN is able to support the whole dispatching process. Based on its modular design and open architecture, the solution however can also run as an integrated module within an existing application. The *optimization engine* is the kernel of LS/ATN, which continuously executes the dynamic transportation optimization [1]. It can be parameterized and configured for different business models. The flexible optimization algorithm used is strictly cost based: all optimization factors are internally mapped to a common cost basis, an economic model. Furthermore, the optimization of LS/ATN is able to take into account both hard and soft constraints. *Hard constraints* (e.g. opening hours of a location) always have to be kept. In contrast, *soft constraints* (e.g. delayed delivery) can be violated to some degree, with the aim of reducing costs, provided that service level agreements are fulfilled, and that alternative transport variants would cause significantly higher costs. LS/ATN is based on a *dynamic optimization approac*h insofar, as that the system always, immediately after the occurrence of an event, again strives for the optimum. Opposite to the well known analytical approaches, LS/ATN first tries by means of heuristic methods to solve the problem in a local context. Then, in a second step, it immediately starts searching for additional optimization options in the overall network. The described optimization can also be run as simulation. This is useful for planning purposes, and to support strategic decisions.

In this perspective, we argue and show that intelligent transport optimization systems are able to dynamically adapt transport plans and schedules according to possible deviations and unforeseen events have the great potential of reducing overall transport costs, by improving the coordination process of distributed dispatchers and improve resource consumption.

The remainder of the paper is organized as follows: Section 2 introduces the problem domain, the constraints that must be taken into account in order to obtain delivery routes that are navigable in the real-world and depicts the cost model at the basis of our optimization approach. Section 3 describes the optimization algorithms and the agent-based design that enable LS/ATN to deal with problem instances consisting of thousands of transportation requests. Section 4 gives an overview of LS/ATN and explains how dispatchers interact with the agent system during the dynamic optimization and execution phases. Section 5 presents and discusses the empirical results obtained by solving a real world optimization problem, before final remarks concluding the paper (Section 6).

2. THE TRANSPORTATION PROBLEM DOMAIN

The transportation problem addressed by LS/ATN consists of finding optimal routes for serving transportation requests (orders) of a (usually large) set of customers. These orders have to be picked up and delivered at specific customer locations. Pickup and delivery has to occur within a specific time window, even though time constraints can potentially be violated within some tolerated degree. A limited number of trucks is available to transport the orders. These trucks may be of different type and capacity and they are usually available in different locations. Additionally, the trucks' drivers have to observe drive time restrictions.

The problem is *dynamic*, because the orders are not all known in advance. New orders can be received and have to be accounted for during the optimization process itself. Further dynamicity is induced by changes that can occur to the orders. The real size of an order, for instance, might have to be corrected only when the driver is already at the pickup location. Also transportation capacity is subject to dynamic change. Trucks may be delayed due to traffic jams or other unforeseen problems or can even become temporarily unavailable.

[1]The LS/ATN solution will become operational at one of Whitestein's logistic customers from April 2005.

In our framework, the term *node* is used to indicate the combination of both location and time (arrival and departure time) for a given truck at that specific location. A *leg* is the path between two nodes. A *route* is a sequence of nodes a truck visits. The truck is assumed to be empty at the beginning and at the end of a route. The terms *order* and *transportation request* are used synonymously to indicate a customer's request to transport some goods from a pickup location to a delivery location. The *solution* of this kind of transportation problem consists of a set of routes including a schedule that specifies the times at which the trucks must be at selected locations. The result is also called a *delivery plan* and its quality or goodness is given by an *objective function*, which in our case is a cost-based function (see Section 2.4). More technical descriptions of this problem domain can be found in [8, 9].

2.1 Characterization of Orders

Every order is characterized by the following parameters:

- Order type
- Volume
- Weight
- Pickup location
- Pickup time window
- Loading time
- Delivery location
- Delivery time window
- Unloading time
- Hazard category of dangerous goods
- Equipment necessary to handle the order
- Time at which the order is known to the system (for simulation runs)

As previously mentioned, orders are usually not all available when starting the route planning and delivery scheduling process, but can arrive subsequently. Therefore, some orders may have already been served when new ones arrive.

2.2 Characterization of Trucks

Every truck is characterized by the following parameters:

- Truck type
- Capacity (volume)
- Capacity (weight)
- Special equipment
- Start location
- Availability time
- Tariff

The tariff indicates a cost class to which the truck belongs (see Section 2.4). A mapping function defines which order types fit to which truck type.

