Agent-based Planning and Simulation of Combined Rail/Road Transport

Luca Maria Gambardella, Andrea E. Rizzoli
IDSIA, Galleria 2, CH-6928 Manno, Switzerland

Petra Funk
DFKI, Stuhlsatzenhausweg 3, D-66123, Saabrücken, Germany

Abstract

A simulation model of the flow of Intermodal Terminal Units (ITUs) among inland intermodal terminals is presented. The intermodal terminals are inter-connected by rail corridors. Each terminal serves a user catchment area via a road network. The terminal is modelled as a set of platforms, which are served by a number of gantry cranes and front lifters. Given the schedule of train connections among the terminals, an agent-based system, the Intermodal Transport Planner (ITP) books ITUs on trains and assigns trucks to deliver them to the source terminal and to pick them up in the destination terminal. The terminal and rail corridor simulation software has been implemented as a discrete-event simulation model, using MODSIM III as development tool. The ITP has been implemented on the basis of the TELETRUCK system. This research has been developed within the PLATFORM project, funded by the Directorate General VII of the European Community.

Keywords: multi-agent simulation, intermodal transport planning, intermodal terminal simulation

1 Introduction

Nowadays many intermodal terminals are still managed without a pervasive support of information technologies: the terminal management highly relies on well-assessed policies, typical of each terminal, which have been defined by the managers on the basis of their experience. In most cases these policies are satisfactory since the terminals have sufficient resources in terms of tracks, equipment, human resources and they can support the current flows of freight. On the other hand, the growth of freight transport shows a rapidly increasing trend in the short and medium terms, which cannot be met by the current infrastructures and management tools. Computer based simulation can provide the decision-makers with the help they need in creating the strategies for development.

The pressure of freight traffic on European roads is pushing the European Community to invest and promote intermodal transport as a viable alternative to long-haul road transport [1]. The PLATFORM project is one of the outcomes of this policy and we expect that this and other demonstrative projects will show the terminal operators ways to invest to improve the efficiency of their management procedures, thus enhancing their competitiveness with respect to road-only freight.

One of the aims of the PLATFORM project was the implementation of a simulation environment for the assessment of impacts produced by the adoption of different technologies and management policies to enhance terminal performances. To achieve this objective, the project needed to encompass all the phases of an intermodal transport of an Intermodal Transport Unit (ITU), a requirement for the comparison of the performance of intermodality against road-only based transport.

An intermodal terminal can be regarded as a node in a network that models the connectivity of the origins and destinations in the supply chain. If we look at the performance of this network, we are interested in understanding if it is possible to increase the throughput of the nodes, that is, of the terminals. Whereas the rail networks can sustain a marginal increase in traffic, an improvement in the terminals throughput might reduce the percentage of long-haul transports on the road. Because of this observation, we model the complete logistic chain in a complex network of intermodal terminals in order to understand how intermodal transport can be put in competition with road-only based transport.

2 The PLATFORM architecture

The PLATFORM architecture consists of two subsystems: the intermodal transport planner (ITP) that manages the planning of the whole intermodal transport chain from origin to destination for an ITU; the simulation system (composed by the road simulation, rail simulation, and terminal simulation modules) that models and simulates the ITU transport process, both assessing the feasibility of the plans generated by the ITP and evaluates the performances of intermodal terminals, thanks to a detailed description of the intra-terminal processes.

The ITP plans the a whole intermodal transport task (ITT) for an ITU thanks to:

- Intermodal Planning and Execution Units (IPnEU) – for planning the whole ITT of an ITU. They split the ITT into its three main parts, the initial and final leg on the road and the main leg by train. They contact the specialised agents for planning, booking, reservation of these parts.

- Forwarding Agents – for planning and booking the ITT of the ITU by truck. These agents are responsible for the planning of delivery ITUs to and their pick-up from terminals. Each forwarder is modelled by such a forwarding agent. A broker agent co-ordinates the planning of the forwarding agents of the area around each terminal

- Booking Agent – for booking the ITT of the ITU by train. This agent checks for availability of places on scheduled trains, checking which bookings are possible. The booking agent then chooses the best one and makes the reservation.

The simulation system performs the execution of ITTs including internal terminal operations to evaluate their feasibility and their performances. It is composed of:

- Road Simulation – simulates the transport of the ITU by truck, as delegated to forwarders. It simulates the flow of incoming and departing trucks at each terminal in the corridor.

- Terminal Simulation – simulates the loading, unloading of ITUs from trucks and trains as well as storing of ITUs in the intermodal terminal. Terminal equipment, loading platforms, yard areas and gate procedures are simulated, in order to demonstrate functionality of the terminal procedures and potential for improvement.

- Train Simulation – simulates the flow of trains within the chosen rail corridor. According to the train time-tables, the flow of trains from and to the terminals is simulated, focussing especially the train flow within the chosen corridor.

A graphical user interface (not described in this paper) allows customising the simulation, to select the important parameters, views and maps, and to inspect of the input and output data of the simulation.

In the following, first we introduce the ITP (Section 3), then we explain the different components of the simulation system, the road network simulation, the rail simulation, and the terminal simulation (Section 4 and Section 5). Finally, we present the system as a whole, detailing the synchronisation mechanism between road and terminal simulations (Section 7).

