Proceedings of The Fifth International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS-2011), Seoul, Korea, June 30 - July 2, 2011.

Agent-based Pickup and Delivery Planning: The Learnable Evolution Model Approach

Janusz Wojtusiak
Machine Learning and Inference Laboratory
George Mason University
Fairfax, VA, USA
jwojtusi@gmu.edu

Tobias Warden, Otthein Herzog
Center for Computing and Communication Technologies
University of Bremen
Bremen, Germany
{*warden, herzog*}*@tzi.de*

***Abstract*—The Dynamic Vehicle Routing Problem (DVRP) is an optimization problem in which agents deliver orders that are not known in advance to the routing. Partial solutions need to be adapted to continuously accommodate new orders within dynamically changing conditions. This research focuses on using a combination of multiagent-based autonomous control with non-Darwinian evolutionary optimization. In order to compile transport plans and render optimized decisions agents managing transport vehicles employ a guided evolutionary computation method, called the learnable evolution model (LEM). Implementation and experimental evaluation of the method is performed within the PlaSMA multiagent simulation platform.**

***Keywords*-Multiagent-based Simulation; Autonomous Logistics; Learnable Evolution Model; Evolutionary Computation;**

I. Introduction

Unprecedented growth of transportation and logistics networks in recent years calls for a shift in planning and control methods. Centralized planning approaches are gradually becoming less suited to handle the complexity of entire logistics networks. When centralized approaches are used, often entire global plans need to be re-computed, which may be very time consuming and in practice infeasible, particularly when frequent disruptions to plans occur. Research in the context of the Collaborative Research Centre (CRC) 637 "Autonomous Cooperating Logistic Processes – A Paradigm Shift and its Limitations" seeks to identify methods to cope with the highlighted challenges. It does so, by promoting the paradigm shift towards decentralized control of logistic processes [1]. In this paper, we focus on multiagent systems as one particular path to implement this control paradigm. The multiagent systems assume autonomy in decision-making of agents that are acting on behalf of logistic entities. As a consequence of the renunciation of a centralized planning strategy, the approach has been shown to be more robust in handling unexpected disturbances at the level of individual agents.

Our research is based on a hypothetical freight forwarding agency, which needs to handle a continuous dynamic flow of transport orders for freight containers. The underlying problem is the well-known Vehicle Routing Problem (VRP). A large number of studies have been conducted, which consider the various specializations of the problem. A good overview of VRP is provided, for instance, by Parragh et al. in [4]. However, a majority of the conducted studies consider static problems where all information is known in advance, i. e. orders are cached first and then planning is completed for a given time period. The Dynamic Vehicle Routing Problems (DVRP) constitute a new problem class, which has recently gained broader attention [2, pp. 207]. As time progresses, new orders are received and they need to be dynamically incorporated into the evolving schedule.

In the model considered here, the orders arrive on a regular basis at certain storage facilities geographically distributed over the territory of Germany. The model assumes that two subsequent transport orders need to be handled for each container introduced in the scenario – the transport of a full container without further distinction of containers with respect to enclosed contents, and return transport of the empty container after its unloading at its delivery facility (reverse transportation). In order to handle the transport orders, the freight forwarding agency is to employ exclusively its own fleet of transport vehicles. For simplicity, we assumed that the freight forwarder operates a homogeneous transport fleet of semi-trailer trucks, equipped to carry exactly one freight container. Each operated truck can only be in of two states: *empty* or *fully-loaded*. The trucks can be operated on a continuous 24/7 basis.

II. Autonomous Containers

In the presented multiagent-based autonomous control system, intelligent agents autonomously plan and determine the best course of action. In the investigated scenario of freight forwarding agency, the method works on two levels: 1) local valuation (prioritization) of pending transport orders, and 2) local selection of orders for handling deliveries by trucks in the transport fleet.

These two components fall into separate areas of responsibility. While the order valuation is part of order management, the operative order handling (which for the purposes of this research subsumes order selection) is part of transport management. This distinction is introduced in the agent-based implementation of the autonomous control system. Specifically, the system design comprises two primary classes of logistic agents which each handle one of the aforementioned management tasks. Each transport agent acts on behalf of a single truck and manages its intra-company order selection, acquisition, and subsequent operative order handling. These agents rely on an adequate valuations of pending transport orders, which constitutes the basis for the order selection. We considered a simple greedy approach as baseline and an evolutionary-based planning approach. The following sections introduce the mechanics of order valuation by the order management agents and the order selection and planning by the transport management agents.