2.3 Problem Constraints

The optimization algorithm has to obey a number of constraints considered during the calculation of routes. Constraints are classified as hard constraints and soft constraints. *Hard constraints* express conditions that must hold and include:

- Load constraints:
  - Precedence (pickup has to be before delivery)
  - Pairing (pickup and delivery have to be performed by the same truck)
  - Capacity limitation of a truck
  - Weight limitation of a truck
  - Order type and truck type compatibility
  - Availability of required equipment
  - Regulations for hazardous goods combined loading on one truck
- Time constraints:
  - Earliest pickup
  - Latest pickup
  - Earliest delivery
  - Latest delivery (all of these four if violated by more than the allowed delay/early time)
  - Opening hours of customers
  - Maximum duration of a route
  - Lead time for ordering a truck on the spot market
  - Driving time regulations for drivers

Soft constraints express conditions that may be violated to some degree. They produce violation costs to discourage, but still allow such violations (see section 2.4). The following soft constraints are considered:

- Earliest pickup time
- Latest pickup time
- Earliest delivery time
- Latest delivery time

2.4 Cost Model

The major concern of logistics companies in general is to reduce their costs [7]. Therefore, the objective function used to evaluate the optimization results is cost-based. In our framework, the cost model was defined in order to properly take into account real-world costs and constraints and thereby enable comparison between the optimization results of our agent-based solution with real transport plans created manually by professional dispatchers. The cost model distinguishes between three kinds of costs: *variable, fixed* and *violation* costs.

Variable costs are assigned according to the length of a route, the amount of transported load and the start and end location of the specified route. The start and end locations are important to distinguish regional cost differences on the spot market. Prices for hiring a spot market truck vary according to the region where a route starts. Also it is more expensive to get a spot market truck if the route ends in a place where it is unlikely for the truck to get another order back. The values used for regional variable costs have been derived from real data. A discount is granted if multiple consecutive routes can be combined (tramp routes). This reflects the fact that in the charter business cheaper offers are given to such routes by a subcontractor. A higher discount is granted if the last route of consecutive routes terminates close to the start location (back routes). This reduces the workload of the subcontractor in terms of looking for freight (and drivers) for a back route of the truck.

Fixed costs may be assigned to a truck of the own fleet. In the analyzed cases, fix costs have been replaced by minimum costs. If the truck's variable costs are below a minimum threshold, the final costs of the route correspond to the minimum costs independent of the trip duration and usage of the truck.

Finally, *violation costs* arise when soft constraints are violated. They are used by the optimizer during the solving process. Violation costs are introduced so that constraints are only violated if it is "worthwhile", i.e. only if the cost savings achieved by violating a soft constraint exceed the violation costs such constraint violation implies.

3. AGENT-BASED OPTIMIZATION

Whenever a new order is made available to the system, the current delivery plan is updated. This is done in a two-phase approach: first, a new valid solution is generated including the new order. Then, the obtained solution is improved by cyclic transfers of orders. The next two sections explain these two steps and Section 3.3 describes how these algorithms can be distributed across multiple agents.

3.1 Solution Generation

The first step taken when receiving a new order is the generation of a new valid solution. The algorithm used for this is a sequential insertion of orders [2]. All available trucks are checked to see if they are able to transport the order and what additional costs are incurred.

Each time an order is added to a route either none, one or two new nodes must be added depending on whether the pickup and/or the delivery nodes of such an order are already part of the route or not. In this way, multiple pickups and deliveries per location are possible. Based on the insertion algorithm, it is also possible that a truck visits the same location several times. Finally, for all combinations of existing nodes a quote is generated for inserting a new pickup and delivery node.

When no truck can transport an incoming new order, which means all trucks would have to break hard constraints, the order's service level is lowered. This implies the order may be transported with configurable soft constraint violations, allowing violations of pickup and delivery times. If still no truck is found the order remains unallocated. This is usually the case if order data is invalid and the consequent transport planning problem is over-constrained (e.g. impossible driving times).

3.2 Optimization Approach

Sequential insertion with requests for quotes to all trucks potentially produces suboptimal solutions, see for instance the example shown in Figure 1. Order 1 is the first to arrive and is assigned to truck 1's route. Order 2 is also optimally assigned to truck 1's route since this produces least additional costs (and kilometers). When order 3 arrives truck 1 is fully loaded, therefore a new truck 2 is used for order 3 and later for order 4.