3 The intermodal transport planner

The planning of intermodal transports is performed by means of an agent-based model of the intermodal transport chain. The transport service operators are represented by individual, intelligent software agents. Each of these agents is equipped with task- and domain-specific planning and scheduling abilities as well as models of their local resources in terms of time, load capacities etc. This approach allows for the distribution of the tasks to be solved in the processing of intermodal transport orders.

Multi-agent systems are highly suitable for distributed problem solving as they offer the possibility to divide the main task into small subtasks. Moreover such handy subtasks may contain overlapping goals, which are addressed by agents through their interactive abilities, i.e. they negotiate about their resources and abilities and elaborate co-operative solutions and plans. A more detailed introduction to multi-agent systems can be found for example in [2], [3] or [4]. For the intermodal transport chain in the PLATFORM tool, we have developed a multi-agent system, which maps the whole transport chain from the origin to the destination loading area of the forwarder. The forwarders are connected by road to intermodal terminals, which in turn are connected through the railway network. In the multiagent system each transport operator is represented as an autonomous agent. An ITT is planned and negotiated in the agent world, as well as co-operatively executed.

The forwarding agents, at the beginning and the end of the chain, are instances of the TELETRUCK system [5], [6]. The TELETRUCK system was designed as an agent-based forwarding system, able to manage the business processes of forwarding companies. The TELETRUCK agent society is implemented as an holonic agent system. An holonic agent or holon is an agent that is composed of sub-agents working together in order to pursue a common goal. The users or the other members of the agent society can interact with a holon as if it were a single agent. This allows to model several level of abstraction in a convenient way. In a holon one agent is distinguished as the head of the holon. The head co-ordinates the resource allocation within the holon and controls the communication with the rest of the agent society. The head can be equipped with the ability to plan for the sub-agents. These agents have their own plans, goals, and communication facilities in order to provide their resources for the transportation plans according to their role in the society. They can merge together with a Plan'n'Execute Unit (PnEU) and form a holon which represents a complete vehicle, as shown in Figure 1.

Agents also represent intermodal terminals. They consist of a terminal agent, which is the head of the agent society and a booking agent managing the booking requests for the trains handled in the terminal. This agent comprises the heart of the terminal services for the negotiation and planning of ITTs. and represent the commercial departments of intermodal terminals. Such a system is comparable to reservation systems for passenger transport. Booking and reservation systems, even though not so common in todays freight traffic, will in the future provide essential support for the smooth management of fast loading devices within intermodal terminals such as the Automated Loading System, the Krupp Fast Handling Device, or the Daimler KombiLifter [7].

In order to be able to plan ITTs we designed a third agent type, the Intermodal Plan'n'Execute Unit (IPnEU) which extends the TELETRUCKs PnEU. The IPnEU is able communicate with forwarding agents and terminal agents to close the transport chain. The IPnEU acts like a virtual cargo carrier imposing highly specific requirements on the cargo type. This agent has twofold abilities: during the planning phases of an ITT it negotiates with the different transport services in order to establish a co-operative transport plan. Later on, during plan execution the agent accompanies the cargo, which is moved through the physical transport network in the respective software network. Thus, this agent provides for advanced planning and online tracking of ITTs.

3.1 An agent for intermodal planning and execution

Intermodal plans are usually generated as the result of the interaction between the forwarders and the terminal companies. We represent the knowledge in intermodal transport planning for each transport operator, by encapsulating it into the IPnEU which is an agent associated with an intermodal order. This means the IPnEU plans and executes the plans for all the goods to be delivered within the same order and not only for a single ITU, as it does the PnEU in the standard TELETRUCK system. If an order contains more than one ITU, it may be splitted over several vehi-

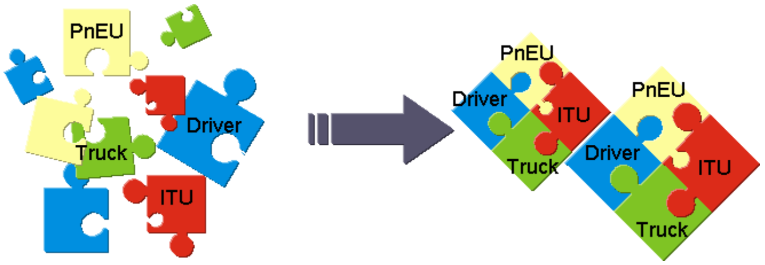

Figure 1: Vehicle agents and PnEU (left) forming a holonic vehicle structure (right).

cles. Yet only one IPnEU is supervising the transport execution. The IPnEU plans and negotiates the intermodal transport of the ITUs it represents and then monitors the execution of the plan, eventually migrating to other software systems, while the ITUs are in transit.

Thanks to its planning capability, the IPnEU has smooth access to all the transport operators in the transport chain to perform the negotiation and planning of an ITT. Moreover, the IPnEU is a mobile agent, which accompanies its cargo in the agent world, while the goods are shipped in the physical world. Mobility in the physical world is mirrored by the ability to migrate through a software network, following the execution of an intermodal order from origin to destination. Figure 2 illustrates this: the intelligent agent on the top is the IPnEU during the planning and negotiation phase; the walking agent is the vehicle holon during the execution and monitoring phase. The small puzzle in the hands of the walking agent indicates the holonic agent society. The black piece stands for the IPnEU's participation in it. The phases are modelled differently, though we are able to mix them freely and thus provide for emergency replanning during execution or due to new incoming orders, allowing for their dynamic insertion into existing plans.