A. Transport Orders

Greedy behavior of transport agents means that they select the most profitable orders. This may lead to orders with low *contractual order value* never be selected for transport. For the freight forwarding agency, such a system behavior evidently violates contract agreements with customers. Therefore, the forwarding agency requires an effective mechanism which is designed to ensure a timely handling of *all* pending orders. Although this research does not assume contractually fixed delivery times, the model is designed to allow all orders to be handled in a sensible global time window, e. g., 48 hours. The approach used here is based on an autonomous re-valuation of transport orders, which emanates from the original real valuation of the orders in terms of monetary value. The values whose calculation is introduced hereafter should be understood as a means of intra-organizational prioritization of orders. The transport management agents internally consider the order value for order selection which is calculated as follows:

$$value(order, t) = price(order) + prior^{+}(order, t) \quad (1)$$

$value(order, t)$ is computed as sum of a first component $price(order)$ which constitutes the initial order price agreed between customer and forwarder (cf. equation (2)) and a second component $prior^{+}(order, t)$ for the order revaluation based on the period of time the cargo has already been waiting to establish a transport contract (cf. equation (3)). The initial order price is computed as:

$$price(order) = d \cdot (c_0 + r \cdot distance(order)) \quad (2)$$

The constant c_0 denotes fixed costs for the operation of a semi-trailer truck, while the constant r denotes variable operating costs per kilometer. In the experiments described in Section IV, these parameters have been chosen as c_0=110,00 € and r=1,42 €. The length of a transport tour is measured in kilometers. Finally, in order to accommodate the two considered types of containers which are associated with transport orders (delivery and return of empty containers) the dampening factor d has been introduced and set to 1 for full and 0.25 for empty containers. All of the used values have been experimentally selected to match real world scenarios.

To increase the value of a container depending on the time it has been waiting, the function $prior^{+}(order, t)$ is used. This function is monotonically increasing with t, thus for containers which are kept waiting for a longer period of time the term $prior^{+}(order, t)$ at some point begins to dominate the initial order price. Thus, assuming a sufficiently large number of transport agents, each order will eventually be selected for transport as its value increases with time. While in general, the $prior^{+}(order, t)$ can be sophisticated and account for, amongst others, prediction of travel times, pickup and delivery time windows, and penalties, the basic version shown below has been used for the initial set of experiments.

$$prior^{+}(order, t) = w_c(order) \cdot t^{\alpha} \quad (3)$$

The constant weight w_c has been selected as follows: $w_c(order_{full})$=0.3 for full and $w_c(order_{empty}) = 0.05$ for empty containers. The power α is selected as 2.0. For example, the *value* of an order for the transport of a full container which was left waiting 10 hours exceeds the real order price by about 30,00 €. However, if the container is left waiting for 2 days (48 hours), its value already exceeds its original price by 691,20 €.

B. Selection of Transport Orders

The transport management agents, which have been briefly introduced in the preceding sections, are routinely faced with the challenge to autonomously render decisions that determine their respective operative transport planning. Although this report concentrates on a particular transport planning approach based on an evolutionary optimization method, a simple non-planning type of transport agent which effectively employs a greedy order selection strategy has been implemented as a baseline for the measurement of transport management performance. Sections II-B1 and II-B2 outline both order selection strategies. A common assumption, which is reasonable within the bounds of a single freight forwarding agency, is that the transport management has access to the entire momentary order situation which comprises the pending orders which are waiting for processing at the distinct storage facilities.

1) A Baseline Approach to Order Selection: Baseline transport management agents which have been implemented as part of the freight forwarding agency employ a *greedy order selection strategy*. Upon initialization, they scan the full set of pending transport orders to identify the most profitable order calculated by

$$\begin{aligned} bestOrder(t) &= \underset{\mathsf{Orders}(t)}{\arg\max}[\,(value(order,t) \\ &- (cost(start(order), dest(order)) \\ &+ cost(pos(truck), start(order))))\,] \end{aligned} \quad (4)$$

where $\mathsf{Orders}(t)$ is the set of all pending transport orders at time t. Once the best order has been identified, the resulting actions which need to be executed by the managed truck can be immediately derived. Two cases can be distinguished: (1) The most profitable order is associated with a transport relation whose starting point corresponds to the momentary position of the truck. In this case, no separate pickup tour is required and it holds that $cost_{Pick}(pos(truck), order) = 0.00$ €; and (2) a pickup tour is required in order to subsequently handle the selected transport order. The choice of the next transport order is repeated each time a truck reaches a storage facility, upon completion of its most recent delivery or a pickup tour. The latter case has significant potential negative implications with regard to the efficiency of the greedy-based operation, since agents are allowed to reconsider their previous delivery choice.