In order to improve the solution a further optimization step is performed by cyclic transfers between trucks [10, 5]. A cyclic transfer is an exchange of orders between routes. Figure 2 shows how the suboptimal example in Figure 1 is improved by a transfer of order

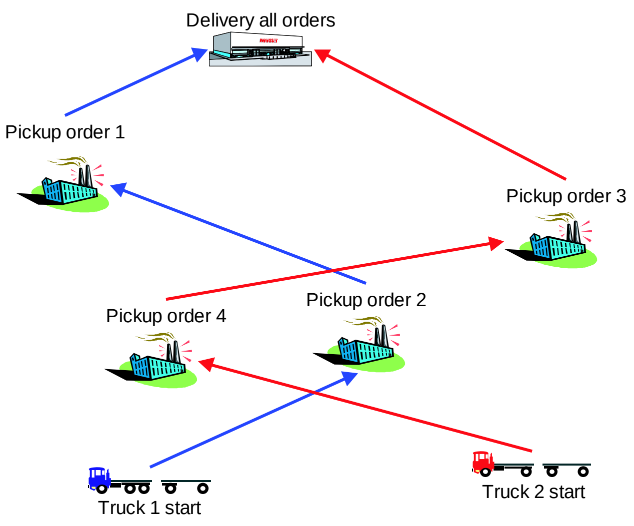

Figure 1: Example of a suboptimal solution given by the insertion algorithm for 4 orders.

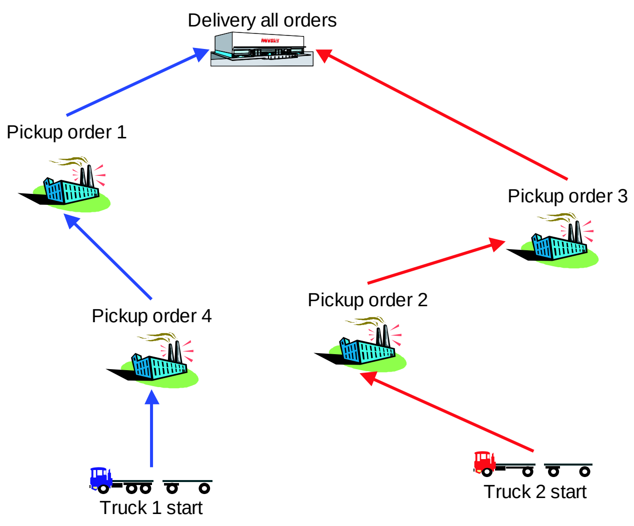

Figure 2: Optimal solution, for the example given in Figure 1, after a transfer.

2 and order 4.

The optimization procedure has to determine which order transfers should be triggered. A simple strategy for this is a hill climbing approach that selects the most cost-saving transfers from a neighborhood of possible transfers. This hill climbing process is continued with all changed routes until no more cost-saving exchanges can be performed.

3.3 The Underlying Agent-Based Design

Solving transportation problems can be distributed among multiple interacting agents in order to (1) achieve scalability of performance with growing sizes of problem instances; (2) directly reflect the distributed nature of transportation networks/organizations and decision making centers; (3) facilitate the handling of local deviations without the need to propagate local changes and recompute the whole solution, and (4) increase robustness (avoiding single point-of-failure).

Several agent-based designs are possible to distribute the work. The most radical one is to represent each truck by an agent [1, 4]. Solution generation by sequential insertion is then handled by a contract-net interaction protocol [1]. The optimization algorithm can be modified in order to trigger a transfer between two trucks whenever this improves the objective function, rather than to look for the best of all possible transfers. This modification improves the handling of multiple routes which can change concurrently. In the modified version of the optimization algorithm, the hill-climbing is no longer along the steepest slope and it is possible that more transfers are necessary to achieve the same solution. Moreover, the optimization process may end in a different local optimum. In such a fully distributed architecture, truck agents with a changed route start an optimization process in parallel to the other active truck agents. The main advantage of such an approach is in its fine granularity and high scalability, while its main disadvantage stems from a considerable overhead in computation time and resource usage. The overhead in computation time is mostly due to more expensive agent communications when compared to a fully centralized solution, while the overhead in resource usage depends on the memory and processing footprint of an agent.

