Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

Research
Report
RR-92-47

A Multi-Agent Approach towards Modeling Urban Traffic Scenarios

Frank Bomarius

September 1992

Deutsches Forschungszentrum für Künstliche Intelligenz GmbH

Postfach 20 80
D-6750 Kaiserslautern, FRG
Tel.: (+49 631) 205-3211/13
Fax: (+49 631) 205-3210

Stuhlsatzenhausweg 3
D-6600 Saarbrücken 11, FRG
Tel.: (+49 681) 302-5252
Fax: (+49 681) 302-5341

Deutsches Forschungszentrum für Künstliche Intelligenz

The German Research Center for Artificial Intelligence (Deutsches Forschungszentrum für Künstliche Intelligenz, DFKI) with sites in Kaiserslautern and Saarbrücken is a non-profit organization which was founded in 1988. The shareholder companies are Atlas Elektronik, Daimler Benz, Fraunhofer Gesellschaft, GMD, IBM, Insiders, Mannesmann-Kienzle, Philips, SEMA Group Systems, Siemens and Siemens-Nixdorf. Research projects conducted at the DFKI are funded by the German Ministry for Research and Technology, by the shareholder companies, or by other industrial contracts.

The DFKI conducts application-oriented basic research in the field of artificial intelligence and other related subfields of computer science. The overall goal is to construct *systems with technical knowledge and common sense* which - by using AI methods - implement a problem solution for a selected application area. Currently, there are the following research areas at the DFKI:

- ❏ Intelligent Engineering Systems
- ❏ Intelligent User Interfaces
- ❏ Intelligent Communication Networks
- ❏ Intelligent Cooperative Systems.

The DFKI strives at making its research results available to the scientific community. There exist many contacts to domestic and foreign research institutions, both in academy and industry. The DFKI hosts technology transfer workshops for shareholders and other interested groups in order to inform about the current state of research.

From its beginning, the DFKI has provided an attractive working environment for AI researchers from Germany and from all over the world. The goal is to have a staff of about 100 researchers at the end of the building-up phase.

Prof. Dr. Gerhard Barth
Director

A Multi-Agent Approach towards Modeling Urban Traffic Scenarios

Frank Bomarius

DFKI-RR-92-47

Parts of this report have also appeared in:

A. Lux, F. Bomarius, D. Steiner: A Model for Supporting Human Computer Cooperation. in: Proceedings of the AAAI 92 Workshop on Cooperation among Heterogeneous Intelligent Systems, San Jose, CA. July 1992.

F. Bomarius, D. Steiner: A Model for Supporting Human Computer Cooperation. at: ECAI-92 Workshop: Application Aspects of DAI. Vienna. August 1992.

F. Bomarius. Using the MECCA System to implement Urban Traffic Scenarios. DFKI Technical Memo TM-92-07. September 1992.

This work has been partially supported by the European Community as part of ESPRIT II Project 5362 IMAGINE.

A Multi-Agent Approach towards Modeling Urban Traffic Scenarios

F. Bomarius

DFKI Kaiserslautern

September 1992

Abstract

This paper investigates the use of the **multi-agent paradigm** *in modeling urban traffic scenarios. It demonstrates that vehicles, pedestrians, traffic-lights, car-parks and even streets can be considered agents in a heterogeneous multi-agent system. Different types of agents in such scenarios will be identified, characterized and constructed by virtue of a general* **agent model**.

The various kinds of relationships and interactions, generally called, **cooperations** *between these agents will be modeled; some examples sketch the major issues developed in this paper.*

Contents

1 Investigating Urban Traffic **2**

2 Using Agents to Model Traffic Scenarios **5**
- 2.1 The Notion of Agents . . . 5
- 2.2 A General Agent Model . . . 6
- 2.3 The Society of Cooperating Agents . . . 7
- 2.4 User Agents . . . 7

3 Examples **8**
- 3.1 Multi-Agent Scenario: Crossroads . . . 8
- 3.2 Multi-Agent Scenario: Car Approaching a City . . . 9
- 3.3 Agents in the Urban Traffic Scenario . . . 10

4 Research Projects in the Field of Traffic Control **13**
- 4.1 PROMETHEUS . . . 14
- 4.2 LISB . . . 15

5 Conclusion **16**

A Towards an Implementation **17**
- A.1 Street Agents . . . 17
- A.2 Crossroads and T-Junction Agents . . . 18
  - A.2.1 Crossroads Agent . . . 18
  - A.2.2 T-Junction Agent . . . 19
  - A.2.3 Computing Flow Volumes . . . 19
- A.3 Vehicle Agents . . . 22
- A.4 Conclusion . . . 23