2) A Planning Approach to Order Selection: Planning transport management agents are routinely faced with the challenge to autonomously render decisions that determine their respective transport plans. These decisions thereby pertain to a choice of adequate action alternatives, the options being for each decision: (1) choosing a transport order whose pickup point is the currently considered storage facility, or (2) postponing that choice and relocate to another storage facility.

In essence, the transport agents need to choose which transport order to pick at a specific time and location. That choice is guided by the gains and costs of transporting the container associated with the order. Therefore, by choosing orders in an optimized way, an agent can maximize its financial balance. The behavior of the transport management agents is thus the result of series of constitutive decisions. This initial situation calls for provident planning in which a transport agent considers several steps ahead. The transport management agents seek an optimized pickup and delivery plan with a plan horizon of size n. Formally, such a plan is defined as:

$$\begin{aligned} plan^n &= (action_1, action_2, \ldots, action_n) \\ &\quad \text{where}\; action_i \in Deliveries \cup EmptyRides \end{aligned} \quad (5)$$

Deliveries refers to the set of possible delivery actions as determined by the pending transport orders which have previously been acquired by the transport forwarding agency. *EmptyRides* in contrast refers to the set of possible empty journeys between storage facilities. Thus, a transport plan as defined above can blend deliveries and empty drives where the latter can often be interpreted as pickup tours. The space of *valid* transport plans is specified by means of constraints:

Let $start : Deliveries \cup EmptyRides \rightarrow SF$ define a function which returns the source location of a particular plan step (i. e., in the case of proper orders, the pickup site). Let further $dest : Deliveries \cup EmptyRides \rightarrow SF$ define the complementary function which returns the target location of a plan step (for proper orders, the delivery site). In both cases, SF thereby constitutes the set of storage facilities in the given scenario.

(1) The first constraint that must hold in admissible transport plans is defined by:

$$\forall i = 1 \ldots n : start(action_i) \neq dest(action_i).$$

The rationale here is that both types of actions that can be carried out as plan steps, i. e., empty relocation from one storage facility to another and execution of a delivery, comprise a non-circular movement of the truck in question. Thus, a single plan step may neither consist of a round-trip nor of a rest or waiting period at a particular storage facility.

(2) Another constraint ensures that the tour specified by a valid plan is contiguous which means that short cycles are precluded by this constraint which thus acts as a sub-tour elimination constraint:

$$\forall i = 1 \ldots (n-1) : dest(action_i) = start(action_{i+1})$$

where $action_i$ denotes the tuple elements of a plan as defined in equation (5).

It is, however, possible for transport tours to revisit certain locations since loops are allowed by the formulation. For instance, let $Loc_a, Loc_b \in SF$, then

$$\begin{aligned} plan_3 &= (\,Del(Loc_a, Loc_b), Empty(Loc_b, Loc_a), \\ &\quad\; Del(Loc_a, Loc_b)\,) \end{aligned}$$

is an admissible plan with a first delivery from A to B, followed by an empty return trip and another delivery from A to B.

The value of a particular transport plan as defined above is thereby determined as follows, based on equation (1):

$$val(plan^n) = \sum_{i=0}^{n-1} val(act_{i+1}) \cdot (n-i)^{\alpha} \quad (6)$$

where $val(act) = val(order, t) - cost(start(act), dest(act))$ iff $act \in Deliveries$. For $act \in EmptyRides$, it holds that $val(order, t) = -cost(start(act), dest(act))$.

Equation (6) shows that the value of the complete transport plan is a weighted sum of the values of the respective plan steps $action_i$. The parameter is thereby used to determine a specific weighting scheme. For instance, if $\alpha = 0$, all plan steps are given equal weight in the calculation of the value for the complete plan. For the scope of the experiments, the value has been chosen as $\alpha = 2.0$ to give initial plan steps much higher weight.

3) Guided Evolutionary Approach to Planning: The learnable evolution model (LEM) is an evolutionary optimization method that employs machine learning to direct the evolutionary process [3]. Specifically, LEM creates general hypotheses indicating regions in the search space that likely contain optimal solutions and then instantiates these hypotheses to generate new candidate solutions. In order to apply machine learning, LEM creates two groups of individuals that are respectively high- and low-performing according to the fitness function being optimized. These individuals can be selected from the current population or a combination of current and past populations of individuals. The group of high-performing individuals is called H-Group and the group of low-performing individuals is called L-Group. Once the groups are selected, LEM applies concept learning to create a general hypothesis describing the H-Group in contrast to the L-Group. The hypotheses are then instantiated to create new candidate solutions. In the final step, a new population is assembled from old and new individuals, and the process is repeated until stopping criteria are met.