In the next sections we show how the announcement of an ITT triggers the inactive IPnEUs (the active IPnEUs are by definition busy with either planning or executing an order) to enter the planning and negotiating phase. The planning phase itself is divided into a negotiation phase (Section (3.2)) with preliminary commitment to the services requested (road based transport, rail based transport and terminal services) and a final commitment phase (Section 3.3), where the information gained during the negotiation will be used in order to place bookings on trains and to route vehicles. The preliminary commitments serve two purposes: reserve resources during planning and negotiation and provide a decision and planning basis for service providers within the intermodal transport chain.

3.2 Communication and co-operation: negotiating intermodal orders

Customers and transport agents communicate to negotiate the contract for the execution of intermodal orders. The customer at the origin requests an ITT to the forwarder of her choice. She may announce the order to several transport operators in order to receive and select the most competitive offer. Figure 3 shows the details of the negotiation and planning phase.

The forwarder recognises that the order requires an intermodal transport and activates an IPnEU to provide for intermodal planning. The IPnEU splits the order into three parts: the rail-based main leg order which constrains the orders for the initial and final road-based legs. The main leg order, on rail, is passed to the booking agent of one or more terminals, who then engage in planning it. The result of this activity, the main leg plan is communicated back to the IPnEU. If the IPnEU has contacted several terminals, it sends one of them a preliminary commitment and adjusts the orders of the initial and final leg to the chosen plan for the main leg (latest arrival time at the terminal gate, earliest pick up time at the destination terminal). Other main leg plans are rejected. Planning of the initial and final leg can be done concurrently. In Figure 3 this is indicated

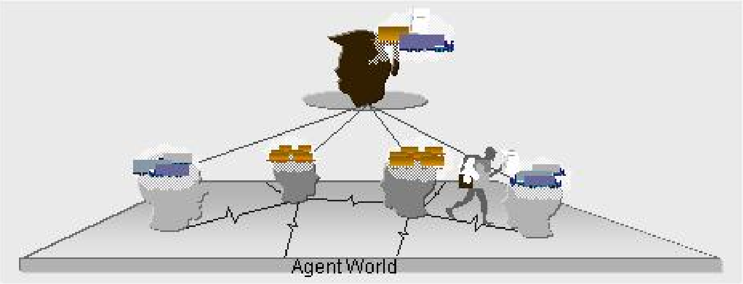

Figure 2: The IPnEU planning and migrating.

by dashed arrows. Planning the initial and final legs involves the usual TELETRUCK planning and scheduling activity, which results in a holon for every vehicle. With the information on the whole transport, the IPnEU can then tell the forwarder the pick up time at the customers depot (possibly a time window) and the delivery time (possibly a time window) at the final destination. The Intermodal Planning and Negotiation Protocol is an application-specific extension and nesting of several classical Contract Net Protocols [4].

3.3 Results of the negotiation and planning phase

The negotiation and planning phase generates an intermodal transport plan. Such a plan is a composition of plans for the different transport legs. The intermodal plan is composed of two road-based transport plans and one rail-based plan. The road-based plans implement the TELETRUCK approach, that is, for each vehicle an holonic structure is generated. Each structure is dominated by a PnEU. The IPnEU participates to each one of these holonic structures. This is possible, because an agent can be part of several holons at the same time.

As in the TELETRUCK implementation, whenever the system plans a road-based transport, the agents representing the involved physical components form a vehicle holon that is headed by a PnEU. If an ITU that is part of an intermodal transportation order is transported, not only the agent that represents that ITU but also the IPnEU that represents the transportation order are incorporated in the vehicle holon. Precisely, the ITU agent and its IPnEU form a holon that is headed by the IPnEU and that represents the ITU.

For the rail-based transport, the train that transports several ITUs can be viewed as a holonic structure that consists of the ITUs, the wagons, and the engine. We do not model the wagons explicitly but represent the whole train by the locomotive agent. The train holon is composed of the IPnEU agents that represent the intermodal orders, the agents representing the ITUs in transit, and the locomotive agent which is the head of the train holon.

As a consequence of the domains structure, holons overlap. The agents form holons at the time of planning and can be members in several holons. The ITU agents are part of the holon that represents the intermodal order, they belong to two vehicle holons for the initial and the final legs, and to a train holon for the main leg. The IPnEU agents exhibit even stronger omnipresence: they participate in the vehicle holons for initial and the final legs of each ITU in the order and they are part of the train holons that carry at least one of the respective ITUs. This agent society is visualised in Figure 4. The picture shows on both sides road-based vehicle holons that consist of a PnEU, a driver agent, a truck agent, and a conjunction of ITU and IPnEU agent. In the middle there is a rail-based train holon that contains the locomotive agent, both ITUs and the IPnEU. The IPnEU is part of all holons involved and supervises the whole transportation chain.