Chapter 1

Investigating Urban Traffic

To participate in traffic, in particular in urban areas with a high volume of traffic, is a highly interactive task. Various types of road users have to continuously adjust their actions in order to avoid crashes and to get on to their destinations. From the point of view of the individual, he or she merely watches the environment and reacts appropriately in order to achieve his or her goals (to get on to some place, survive, etc.). These local goals are the driving forces that let the individual move and behave in certain ways.

In most situations this strategy is successful since human beings, like many other beings, have developed appropriate behaviors throughout their evolution that allows the individual to proceed in the face of arising conflicts. Some inherited and some learned behaviors plus some additional rules (e.g. traffic regulations) suffice. So, in most cases, traffic flows and the participants achieve their goals.

However, sometimes too many individuals with too many conflicting goals meet. This usually results in a traffic-jam or, in severe cases, a crash. The latter will not be investigated here; the former seems worthwhile thinking about, since it has to do with unintentional misbehavior (misguidedness) due to a lack of environmental information. In other words: nobody would voluntarily join a traffic-jam if he or she knew in advance that it will occur. Thus, anticipation is an important factor for the reactive behavior of road users. At present, the scope of anticipation is limited to the range of the human senses, in particular seeing and hearing.

Participating in traffic is, to a great extent, a matter of communication. Humans have communication channels that, in the past, perfectly served needs with regard to bandwidth, sensitivity, speed and range. They turn out to be insufficient in the face of high volumes of traffic including fast moving vehicles and great numbers of individuals. In such complex situations humans can not anticipate, from their local point of view, the overall behavior of the majority which is necessary in order to react in a timely fashion. Radio traffic services try to extend the 'senses' of the individual by providing information about situations that lie beyond the individual's range. Unfortunately, these services tend to be slow and misguiding. They often provide information when it is too late to react appropriately. In some cases the situation gets even worse due to outdated information or over-reactions of the receivers.

Information provided by those services is coarse grained and unspecific from the point of view of the individual road user. In order to find an optimal or at least a good route, with respect to time and distance, each road user requires different sorts of information. At present several sources of information are available:

1. The well known *radio traffic services* provide a general overview of the current traffic situation, similar to a snapshot. This overview is of limited use for the individual in forecasting future traffic situations since it does not take into consideration the destinations and planned routes of the road users currently on their way.

2. Since each road user has *common sense knowledge*, he or she can anticipate situations that are likely to happen on a regular basis. For example, every morning and afternoon rush-hour traffic must be expected. At the beginning and end of vacations or on holidays like Christmas, certain typical traffic situations can be foreseen.

3. In much the same way, *experiences* about peculiarities of regularly used routes like road-works, traffic-lights that are not adjusted to the varying traffic volumes, notoriously overcrowded car parks and the like have influence on decisions of the individual.

4. *Observations* of the environment, e.g. the weather or an accident up ahead, may be of immediate impact on decision making.

Based on this information, each road user plans a route, selects a means of transport and calculates the estimated time needed. While en route deviations from this plan are usually not possible. In particular, the road user nowadays would not get enough information to redo planning and come up with an alternative, more viable, plan.

This is where computer and communication technologies come in. If each road user could be embedded into a network of telecommunication links that spread all over the country and connect every relevant passive or active entity involved in the traffic scenario then significant improvements are conceivable. In the following chapter such a network will be outlined and some expected advantages will be pointed out. The main goal of this paper is to demonstrate the applicability of the major concepts of the so called *multi-agent paradigm* for the application domain 'urban traffic scenario' (UTS). Due to the complexity of the problem domain it is not possible to give a comprehensive treatise. Hence, many interesting aspects can not be mentioned here which should also be investigated, such as vehicles moving on autopilot or systems for automatic collision prevention. This paper will, therefore, focus on a choice of aspects abandoning (but not ignoring) the others.