This research uses the third generation of LEM software, called LEM3. LEM3 dynamically selects one or more innovation methods to create new individuals. These methods are: *Learn & Instantiate*, the aforementioned main mechanism for creating new individuals in LEM3; *Adjust representation*, to change the discretization of numeric attributes; *Probe*, to apply traditional operators such as mutation and crossover; *Search locally*, to apply a user-defined local search method; and finally *Randomize*, to add to the current population a number of randomly created individuals, or restart the evolutionary process. LEM3 also has the ability to automatically adjust the representation space through constructive induction [7]. Theoretical and experimental work indicates that LEM is particularly suitable for optimization problems in which the fitness evaluation is costly. This is because of the trade-off between significantly shorter evolution length [3], [8], and more complex learning and instantiation when compared to simple operators used in evolutionary computation. Moreover, the use of machine learning to guide evolutionary computation extends the applicability of LEM. For example, due to the use of AQ21 as a learning engine in LEM3, it is able to handle optimization problems naturally described using different types of attributes (nominal, structured, ordinal, cyclic, interval, ratio, and compound) and background knowledge provided to the learning program [9].

4) Employing LEM3 for Transport Planning: LEM3 is a multipurpose system for evolutionary optimization that has been adapted to agent-based planing. The problem definition for the application of LEM3 planning incorporates:

1) the storage facility where the truck is located at the time of planning,
2) a complete list of storage facilities where transport orders may be pending, and
3) the size of the plan horizon.

Based on the problem definition, as given by a list of possible locations to visit, LEM3 searches for the best plan. It starts with an initial population of candidate plans, which is randomly generated. Due to the reduced problem definition these candidates constitute what has been referred to as *plan skeleton* rather than a fully-fledged transport plan. The concept can be formalized and related to the definition of proper transport plans in Equation (5), assuming a plan horizon of size n:

$$\begin{aligned} planSkel(plan^n) &= (\, dest(action_1), dest(action_2), \\ &\quad \ldots, dest(action_n)\,) \\ &\equiv (\, storage_1, storage_2, \ldots, storage_n\,) \end{aligned} \tag{7}$$

A plan skeleton is thus a n-tuple of storage facilities. However, with regard to the planning problem, only a subset of the set of all plan skeletons of size n is admissible in terms of compliance with the following constraints:

1) $\forall i = 1 \ldots (n-1) : storage_i \neq storage_{i+1}$ since all actions in proper plans involve a relocation of the truck between two distinct storage facilities, either via an empty ride or a proper delivery of a container.
2) $pos(truck) \neq storage_1$ where $pos(truck), storage_1 \in SF$. In particular, the storage facility where the truck is located at planning time must not be identical with the first storage facility in the plan skeleton.

For the application in plan optimization, the aforementioned constraints have been implemented as part of the problem-specific LEM3 integration. As a consequence, after the creation of new candidate solutions, LEM3 is enabled to detect the number of constraint violations and filter inadmissible solutions immediately.

Since these plan skeletons without additional processing do not describe directly any particular plan which can be evaluated, an unequivocal conversion into a proper transport plan needs to be established. This conversion is possible due to two assumptions. First, transport plans always have as origin the current location of the associated truck which corresponds to a storage facility ($pos(truck)$). In addition, as the transport agent seeks to maximize its financial balance for each plan step, it is rational to choose the most profitable action alternative using a function $bestAct : SF \times SF \rightarrow Deliveries \cup EmptyRides$ which accepts transport end points as input and returns either most profitable real transport order, or, as a fallback if no orders with the specified transport endpoints currently exists in the system, the empty drive order. Based on a plan skeleton, the corresponding plan is:

$$\begin{aligned} plan^n \quad = \quad & (\, bestAct(pos(truck), storage_1), \\ & \ldots, bestAct(storage_{n-1}, storage_n)\,) \end{aligned} \tag{8}$$

The transformation from plan skeletons which constitute the plan suggestions created iteratively by LEM3 into proper candidate plans from the point of view of the planning agent is a mandatory prerequisite in order to apply the domain-specific weighting function which is used to evaluate candidates and thus drives LEM's search in the space of possible plans. The approach which is currently applied with the integration of LEM3 is basically to out-source the weighting function from the library to the planning agent as user of the library which is also equipped with the required domain-specific knowledge to execute the candidate valuation.

In order to directly calculate the value of a suggested plan skeleton, the following equation (9) can be applied:

$$\begin{aligned} & value(planSkeleton^n) \\ &= balance(bestAct(pos(truck), storage_1)) \cdot n^\alpha \\ &+ \sum_{i=1}^{n-1} [balance(bestAct(storage_i, storage_{i+1})) \cdot (n-1)^\alpha] \end{aligned} \tag{9}$$

The function $balance : SF \times SF \rightarrow$ Euro takes the current order value (which for all orders that have been waiting for some time is higher than the initial order price, cf. Section II-A) and subtracts operation costs for the execution of these orders. Once plans have been evaluated, LEM3 checks stopping criteria (reached desired value of plan or the maximum length of evolution is reached).