4 The road network simulator

While the planning phase can be compactly described with the protocol illustrated in Figure 3,

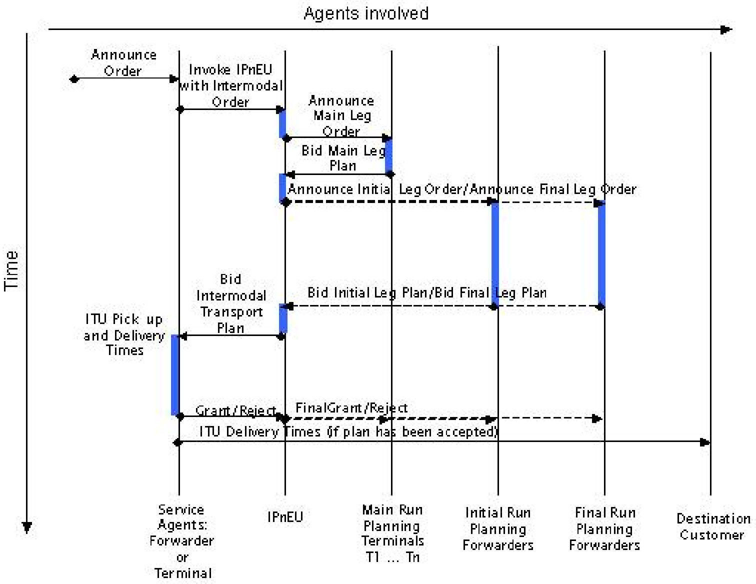
Agents involved
Time
Announce Order
Invoke IPnEU with Intermodal Order
Announce Main Leg Order
Bid Main Leg Plan
Announce Initial Leg Order/Announce Final Leg Order
Bid Intermodal Transport Plan
Bid Initial Leg Plan/Bid Final Leg Plan
ITU Pick up and Delivery Times
Grant/Reject
FinalGrant/Reject
ITU Delivery Times (if plan has been accepted)
Service Agents: Forwarder or Terminal
IPnEU
Main Run Planning Terminals T1 ... Tn
Initial Run Planning Forwarders
Final Run Planning Forwarders
Destination Customer

Figure 3: The intermodal planning and negotiation protocol.

the execution requires some more elaborated methods and competencies. The execution itself can also be described in a protocol-like diagram, where messages about the result of the execution are communicated (Figure 5). Within this figure the grey arrows indicate the transport control or supervision by the IPnEU.

The IPnEU splits the order at hand into a main leg to be on train and an initial and final leg to be executed by trucks. The intervals are based on estimated duration of the different legs. For the main leg it consults the booking agents, while the relative road network area brokers are contacted for the initial and final legs

When the booking agent and the road network area brokers return the booking for the main leg and the plans for the initial and final leg, respectively, the IPnEU must integrate the three plans into the intermodal transport plan for the current order. If the integration fails, another iteration with modified time windows takes place.

For the region around a terminal, a road area network broker agent is responsible for the organisation of the initial or final legs. This agent is implemented as an autonomous copy of a TELETRUCK system, but without subagents representing the transport units. It knows about the forwarders in the region and can contact their forwarding agent, in order to announce the transport task to them asking for their bids for that task. Modified negotiation and communication facilities had to replace therefore the original ones which were used normally for the distribution of orders to the transport subagents.

The forwarding agents of the local forwarders are again implemented as autonomous copies of the TELETRUCK system, of course with their usual subagents for the transport units and their components. In extension of the normal TELETRUCK functionalities the forwarding agents must be able to communicate and negotiate with the road network area broker. That is, the forwarding agents now receive the announcements of the (initial or final leg) orders that are returned by the road network area brokers and they reply with a bid for that order. In order to provide the bid, they compute their potential plans for executing that order in collaboration with their transport subagents. The road network area broker selects from the offered plans the most suitable and returns it to the IPnEU, which, in turn, integrates it with the main leg plan and the second road haulage plan.

The trucks generated by the road simulation module will then be directed to the intermodal terminal gates. The synchronisation of the road simulation module with the terminal simulation is

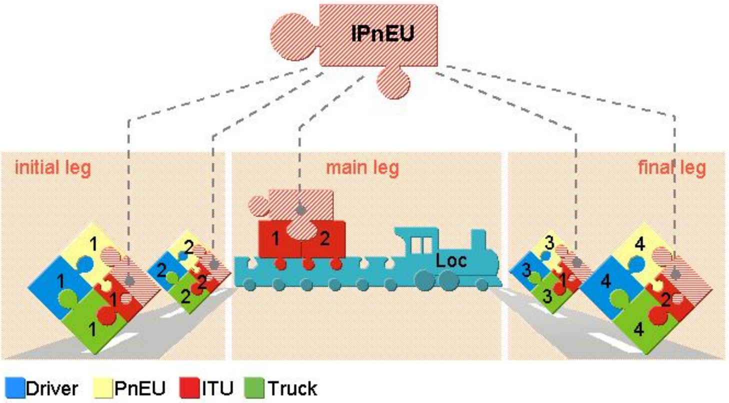

Figure 4: A holonic intermodal transport chain with two road holons and a rail holon.

described later in Section 7. Now we introduce the simulation of the rail-based leg of the intermodal transport, which involves the simulation of the intermodal terminal activities and of the rail corridor transports.