Two major areas of research have impact on the UTS domain, namely electrical engineering (telecommunications in particular) and computer science:

Electrical Engineering Most of the technologies that have to be taken in consideration here have already reached high standards and are, for a large part, ready to use:

- Remote Sensing
- Signal Processing
- High Bandwidth Radio Communication
- Networking

In the following it is assumed that these technologies are freely available as ready-to-use components which interface easily to computers. All data sampled or transmitted by these components originate from or are handed to these computers. Low level issues like message formats, means of transmission (radio, infra-red light, etc.) and interfacing to computers (including conversion of signals to symbols) will not be mentioned here.

Ubiquitous computing power is assumed [1], so every entity involved in the traffic scenario is equipped with a local (or has direct access to a remote) computer which is in direct contact with the entities' computers in the environment.

Computer Science adds to the above listed issues the concepts, methods and technologies to exploit and hopefully control what is technically feasible. In particular, the following areas of research are relevant here:

- Distributed Artificial Intelligence (DAI)
- Computer Networks
- Distributed Processing

The above mentioned ubiquitous computing requires powerful models of interaction between the vast amount of entities involved in traffic scenarios. Research in DAI, distributed processing, and computer networking has recently begun to reveal the prerequisite technologies to tackle the problems at hand.

The details of computer networking and distributed processing are not investigated here. Emphasis will be put on the DAI concepts.

It is the goal of this paper to apply DAI concepts to the UTS. The discussion will be accompanied by some examples that demonstrate the appropriateness of the major concepts of the multi-agent paradigm in the traffic control scenario.

Note that since this paper is a first approach to apply multi-agent system technology to a real world scenario of very high complexity, simplifications will be introduced in order to keep the examples comprehensible. Nevertheless, the author believes that all concepts presented here easily scale up to and perform well in real world conditions.

Chapter 2

Using Agents to Model Traffic Scenarios

Distributed computing scenarios have been under investigation for a long time. But, due to the lack of technical feasibility of distributed systems that are comprised of a considerable number of computing nodes, research was focused on problems that were tractable on a theoretical basis. A significant property of the results of this work is that, in most cases, the resulting multi-processor systems rely on regular interconnection patterns and are comprised of identical (special purpose) computing nodes. These *homogeneous* systems could be realized with reasonable efforts and were successful in very specific application domains.

Recently, *heterogeneous* computer networks which are comprised of different kinds of general purpose computing nodes that are interconnected in non-regular and/or regular fashions have been investigated. The results of DAI (in particular distributed problem solving) and, to some extent, work in the field of HCCW (human computer cooperative work) [8] [7] promoted the development of concepts for these heterogeneous computer networks. The term *multi-agent system* was coined to denote conceptual work in this field [2].

Whereas the above mentioned 'classical' distributed computing is of minor use use in heterogeneous domains such as traffic control, multi-agent system technology provides a much more appropriate approach.

In the following sections a brief introduction to multi-agent systems, MAS for short, is given. Next the MAS approach to the traffic scenario will be developed.

2.1 The Notion of Agents

At present, the MAS-researchers have not yet settled on concise and unopposed definitions. So the term *agent* is used in an intuitive fashion [2]. It refers to the computational representation (simulation) of some entity in the real world. The agent, like the respective real world entity, exhibits observable behavior. Note that this entity does not have to have substance, it may be an abstract concept such as a bank account.

Agents may be compared to *objects* known from the domain of object-oriented programming. In fact agents can be implemented as objects. But agents are considered to be a higher level concept in that they abstract from the implementation details entailed by the term 'object'. Agents are much more sophisticated, they are autonomous and often thought of as 'intelligent', they are comprised of logical components whose implementations may be neglected.

Agents stand for a wide variety of real world entities. In the traffic scenario agents may be intelligent robot-vehicles as well as rather simple traffic-lights or parking lots. Even humans may be considered agents.[1] So the model of agents has to fit a broad spectrum of relevant entities in the world of discourse.

[1] Note that the author does not think of humans themselves as agents in the MAS sense. Humans are considered equipped with a machine agent, which acts on their behalf and interfaces its user to the MAS. This special type of agent is called *user agent* (see also Section 2.4).