III. Multiagent-based Implementation

The freight forwarding agency described in the introduction has been implemented as a multiagent system (MAS) for an evaluation of emergent system behaviour in the multiagent-based simulation system PlaSMA[1] [5]. The realized MAS involves three distinct agent types, namely order management agents which implement the order valuation strategy proposed in Section II-A, transport management agents which need to compile transport plans autonomously, following one of the strategies from Section II-B. Finally, there is an order information service agent (OIS) which maintains a company wide-database of pending transport orders. The short-lived order agents which oversee the regular and subsequent reverse transport of one particular order interact with the OIS to publish orders and update their prioritization over time. The transport agents query the OIS to acquire the knowledge about the momentary oder situation of the forwarding company which is the based for an informed compilation of the local transport schedule. Details on the specification of the aforementioned agent types and technical integration of LEM3 with the transport management agents are discussed in [6]. For this article, we proceed to the discussion of experiments and findings.

IV. Simulation Experiments and Evaluation

To thoroughly evaluate the performance of transport management agents employing LEM3 for route planning relative to the baseline, both strategies were tested with the freight forwarder MAS within PlaSMA simulations. A realistic, low-detail traffic network represented as a directed graph has been used as a common basis for the simulation experiments. The network covers the area of Germany. It contains 359 nodes and 1.044 edges. The nodes comprise besides pure traffic junctions and path subdivisions (309 nodes) the major cities of Germany (50 nodes). The edges constitute transport routes. These represent a significant part of the German motorway network (750 edges) and, to a lesser degree in order to connect motorway sections or cities in the motorway network, federal roads (152 edges) and inner-city roads (28 edges). In the presented work, each city hosts exactly one facility whose relevant modeling parameters comprise its overall storage capacity, uniformly specified as 100 container units, and simulation-specific parameters to control the external intake of new orders (for details, cf. [6, p. 23]). Ten out of 42 storage facilities were configured with an external intake of transport orders. These have been chosen due to their geographic location such that a reasonable distribution of orders could be established. The experiments consisted of two experiment series which differ in the amount of orders created at high-intake storage facilities, i. e., creation of orders every six hours (experiment series 1) or every five hours (experiment series 2)[2]. Due to space constraints, this article reports only findings from the latter series. The experiments feature a homogeneous fleet of 16 trucks which can transport a single container at a time. Three configurations of the transport management agents were employed: first, a configuration where all trucks are managed

[1] PlaSMA web site: http://plasma.informatik.uni-bremen.de

[2] In both cases, orders are created every 12 hours for low-intake facilities. Details are provided again in [6, p. 23]

by baseline transport agents; second, a configuration featuring LEM-enabled transport agents; and a third 'competitive' scenario, where eight trucks respectively are managed by agents of either type. The last scenario was chosen specifically to analyze the emergent effects of intermixing transport planning strategies of different complexity. The LEM3 system employed by the planning transport agent has been executed with default parameter settings (cf. [6, p. 23]).

A. *Experimental Results*

The presented evaluation is based on performance indicators (PIs), collected by the transport management agents (20 PIs), the order management agents (7 PIs) and the OIS agent (3 PIs).

Fleet-Level Financial Analysis When considering the fleet balance (Fleet Bal. ($\mu_s \pm \sigma_s$) in Table I) first, the direct comparison of the performance of the heterogeneous transport fleets shows a significantly higher mean overall revenue when using the planning approach, i. e., 866.038,94 € vs. only 708,409,97 € for the baseline approach. The standard deviation – even though still residing at $\sim$ 3% is notably higher for the baseline approach (i. e., 10.051,49 € for the planning fleet and more than double that value, namely 21.565,53 € for the baseline fleet). The data for costs and gains which is also provided in Table I shows that this increase in the standard deviation is caused by more variation in the revenue generated by handling transport orders. Therefore, the data seems to suggest that the planning approach exhibits better stability in a more demanding market situation. It should be noted further, that in the considered scenario the planning approach generates slightly higher costs than the baseline approach which are, however, compensated by the higher revenues.