5 The terminal simulator

In the PLATFORM project we modelled the logistic chain as the interaction of road networks, terminals and the rail network. We do not take into account decisions taken in the rail network, which is seen as a black box, but we focused on modelling the effects of decisions taken in planning road transport and the management of the terminals, as requested by the user requirements of the project [8]. In the preceding sections we have examined what happens outside the gates of the terminal, now, thanks to the terminal simulator, we examine and assess the effects of management policies within the terminal and their side effects on the whole logistic chain.

The terminal simulator (TS) has been developed in MODSIM III [9], a commercially available object-oriented and process-oriented simulation language. The adoption of the object-oriented paradigm allowed software components to be defined that correspond to their real-world counterparts and with a similar behaviour. The terminal components modelled in the terminal simulator are:

- the road gate, where trucks enter and leave the terminal;
- the rail gate, where trains enter and leave the terminal. The rail gate is connected to the shunting area, outside the terminal, where the rail network operator shunts trains before they enter the terminal. The rail gate is also connected to the rail tracks inside the terminal;
- the platforms, each composed by a set of rail tracks and by a buffer area. The buffer area is a temporary storage area for ITUs that are waiting to be loaded/unloaded to and from trains entering the platform. Each platform is served by a set of gantry cranes, spanning the platform length and serving the set of rail tracks and the buffer area;
- the storage area, a longer term (usually 24 hours) area to park ITUs. The front lifters, serve the storage area, they serve trucks directed to the storage area picking up the ITUs and storing them in the storage area.

These components are implemented in the simulation code as classes, using the object-oriented programming language provided by MODSIM III. The modeller can easily assemble a rail/road terminal model creating instances of these classes. Moreover, since the simulation model reads from a database the structure of the terminal model and creates the instances of the model components, the modeller does not need to write code to create different terminal instances. This allows the model

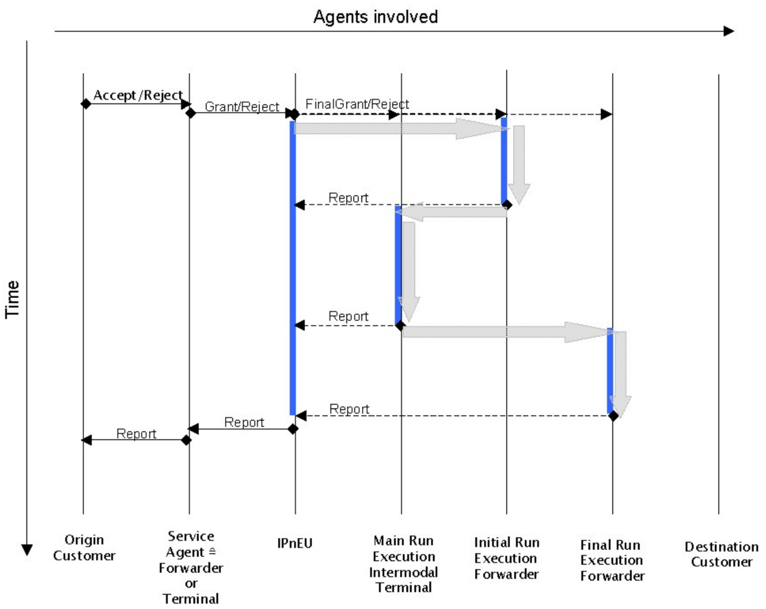

Figure 5: Execution protocol of an intermodal order (message and control flows).

to be quite generic and to be able to describe a variety of different terminal layouts and equipment.

The modeller creates model instances either specifying some characteristic parameters (e.g. the service time of a crane) or the subparts of a component (e.g. the number of rail tracks in a platform). In particular, s/he can define the yard layout: how many platforms are present in the model, the capacity of the associated buffer areas, the number of gantry cranes working on the platform, and the number of rail tracks in the platform. Only one storage area can be defined and its capacity must be entered too. For each gantry crane, the modeller must specify the average number of moves per hour and the crane operating cost per hour. Then, the terminal storage must be defined: the size of the storage area and the number of front lifters serving it, with their performances (number of moves per hour and cost per hour). Finally, the modeller defines the terminal interface to the external world. The road gate is identified by the number of lanes and the average time needed to service a truck. This service time is an aggregate representation of the time required processing the papers when a truck shows up at the road gate. The rail gate is represented by the number of shunting tracks, the number of link tracks between the shunting area and the terminal. The user must also specify the average time required to move a train from the shunting area into the terminal. This average value is easy to compute since it mainly depends on the distance from the shunting area to the destination rail track.

In the next sections we describe how the simulator handles truck arrivals and departures at the terminal gate (Section 5.1), and how the train loading and unloading processes are modelled (Section 5.2). Remember that ITUs which travel on the corridor are generated by the IPnEU and the inter simulation communication between the road simulator and the terminal simulator is described in Section 7.