The agent design chosen for our work reflects the way logistics companies today manage the complexity of this domain. The transportation business is usually divided into *dispatching regions*. Orders arriving at a region are first tentatively allocated and possibly optimized within that region. If the order's pickup or delivery location is in a different region, the other region is also informed and asked to handle the order if it can do so in a cheaper way. In our agent-based framework, distinct software agents represent different regions. All trucks starting in the region of an agent are managed by a local *AgentRegionManager*. Incoming orders are distributed by a centralized *AgentDistributor* according to their pickup location (see Figure 3). Sequential order insertion can be achieved as summarized in Section 3.1. The only difference is that 'all trucks' in this case is restricted to all trucks within a given region. The quality of the solution (from a global perspective) is expected to decrease with the number of defined regions since order insertion does not take trucks of other regions into account. This is compensated by optimization transfers. The optimization algorithm within a region is the same as described in Section 3.2. Additionally, trucks with routes spanning across other regions may also initiate transfer requests among regions. The main advantage of this latter design stems from its direct mapping to today's transport business organization and its good scalability. The computational overhead incurred by such a multi-agent based solution is also much lower than that occurring in a fully distributed solution. Nevertheless, besides degradation of the solution's quality when compared to a fully centralized approach, the main disadvantage (also with respect to the fully distributed option) is that optimization within a region and among regions has to be handled slightly differently.

4. THE LS/ATN SOLUTION

The agent-based optimization described above is the core part of the LS/ATN logistics optimization solution. The two major requirements for such a solution to become a real-world production system are that, first, LS/ATN has to be integrated into deployed IT infrastructures and interact with a number of other systems and, second, LS/ATN users have to be enabled to interact with the system in a way that is possible to overrule suggestions and handle incoming events.

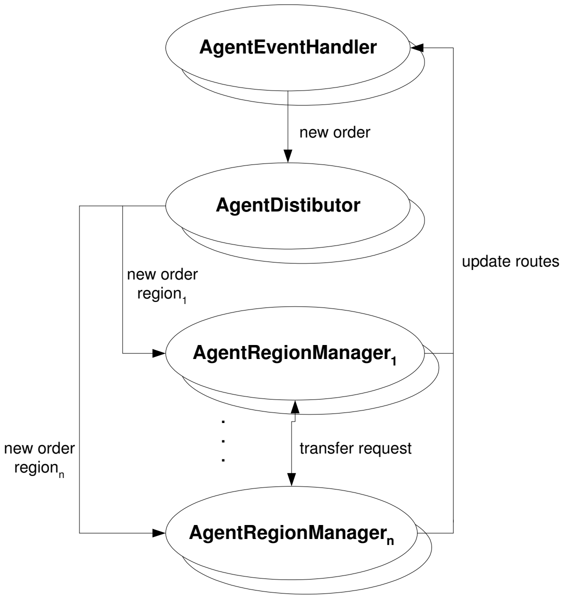

Figure 3: A summarized view of a region-based agent design.

4.1 System Integration

In a real-world production environment an agent based solution has to interact with a number of other systems. In the case of LS/ATN, this firstly translates into the need of integrating the agent-based optimization solution with the deployed *transport management system* (TMS). Proper interfaces are necessary to the "order acquisition" process, and to the deployed billing and reporting TMS specific sub-systems. In addition, in order to create transport plans *geo-coding information* has to be available to compute distances as well as drive times between locations. This usually also requires to retrieve and display corresponding geographic maps. Transportation plans and other information have to be stored in a proper *database* infrastructure to ensure persistence. Moreover, integration into a *telematics infrastructure* is necessary. Tracking information (pickup and delivery times) has to be passed to LS/ATN to react to plan deviations and update current plans. In addition, LS/ATN also supports to offer routes to *freight exchange* platforms such as Teleroute or TimoCom. Finally, all information a specific user needs to visualize and manipulate has to be properly displayed by means of appropriate graphical user interfaces.

Figure 4 shows the detailed view of a route as suggested by an LS/ATN optimizer agent. The dispatcher is informed about the locations to be visited along the selected route, namely time schedule, orders, distance of the route etc. (see top table). In the central part of the display, additional information such as pickup and delivery time windows, load and weight can be found. A detailed schedule in tabular and graphical representation can be seen in the bottom left part of the figure. Finally, a map shows the depiction of the selected route.