In the competitive scenario where both the baseline and the planning approach were employed to equal proportion, the mean overall revenue of 846.523,42 € comes very close to the 866.038,94 € for the homogeneous planning fleet. Compared to the results measured for a scenario with a lesser order generated throughout the simulation as shown in [6, Section 4.4.1, pp. 24], the insertion of planning transport management agents into an otherwise non-planning baseline fleet yields even more notable positive results in a scenario with a comparatively strong inflow of transport orders. However, the effect of an increased standard deviation compared to a fleet operated by planning transport agents alone which has been observed before is evident once more in the data presented in Table I. The transport management agents which employ LEM-based planning on average significantly outperform their less provident counterparts with a mean revenue of 61.201,87 € ± 3.856,59 € compared to 44.613,56 € ± 5.062,45 €. More interesting than those raw numbers is a comparison of the per agent performance between the homogeneous and the mixed setting. Here, the data shows that the baseline agents retain their performance values with respect to mean revenue while the standard deviation rises, from 4.347,73 € up to 5.062,45 €. The planning agents, however, manage to thrive in a situation with less peer competition and the insertion of baseline agents to compete against. This is documented by the fact that the per agent revenues rise significantly from a mean of 54.127,43 € ± 4.415,25 € up to 61.201,87 € ± 3.856,59 €.

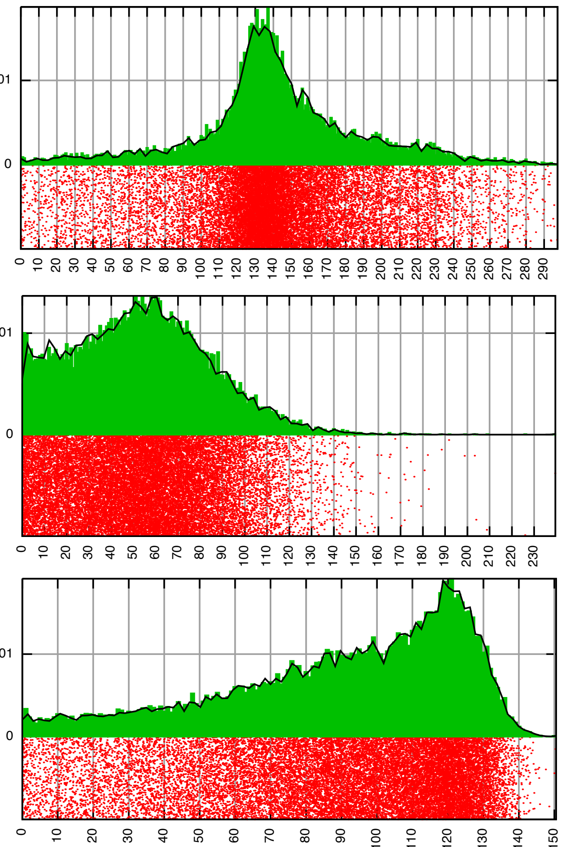

Figure 1: Distribution of container pickup times (for reverse transports) for 10 simulations: Baseline (top), planning, and mixed fleet.

Pickup Times The data in Table II shows that the LEM-based planning approach leads to a significant reduction of waiting times both for the pickup of containers for regular and reverse transports of empty containers.

For the case of reverse transportation, the effect of combining both approaches to equal fractions in the mixed transport fleet can be succinctly described as follows: While retaining the single peak at about 120 hours also found in the baseline case, the majority of the slope from that case is cleared away, the respective

Table I: Financial results for different transport management approaches, measured in €/60 days ($\mu_s \pm \sigma_s$).

€/60d	Mixed Fleet	LEM Fleet	Baseline Fleet
LEM3 Route Planning			
Σ Costs	$606.931, 42 \pm 415, 90$	$1.213.175, 1 \pm 930, 35$	–
Σ Gains	$1.096.546, 4 \pm 15.878, 41$	$2.079.214, 0 \pm 10.981, 84$	–
Σ Balance	$489.614, 98 \pm 15.462, 51$	$866.038, 94 \pm 10.051, 49$	–
p.A. Balance	$61.201, 87 \pm 3.856, 59$	$54.127, 43 \pm 4.415, 25$	–
Baseline Order Selection			
Σ Costs	$609.229, 46 \pm 920, 48$	–	$1.212.536, 1 \pm 885, 16$
Σ Gains	$966.137, 90 \pm 10.248, 10$	–	$1.920.946, 1 \pm 22.450, 69$
Σ Balance	$356.908, 44 \pm 9.327, 62$	–	$708.409, 97 \pm 21.565, 53$
p.A. Balance	$44.613, 56 \pm 5.062, 45$	–	$44.275, 62 \pm 4.347, 73$
Fleet Bal.	$846.523, 42 \pm 24.790, 13$	$866.038, 94 \pm 10.051, 49$	$708.409, 97 \pm 21.565, 53$

Table II: Waiting times until a container is picked up.