5.1 ITU arrivals and departures

When a truck with an ITU arrives at the terminal, it joins a First In First Out queue at the road gate. Each road gate is represented by a FIFO queue. The service time of the road gate is a parameter set by the simulation user, who can also decide how many road gates are used in the simulation. When the truck has been processed by the gate and enters the terminal one of these three cases is given: a) the ITU arrives well ahead of the deadline (the time when the train on which it was booked must leave); b) the ITU arrives just before the deadline; c) the ITU arrives after the train has left. In cases a) and c) the ITU is placed in temporary storage areas, in case b) a direct transshipment on the platform is performed: the truck is directed to a queue associated with the train loading process and the ITU is directly loaded on the train. From the point of view of the crane, this kind of operation has a high priority.

Trains depart from the terminal according to a fixed timetable (see Section 6). This constraint is never violated in the simulation model and therefore it might happen that some trains depart before all the booked ITUs have been loaded. Such an event is used as an indicator of a problem in terminal management (e.g. insufficient resources to sustain the throughput).

When the train arrives to the destination terminal, it is directed to the shunting area where it waits for the availability of the rail gate (it might be engaged by another train). In addition, the availability of a rail track on the trains destination platform must be checked. When these preconditions are satisfied, the train may enter the terminal and the unloading operations start.

In the meantime, trucks are arriving to pick-up the ITUs delivered by the train. Truck arrivals for ITU pick-up are symmetric to arrivals for delivery, in the sense that trucks arrive generally after the train has arrived and the highest number of ITUs is picked up a few hours after train arrival.

When an empty truck arrives at the terminal, it waits in a queue at the gate and then enters the terminal. According to the availability of the ITU, either a direct transhipment on the platform is performed or the truck is directed to storage areas where the ITU has been stocked or to a waiting queue for incoming trains.

5.2 Train loading/unloading

The proposed modelling approach for train loading/unloading operations is platform-centred. A platform scheduler is associated with each platform, which assigns operations to the available cranes. The storage area is managed via a FIFO queue which accesses the pool of front lifters. Trains are modelled as sets of ITUs to be moved. Each move is an operation and a sequence of operations is a job. Thus, a sequence of loading/unloading operations for a train corresponds to a job. Each operation has a priority (for instance, if the ITU is to be picked-up by a truck). The platform scheduler assigns each operation to the available cranes, ordering by priority. Operations with the same priority are scheduled with a round-robin policy (one ITU for each job).

The order of the operations when loading a train is the following: first the ITUs on waiting trucks (direct transshipment) are loaded, second the ITUs on the yard. The trucks arriving when the corresponding train is being loaded join the queue of waiting trucks and are given the highest priority. This allows the loading process to be quicker and to minimise the time spent by the trucks waiting for service.

When unloading a train the order is: first move the ITUs on the waiting trucks, then the remaining ITUs. Thus, a truck arriving for pick-up is served with the highest priority. The cranes available for the loading/unloading operation must be allocated in the current work shift and suitable for the operation. The list of the cranes that will be active during a given work shift are set by the simulation user in the resource allocation table.

At least one gantry crane must be allocated at all times to perform both direct truck/train transshipment and storage in buffer areas. At least one front lifter is needed if the storage area must be accessed. When deciding which resource should be used to carry out an operation, it may happen that different operations compete for resources. A round-robin resource allocation policy was adopted to avoid starving and deadlocks.

Crane service time was modelled by a Gaussian distribution. In reality, each ITU is served in a time t, which is the average value of the time taken to travel along the rail track to pick-up/unload the ITU.

6 Rail corridor and rail network simulation

A rail corridor is a privileged point-to-point railway connection between two terminals, and it enables intermodal transport to try to compete with road-only transport not only in terms of cost, but also in terms of time. From the modelling view-point it consists of the allocation of appropriate time slots on the rail network. A corridor is thus an abstract representation of a path in a complex rail network.

Exploring the performance of intermodal transport over rail corridors was one of the objectives of PLATFORM project and the characteristics of the corridors made its simulation a simple problem to be solved. The simulation of a corridor link is driven by a timetable, which contains the departure and arrival times of trains. When a train travels from an origin terminal to a destination along a corridor, the simulator makes the train arrive in the destination terminal after a set time, given by the arrival time minus the departure time, plus a stochastically generated delay, to account for small deviations from the expected schedule.

The two nodes of the corridors are terminal models that are concurrently running in the same simulation. A train travelling in a corridor must be loaded in the source terminal and unloaded in the destination terminal. In the source terminal, it is represented as a list of ITU bookings. Booking are placed by the IPnEU, as described in Section 3, which will send trucks to deliver ITUs according to the booking details (hour of train departure, type of ITU, weight allowance, etc.). When the train is loaded and its departure time has arrived, it leaves the source terminal for its destination on the corridor. At the destination, the train is unloaded and trucks arrive to pick up the transported ITUs.