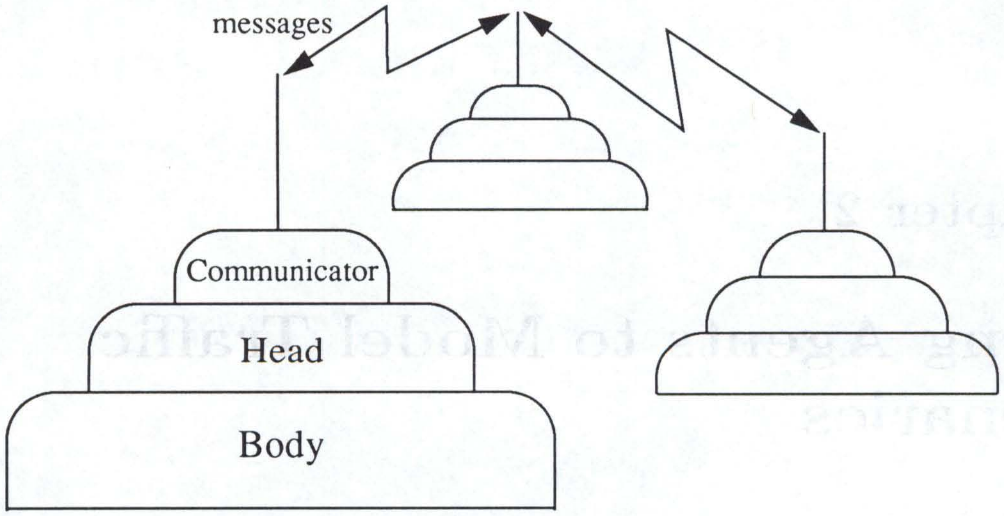


Figure 2.1: Agent Structure

2.2 A General Agent Model

Every entity in the world interacts with its environment. Interaction may be on its own behalf (active entities) or due to external stimuli (re-active or passive entities). Interaction may be through different kinds of media. In the world of computer-based agents *message-transmission* between agents is the usual (and sufficient) way to model interaction. (Others are conceivable as well, but will not be investigated here.)

Closely related to interaction is the *observable behavior* of an entity. This is the way it (re-) acts in given situations. Given that an agent interacts only by means of messages, its behavior is a mapping from received messages to emitted messages. It is assumed that (most of) this behavior is explicitly represented in the computer-based agent.

Since, in the real world, nothing gets altered by pure message passing, some *functionality* must be introduced into the model that either imposes changes to the physical world (by means of actuators) or computes new information that may be communicated. This agent-specific functionality is the means by which the *world state* is altered.

According to these observations every agent can be divided into three main components, namely *communicator*, *head* and *body* which implement message-transmission, the observable behavior and the functionality respectively. Figure 2.1 sketches the agent structure on this abstract level. In this model all interactions are modeled by means of message passing.

Communicator The communicator is in charge of establishing and maintaining the physical communication channels to other agents in the environment. It continuously 'listens' for incoming messages, delivers them to the head, and emits messages originating from the head.

For the whole agent to communicate efficiently, its communicator has to provide a sufficiently abstract communication interface to the agent's head. That is, it has to hide all technical details of communication and provide reliable transmission channels to other agents in the environment.

Head The head, relieved from primitive communication tasks by virtue of the communicator, has to plan, negotiate and interact on a high level of abstraction. It may be conceived as a knowledge-based component, being aware of goals, intentions, functionalities and resources of its associated body and of other agents as well. It strives to achieve local and/or common goals, while acting simultaneously in different interactions. This, all together, comprises the observable behavior of an agent.

Knowledge based decision making components are considered necessary only for sophisticated agents. Simpler agents, like e.g. a traffic-light, might do without knowledge based components. They will not participate in complex negotiations with other agents, but will rather react on a small number of messages they understand. Their behavior is representable by means of hard-coded procedures; planning and knowledge-based decision making seem dispensable.

Body The agent's body constitutes those parts of its overall functionality it performs in order to (intentionally) alter the world state. While communicators and even heads may be quite similar for different types of agents, their bodies will differ significantly.

In the traffic scenario, for example, only vehicles have functionalities that enable motion, while traffic-lights, streets or parking lots do not. Bodies of vehicles and traffic-lights both emit optical signals. Humans and vehicles may also employ acoustic signals and gestures.

It is noteworthy that the model presented is by no means domain specific. In particular, the agent structure allows to describe agents of almost any domain in a general fashion.

2.3 The Society of Cooperating Agents

Multi-agent systems are often paraphrased by the metaphor 'society of cooperating agents'. This metaphor stresses the most prominent aspects of multi-agent systems:

Heterogeneity: A MAS is a heterogeneous collection of agents. Each agent has specific properties (behavior, capabilities, authorities) that distinguishes it from other agents. Similar agents may be considered as being of a common type. For example all traffic lights in UTS are of the same type.