Unit: *hours*	Mixed Fleet
Full ($\mu_s \pm \sigma_s$)	$12, 597 \pm 11, 899$
Empty ($\mu_s \pm \sigma_s$)	$90, 029 \pm 33, 761$
	LEM Fleet
Full ($\mu_s \pm \sigma_s$)	$7, 926 \pm 8, 706$
Empty ($\mu_s \pm \sigma_s$)	$53, 832 \pm 30, 718$
	Baseline Fleet
Full ($\mu_s \pm \sigma_s$)	$40, 820 \pm 24, 855$
Empty ($\mu_s \pm \sigma_s$)	$141, 564 \pm 46, 675$

weight of the distribution being transferred to the left slope (cf. Figure 1). As already noted in the analysis of the first set of experiments, the mixture of approaches also helps to suppress the undesirable outlier cases which were once again observed when employing only planning transport management agents.

For the inventories of the storage facilities in the planning case, the experiment data shows that low inventory levels can be maintained throughout the simulation runs. Specifically, stock levels max out at 6–8 containers at a time. Taking also into account the findings from the preceding section, it can be concluded that the applied load on the forwarding agency was too much to be handled effectively when employing the baseline approach to transport planning. With these results in mind, it is interesting to note that the mixed transport fleet in which baseline and planning transport management agents work side by side leads to a stable system where both the stock levels at the individual storage facilities and the total amount of pending transport orders tune in on a certain level. However, a plot of the stock levels [6, p. 35] suggests significantly increased peak amplitudes compared to the planning case. In addition, although the total amount of pending transport orders levels off after the initial phase of the simulation it does so at higher levels, i. e., at ~ 20 regular transport orders (compared to ~ 15 in the planning case) and ~ 140 reverse transport orders (compared to ~ 90).

Concluding the analysis with regard to pending orders and inventory levels, the superiority of the planning approach to operative transport planning could be affirmed. While this result for itself bears little surprise, with the presented experiments it could be shown where the baseline approach used in the experiments has its limits and, in particular, how these limits may be extruded by mixing in planning transport management agents into a baseline transport fleet.

Transport Operations The data in Table III shows that the planning approach leads to a significant increase in the overall amount of successfully operated container transports ($4.301, 30 \pm 5, 88$) compared to the baseline approach ($3.824, 70 \pm 80, 48$). Even though a higher number of both regular and reverse transports are handled by the planning fleet, the data also suggests a tendency of this approach towards a reduced fraction of regular transports (51, 0 % compared to 54, 2 %).

In addition to the superiority with regard to the raw amount of handled transports and an slightly increased total length of delivery tours, the fraction of container transports on all truck operations is in the average notably higher for the planning transport fleet (65, 5 %) than for the baseline (57, 4 %). When comparing these results with their counterparts measured in experiments with a lower order inflow (cf. [6, Section 4.4.1, pp. 24]), the data suggests that given a higher amount of orders in the system, the planning fleet can substantially improve its truck utilization. To be more precise, the fraction of 47, 2 % of transports on all truck operations, when running the same experiment presented here with high intake storage facilities receiving new orders only every 6 and not every 5 hours, is brought up to 65, 7 %. The data raises the assumption, to be affirmed or disproved by further experiments, that the planning fleet performs particularly well in scenarios with a high order inflow. However, the data acquired from the experiments so far also shows that the goals of high capacity utilization, increasing the freight forwarders profitability, and low pickup times, improving the quality of offered services, are related such that one is faced with a multi-criterial optimization problem.

Shifting the focus from the analysis of the homogeneous settings towards the mixed case with an equal amount of both agent types, the first thing to notice in the data presented in Table III is the significant difference in overall operated container transports for both fleets. Both with

Table III: Overview of transport operations for trucks managed by different transport agents.

	Mixed Fleet, pt. a (8 Trucks)	LEM Fleet (16 Trucks)
LEM3 Transport Operations		
Σ Deliv ($\mu_s \pm \sigma_s$)	$2.478,90 \pm 17,06$	$4.301,30 \pm 5,88$
Σ Deliv $_{\text{full}}$ ($\mu_s \pm \sigma_s$)	$1.231,20 \pm 12,93$	$2.194,50 \pm 1,92$
Σ Deliv $_{\text{empty}}$ ($\mu_s \pm \sigma_s$)	$1.247,70 \pm 17,60$	$2.106,80 \pm 5,29$
Σ Fraction Full (μ_s)	$0,497$	$0,510$
Σ Fraction: All (μ_s)	$0,808$	$0,655$
Σ Length Del. ($\mu_s \pm \sigma_s$)	$1.074.057 \pm 8.437$	$1.974.968 \pm 12.396$
	Mixed Fleet, pt. b (8 Trucks)	Baseline Fleet (16 Trucks)
Baseline Transport Operations		
Σ Deliv ($\mu_s \pm \sigma_s$)	$1.740,70 \pm 12,80$	$3.824,70 \pm 80,48$
Σ Deliv $_{\text{full}}$ ($\mu_s \pm \sigma_s$)	$954,80 \pm 12,54$	$2.074,20 \pm 22,40$
Σ Deliv $_{\text{empty}}$ ($\mu_s \pm \sigma_s$)	$785,90 \pm 15,18$	$1.750,50 \pm 58,53$
Σ Fraction: Full (μ_s)	$0,549$	$0,542$
Σ Fraction: All (μ_s)	$0,562$	$0,574$
Σ Length Del. ($\mu_s \pm \sigma_s$)	868.476 ± 6.993	$1.759.706 \pm 30.745$