However, a terminal does not exchange ITUs only along a corridor, it is also linked to many other terminals, which often generate the major share of the rail traffic (we examined the Verona terminal in Northern Italy where traffic on the Verona-Munich (Southern Germany) corridor accounted only for 15% of the total rail traffic in one week). For this reason, the PLATFORM terminal simulator takes into account these external contributions by means of the rail network simulation. Again, a very simple simulation module generates train arrivals and departures according to a timetable. The difference with the previous corridor simulation is that departing trains must not be transferred to another terminal model, but they are simply discarded to represent the traffic of ITUs brought in by trucks and then leaving for other terminals, external to the corridor. On the other hand, incoming trains bring in new ITUs, which are later picked up by trucks, again contributing to the global traffic in the terminal. Note that the trucks which pick-up and deliver ITUs outside the rail corridor are not managed by the IPnEU agents. For this reason, a stochastic process artificially generates them, as described in [10].

The software structure used to model train arrivals and departures is a priority ranked queue, where the order is given by the time stamps associated with the departure and arrival events. During simulation, the train generation module inserts arrival and departure events in the Future Events List [9] of the terminal simulators. When the event time is reached, the train is then handled by the terminal rail gate and finally handed over to the inner terminal for the simulation of loading and unloading.

7 Inter simulation communication

The PLATFORM terminal simulation (TS) module has been designed in order to work in cooperation with the IPnEU agents. The aim of the IPnEU agents is to synchronise the truck arrivals in the terminal in order to minimise waiting times and reduce the queue length at the gates. If the planning of transport on the road network is well planned, the trucks would arrive in the best order, so the crane is making incremental moves, which are time efficient, and the required time to load/unload a train would decrease. If the planning is poor, the crane will probably travel back and forth to serve unexpected trucks, thus increasing the average service time. Thus, the advantage of planning the road network is that it is possible to improve the crane performance, since a better synchronisation of the truck arrivals would transform most operations in direct train/truck transshipments.

The TS is therefore fed by the IPnEUs. The distributed architecture of the PLATFORM simulation implies that TS and IPnEUs are different programs, exchanging data during their execution. In this section we describe the interaction of the two programs. Please note that all the processes are symmetric in the sense that the origin terminal in one transport plan can become the destination terminal in another plan.

7.1 Accessing information to generate transport plans

The IPnEU agents must interact with the TS to acquire the information required to place bookings on trains. First, an IPnEU queries any terminal to check whether a corridor exists between the origin and destination terminals. If so, the IPnEUs can book places on a train connecting the origin terminal with the destination terminal. The train departure times from the origin and its expected arrival time at destination are retrieved from the timetable of the rail corridor, stored in the TS database. Once a train has been chosen, the IPnEU fills in the bookings on this train. This procedure must be repeated for all the trains that leave the terminal during the simulation horizon.

The IPnEU must then schedule trucks which will deliver the ITU to the origin terminal from its origin on the road network and that will pick-up the same ITU at the destination terminal, thus bringing it to the final destination. The IPnEU must therefore generate the arrival of trucks at the road gate in the origin terminal. This gate is represented as a buffer which is periodically read by the TS module and used to generate the truck arrivals in the terminal model. The origin terminal then unloads the truck and sends it back to the road network, writing the truck data in the outward road gate.

In the meantime, the TS is loading the train for which the ITUs were booked. When the loading process is over, the train leaves the origin terminal and is transferred to the destination terminal via the corridor.

At the same time, the IPnEU has generated the arrival of a truck for ITU pick-up at the destination terminal, according to the transport plan. This truck is also inserted in the road gate, this time of the destination terminal. The terminal simulator uses this information to generate the truck arrival; it unloads the ITUs from the train on the truck and registers the outgoing trucks in the road gate, which is then accessed by the IPnEU to perform the last leg of the ITU delivery.

7.2 Synchronising the terminal simulation with the IPnEU agents

Given the distributed nature of the intermodal chain and the ability of IPnEU agents to migrate from machine to machine in a computer network, the two programs, TS and IPnEU, might be executed on different machines, which run at different clock speed, and their local simulation time can easily diverge. Lets assume that the two simulators start at the same instant, and consequently also their local simulation time is the same. In principle, just after a few units of processing time, the simulation time in the two simulators may have sensibly diverged. For instance, the simulation time in the IPnEU may lag behind the simulation time in the TS. In this case, when the IPnEU generates an event (a truck arrival), the TS sees an event with a time label which is in its past.

In Figure 6 it is graphically represented what happens if an event arrives after its simulation time has elapsed. The event D3 should have been executed by the TS before events I3 and I4. If there is no possibility to backtrack the simulation, cancelling events I3 and I4 (thus applying the time warp simulation method), the consequence is that the two simulators will proceed at the speed of the slowest of the two.

With the constraint of not being able to use time warp, because of the simulation environment currently adopted, the best approach, to minimise the waste of CPU time, is to adopt a unique future event list (FEL) where both programs write the next scheduled events. With such a shared structure, it would be possible to keep the two simulations synchronised. This is shown in Figure 7, where the same events of Figure 6 are ordered in a single FEL. When the TS starts, it finds that the next event is D1, scheduled at time 5. The TS would then wait that D1 is executed before executing I1, scheduled at time 7.