4.2 User Interaction

Agents proposed transport plans have always to be considered as suggestions. Human dispatchers have to be able to overrule such suggestions and change transport plans according to real-world requirements, policies or decisions not accounted into the agent system. To this purpose, LS/ATN supports a number of interaction modalities for dispatchers. Orders may be added to a truck, removed from a truck and moved between two trucks by drag and drop option in the dispatcher board (see Figure 5 arrows 1 - 3). The dispatcher may also change the order in which nodes are visited (arrow 4) or insert new nodes to be visited to redirect a truck. In all these operations, the dispatcher is supported by the optimizer agents that inform about constraint violations arising. Furthermore, operations are not accepted if the violations are above a level that is acceptable for manual dispatching. Manually dispatched routes will then not be changed by the optimizer agents although suggestions to add orders to these routes are still performed. However, the dispatcher may return orders and routes back to the control of the optimizer agents.

Another way for the dispatcher to interact with the system is event-driven. Whenever an unexpected or important situation arises the dispatcher is informed about the event. This happens, for example, if an order changes or is cancelled. If the corresponding route is still in planning the optimizer agent automatically re-optimizes the transport plan according to the changed situation and informs the dispatcher of the change. If the route is already in execution the agent suggests how to change the transport plan but leaves the final decision to the dispatcher. Other events the dispatcher is informed about are opportunities to add new orders to already executing routes or if a new order arrived to the system that is not transportable due to over-constraint order information.

Finally, the dispatcher may feedback tracking information during the execution of a route. This is usually done automatically by a telematics system, but may be changed or improved by the dispatcher if additional information is made available elsewhere. Tracking information includes times at which a pickup or delivery of an order has occurred. The dispatcher is supported in this work by an agent that proactively informs the dispatcher once loading or unloading should have happened according to plan. Tracking information may also be information about the future of a route. As soon as the dispatcher gets feedback from the driver about delays (or being ahead of schedule) the expected arrival time at specific nodes may be changed accordingly. The optimizer agent recomputes then the schedule and informs the dispatcher in case an order on that route may not be transported in time. The route may then be proactively changed.

5. EXPERIMENTAL RESULTS

Empirical tests were run for a European logistics company with the aim of evaluating the potential cost savings when introducing the LS/ATN transport optimization system. The dataset we analyzed contains roughly 3,500 real-business Orders. The constraints and cost model have been modelled and used as described in Section 2. While the primary goal of the logistics company was reduction of costs, further objectives have focused on the reduction of the number of deployed trucks and the total amount of driven kilometers, while increasing the utilization of the trucks. The agent platform used for the evaluations is Whitestein Technologies' *Living Agents Runtime System* (LARS)[2].

In order to produce results of agent-based optimization that are comparable with the results achieved by the professional dispatchers, it has been necessary to address a number of issues. We used the same underlying geo-coding information system (distance and drive time information) that was used by the dispatchers. The average cost values have been obtained from the real costs incurred when purchasing the trucks for these orders. Soft constraint vio-

[2]LARS is a predecessor of Whitestein's now available LS/TS, *Living Systems Technology Suite*, see http://www.whitestein.com for more details.