regard to regular and reverse container transports, the planning fleet clearly outperformed the baseline fleet, resulting in $2.478 \pm 17,06$ deliveries for the former vs. only $1.740,70 \pm 12,80$ deliveries for the baseline fleet. More interesting than the plain amounts of deliveries are the results for the respective fractions of regular transports on all transport operations and of transports on all truck operations. If only considering data from the mixed scenario, it shows that the mean fraction of deliveries on all truck operations is higher for the planning sub fleet (80,8%) than for the baseline sub fleet (56,2%). This suggests that the former sub fleet actually profits from the competitive setting at the expense of the baseline sub fleet. A related result is that the planning fleet steals a significant amount of individually less profitable reverse transports from the baseline fleet. However, as documented in Table I, this does not have a negative effect on the revenue generated by the planning fleet. This documents the positive effect of planning in a competitive setting with regard to truck utilization and profitability.

V. Conclusion

The paper presented a methodology for decentralized order pickup and delivery planning by autonomous agents, which use the learnable evolution model to create optimized sequences of container deliveries. In the presented model each order, be it a regular or reverse transport, is assigned a intra-organizational prioritization which is updated by dedicated order management agent as a function of pickup waiting time. Experimental results indicate that the method performs better when compared to a baseline approach, in which each agent selects the best container at its given location. This result was obtained when greedy and LEM agents were simulated separately, and in a mixed scenario that combined both types of agents. All experiments were performed using the PlaSMA multiagent system [5].

Acknowledgements

This research has been supported by the German Research Foundation (DFG) within the Collaborative Research Centre 637 *Autonomous Cooperating Logistic Processes – A Paradigm Shift and its Limitations* at the Universität Bremen, Germany. Development of LEM3 has been supported in part by the National Science Foundation grants. Current research on the learnable evolution model is supported by the National Institute of Standards and Technology grant.

References

[1] M. Hülsmann and K. Windt, editors. *Understanding Autonomous Cooperation and Control in Logistics. The Impact on Management, Information and Communication and Material Flow.* Springer, 2007.

[2] A. Larsen, O. Madsen, et al. Recent Developments in Dynamic Vehicle Routing Systems. In *The Vehicle Routing Problem: Latest Advances and New Challenges*, volume 43, pages 199–218. Springer US, 2008.

[3] R.S. Michalski. Learnable Evolution Model: Evolutionary Processes Guided by Machine Learning. *Machine Learning*, 38:9–40, 2000.

[4] S. Parragh, K. Doerner, and R. Hartl. A Survey on Pickup and Delivery Problems. *Journal für Betriebswirtschaft*, 58:81–117, 2008.

[5] T. Warden, R. Porzel, J. D. Gehrke, O. Herzog, H. Langer, and R. Malaka. Towards Ontology-based Multiagent Simulations: The PlaSMA Approach. In *24th European Conf. on Modelling and Simulation*, pages 50–56, 2010.

[6] T. Warden and J. Wojtusiak. Learnable Evolutionary Optimization in Autonomous Pickup & Delivery Planning: A Scenario, System Architecture and Initial Results. TZI Technical Report 55, University of Bremen, 2010.

[7] J. Wojtusiak. Data-Driven Constructive Induction in the Learnable Evolution Model. In *Proceedings of the 16th International Conference on Intelligent Information Systems*, Zakopane, Poland, June 16–18 2008.

[8] J. Wojtusiak and R.S. Michalski. The LEM3 Implementation of Learnable Evolution Model and Its Testing on Complex Function Optimization Problems. In *Proceedings of Genetic and Evolutionary Computation Conference (GECCO 2006)*, Seattle, WA, Juli 8–12 2006.

[9] Janusz Wojtusiak. The LEM3 System for Multitype Evolutionary Optimization. *Computing and Informatics (28)*, pages 225–236, 2009.