The synchronisation algorithm could be based on the mutual exchange of the next event time. That is, both simulators start at the same time, TS knows that the next local event is scheduled at time 7 and publishes it in a global variable (which can be a text file), while ITP knows its next local event is at time 5. TS then checks if the simulation time of event I1 is less than the simulation

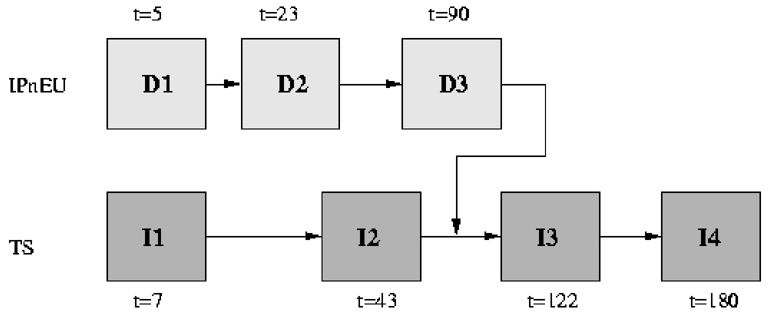

Figure 6: The Future Event Lists of the two simulations running independently.

time of D1, if not, it inserts a virtual event in its local FEL, thus jumping directly at time 5, when an event from ITP could be received. Note that not all the ITP events have an influence on the TS. If the interacting events would be known in advance, the synchronisation could be done on fewer events, but this is difficult to achieve, if not impossible, in a simulation where random events are generated.

8 Conclusions

We have presented an agent-based planner and simulator for intermodal transport, developed in the framework of the PLATFORM project. The software architecture is divided into the Intermodal Transport Planner, and in the simulation system. The former is an agent-based application which takes care of organising transport planes for the dispatching of Intermodal Transport Units along the various stages of an intermodal transport, from origin to destination. The latter is a discrete-event based simulator that has been designed and implemented to verify the feasibility of these plans and measures their performances. The simulator also focuses in the detailed modelling of the internal terminal processes in order to let the terminal managers evaluate the impact of new technologies and infrastructures on existing terminals.

9 Acknowledgements

This paper is based on work carried out under the European Commissions PLATFORM project (http://www.idsia.ch/platform) within the transport RTD programme (Task 3.2/7). We also wish wish to thank: Cosimo Epifani and Adriano Alessandrini for their contributions in the experimentation with the PLATFORM simulation module, Hans-Jürgen Bürckert, Gero Vierke,Volker Rueckel, Andreas Gerber, Ingo Zinnikus, Rainer Siedle for their support in the development of the Intermodal Transport Planner and Nicoletta Fornara for her contribution to the design and implementation of the Terminal Simulation module. The graphics for the ITP sections have been prepared by Jasmin Schneider.

References

[1] Directorate General VII of the European Commission, Combined goods transport: intermodality of goods transport, SCADPlus Union Policy (http://europa.eu.int/scadplus/leg/en/lvb/l24179.htm).

[2] J. Müller (Ed.). Verteilte Künstliche Intelligenz: Methoden und Anwendungen; BI Wissenschaftsverlag, Mannheim, Leipzig, Wien, Zürich; ISBN 3-411-16181-7; 1993.

[3] G. M. P. O'Hare and N. R. Jennings (Eds.). Foundations of Distributed Artificial Intelligence. Wiley and Sons, Inc.; ISBN 0-471-00675-0; 1996.

[4] R. G. Smith. The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver. IEE Transactions on Computers, Series C-29, 12, pp 1104–1113, 1980.

[5] H.-J. Bürckert, K. Fischer, G. Vierke, Transportation scheduling with holonic MAS the

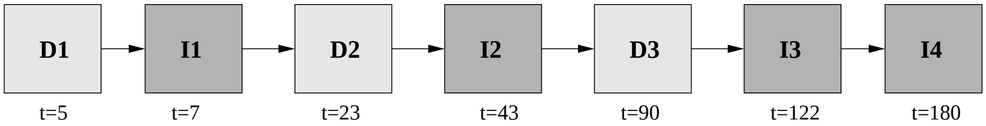

Figure 7: The Future Event List of the two simulation running interleaved in parallel.

TeleTruck approach. In Proceedings of the 14th European Meeting on Cybernetics and Systems Research, (Edited by R. Trappl), Vol. 2, (1998) 695-700.

[6] H.-J. Bı̌ckert, K. Fischer, G. Vierke. Transportation Scheduling with Holonic MAS The TeleTruck Approach. Proceedings of The Third International Conference on Practical Application of Intelligent Agents and Multi-Agent Technology (PAAM98), edited by H.S. Nwama and D. T. Ndumu, 1998.

[7] P. Funk. Fast Loading Devices: Planning and Scheduling Requirements. DFKI Technical Memo No. TM-98-03; DFKI GmbH Saarbrücken, Germany; May 1998.

[8] PLATFORM Consortium, Guidelines for a computer-controlled freight PLATFORM for a time-tabled railway freight network, user requirements and functions. PLATFORM Deliverable D1, 1998.

[9] CACI Products Company, MODSIM III Users Manual, La Jolla, 1997.

[10] A.E. Rizzoli, N. Fornara, L. M. Gambardella, A Simulation Tool for Combined Rail/Road Transport in Intermodal Terminals. To appear in: Mathematical and Computer Modelling, 2002.

[11] P. Funk. A Multiagent System for Intermodal Transport Chains. Dissertation Thesis, University of the Saarland, Germany. To Appear.