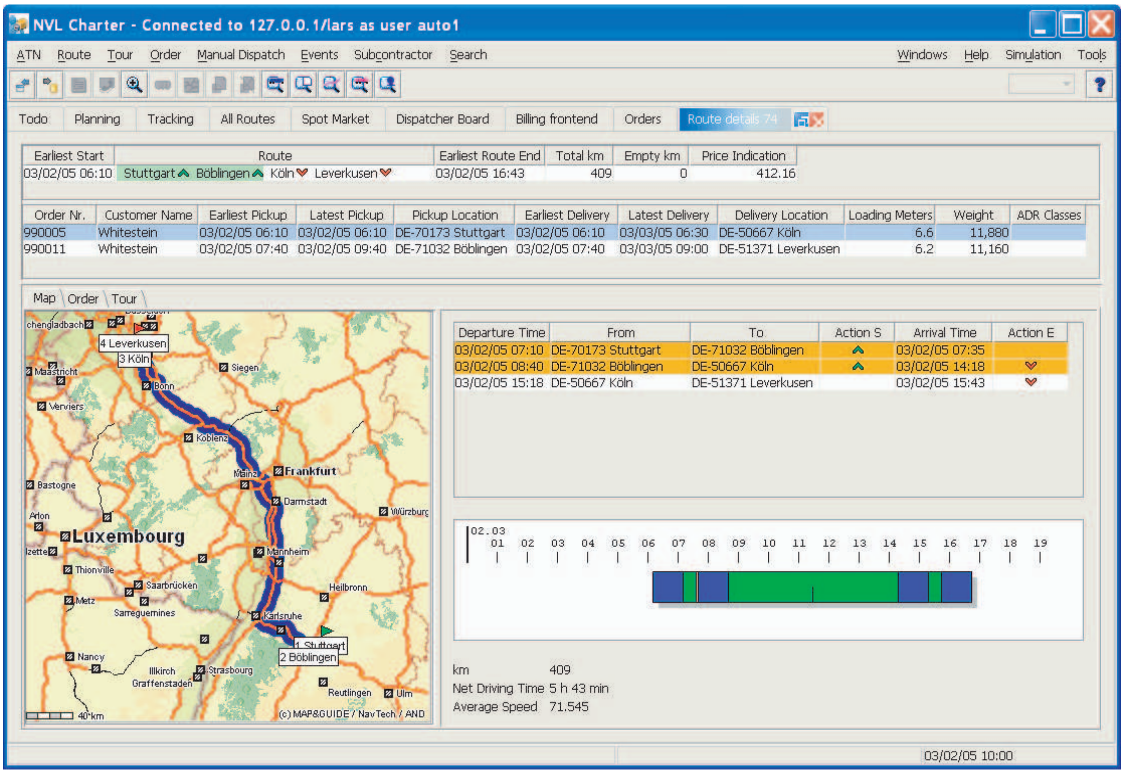

Figure 4: Details of a route in LS/ATN as suggested by an optimizer agent.

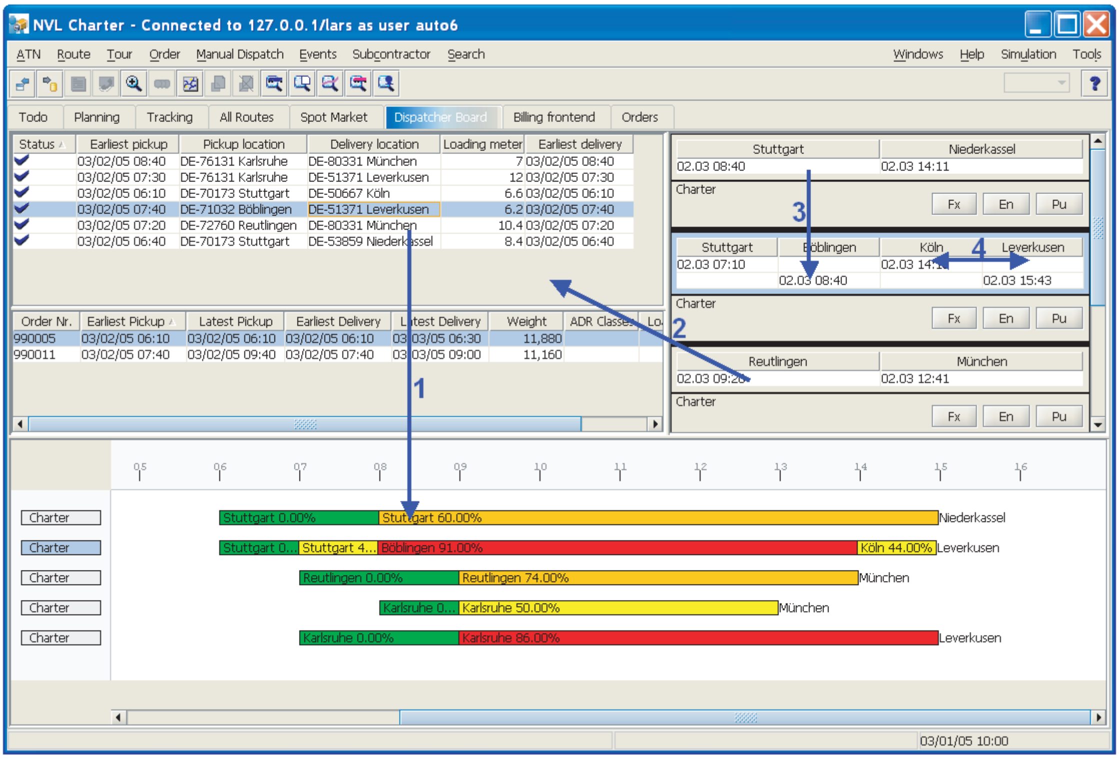

Figure 5: The LS/ATN dispatcher board for supporting manual dispatcher-driven interaction (Example actions: 1-3 Moving orders to, from and between trucks, 4 changing the visiting order of nodes).

Evaluation Parameter	Savings (agent-based)
Overall Cost	11.7%
Driven Kilometers	4.2%
Deployed Trucks	25.5%

Table 1: Savings gain achieved by LS/ATN agents in comparison with manual dispatching.

lations were only allowed if order data did not allow an in-time pickup or delivery. The resulting delivery plan was checked by dispatchers for feasibility and drivability. Table 5 summarizes the major results. A total of 11.7% cost savings was achieved, where 4.2% of the cost savings stem from a corresponding reduction in driven kilometers. Another 2.2% is achieved by increasing the number of cost-saving tramp routes by 380%. The remainder stems from buying cheaper trucks. The cost-based optimization prefers routes that start in cheap regions. An additional important achievement is that the number of trucks used is 25.5% lower in comparison to the manual solution. This is due to a higher utilization of the trucks and an on-average longer usage of a single truck. Even if some of the potential is lost on the way from a simulation to a production system, optimization results achieved on various customers' data proved that a saving potential of 3% to 6% of current transportation costs is achievable by using LS/ATN.

6. CONCLUSION

In this paper, an agent-based solution to solve real-world dynamic transport optimization problems has been presented. The problem model, including constraints specification and cost function definition, has been expressed in a way that allows efficient computation of transportation plans (validated in real-world transportation scenarios) that can be executed in daily business. Our agent-based solution, LS/ATN, produced significantly better outcomes in comparison with the results of professional dispatchers.

In real-world production environments, optimization is a continuous, non-stop process. With more computational power available in the near future, the use of even more sophisticated optimization approaches can be envisaged to further explore and refine the solution space and thereby improve the quality of the achieved results [6]. In this perspective, dynamic coordination of the various computational agents makes it possible to more easily orchestrate and master a change in the deployed problem solving techniques. LS/ATN, which exploits these advanced concepts of problem solution as well as the corresponding solution deployment, is a full-fledged agent-based application that provides a solution to a real-world challenge: increasing the profitability and thus market success of logistics providers through cost reductions of between 3% and 6%. Looking at the road logistics market in Europe which has a total revenue of around 167.5 b. Euro [3], this is a potential saving of at least 5 b. Euro.

7. REFERENCES

[1] K. Fischer, J. P. Müller, and M. Pischel. Cooperative transportation scheduling: an application domain for DAI. *Journal of Applied Artificial Intelligence*, 10, 1995.

[2] J.-J. Jaw, A. R. Odoni, H. N. Psaraftis, and N. H. Wilson. A heuristic algorithm for the multi-vehicle advance request dial-a-ride problem with time windows. *Transportation Research*, 20 B(3):243–257, 1986.

[3] P. Klaus. *TOP 100 in European Transport and Logistics Services*. Deutscher Verkehrs-Verlag, 2004.

[4] Robert Kohout and Kutluhan Erol. In-time agent-based vehicle routing with a stochastic improvement heuristic. In *Proceedings of the 6th National Conference on Artificial Intelligence (AAAI-99); Proceedings of the 11th Conference on Innovative Applications of Artificial Intelligence*, pages 864–869, Menlo Park, Cal., July 18–22 1999. AAAI/MIT Press.

[5] Snezana Mitrovic-Minic. Pickup and delivery problem with time windows: A survey. Technical Report TR 1998-12, School of Computing Science, Simon Fraser University, Burnaby, BC, Canada, May 1998.

[6] W. P. Nanry and J. W. Barnes. Solving the pickup and delivery problem with time windows using reactive tabu search. *Transportation Research*, 34B:107–121, 2000.

[7] Exel News. Uk consumer products industry cites cost reduction as its biggest logistics challenge. In *http://www.exel.com/exel/home/media/news/newsreleases/-pressreleasecostreduction.htm*, London, UK, 2002.

[8] H. Psaraftis. Dynamic vehicle routing: status and prospects. *Annals of Operations Research*, 61:143–164, 1995.

[9] M. W. P. Savelsbergh and M. Sol. The general pickup and delivery problem. *Transportation Science*, 29(1):17–29, 1995.

[10] P. M. Thompson and H. N. Psaraftis. Cyclic transfer algorithm for multivehicle routing and scheduling problems. *Operations Research*, 41(5), 1993.