Cooperation: Agents interact. Some interactions will be cooperations between agents working on a common task or goal. Agents cooperate in order to achieve goals they could not achieve on their own. For instance in the UTS all traffic-lights of a crossroads may cooperate (by adjusting the timing of the light sequence) in order to maximize throughput and minimize the likelihood of traffic-jams.

Grouping: Cooperating agents may join together and form groups. The grouped agents collaborate in order to achieve common goals. They may be represented by a special agent called the *group agent* which is the representative of the group in interactions with other agents or groups. An individual agent may be member of different groups simultaneously. Furthermore, groups may form higher order groups. For example, in the UTS, parking lots may join into a car-park and all car-parks of a city may jointly interact in order to balance parking space occupation.

2.4 User Agents

Humans are integrated into this 'machine society' by means of *user agents*. These agents can be considered prosthesises that widen the range of human senses by providing information they could not gather otherwise. Information is guaranteed to be up-to-date and is filtered according to the current needs. So a user will not be overwhelmed with outdated or irrelevant data, but will receive information that concerns the goal at hand, i.e. reaching the current destination.

Furthermore, the user agent will, depending on its level of sophistication, act intelligently on behalf of its human user. Thus, a user agent can be regarded as an 'intelligent assistant' or 'junior partner' of the human user [3].

In the UTS domain a user agent might be built into cars, or might be publicly available at bus stops and underground stations to provide users with the required information. In the future it is conceivable that everybody carries a personal user agent, that continuously connects to other agents in the environment [1].

In the next chapter we revisit the above mentioned aspects in the context of some exemplary traffic situations.

Chapter 3

Examples

3.1 Multi-Agent Scenario: Crossroads

Figure 3.1 shows a typical traffic situation at an intersection. Although it captures only a small part of the overall traffic scenario it allows to demonstrate the applicability of most of the MAS concepts in the UTS.

Grey arrows indicate goals of the agents while black arrows visualize communication links.

There are two eastward bound cars: `car-1` and `car-2`. As indicated by the gray arrows `car-2` intends to cross `NS-Street`, while `car-1` intends to turn right into `NS-Street`. Both cars are agents that communicate with the next traffic-light agent ahead, i.e. `tl-1`. So `tl-1` gets informed about the intentions of `car-1` and `car-2`. `car-3` which intends to turn left into `EW-Street` in order to occupy the free parking lot communicates with `tl-2`.

Traffic at the crossroads is controlled by the cooperation of the four traffic-lights. All four traffic-light agents *negotiate* about the length of the phases of the next light-sequence in order to minimize waiting-time for the cars.

Since there are currently no cars approaching `tl-3` and `tl-4`, these lights will not impose further constraints on the determination of the next light sequence. So `tl-1` and `tl-2` will decide if they give preference to `car-3` or to `car-1` and `car-2`.

`car-3` is in contact with the `car-park` because its driver wants to park the car. `car-park` is an agent which communicates with approaching cars on behalf of the set of `parking lot` agents. It represents all `parking lot` agents that comprise the `car-park`.

This type of agent is called a *group agent*. Group agents are responsible for the cooperation among group members and for the cooperation of the group with the world. A group agent hides the details of the group organization and provides a (sophisticated) service to clients of the group. Depending on the organizational principles of the group, the group agent may have the authority to give orders to the group members or it may interact with them on a peer-to-peer basis. The group agent may be a special agent designed to organize group behavior or it may be one out of the group that has been selected as group representative.

The four traffic-lights form a group as well but, as opposed to the car-park group, they do not have a representative agent. Each member of the group communicates with the world on its own behalf and on behalf of the group. Here, group behavior is determined by peer-to-peer negotiations, i.e. the group members have equal rights and responsibilities.

The scenario of Figure 3.1 exemplifies how client agents (cars) contact agents with higher authority (traffic-lights) in order to receive orders (light signals from the traffic-lights). It gives two examples of how agents may organize into groups. The traffic-lights make up a group of equal members where each member communicates with clients, whereas the car-park represents and governs a group of parking lot agents. For the parking lots, communication is restricted to the car-park group.

Thus, this scenario is an illustration of the main aspects of multi-agent systems which were listed in Section 2.3. It shows a *heterogeneous* set of agents that *cooperate*, some of them joining into *groups*.

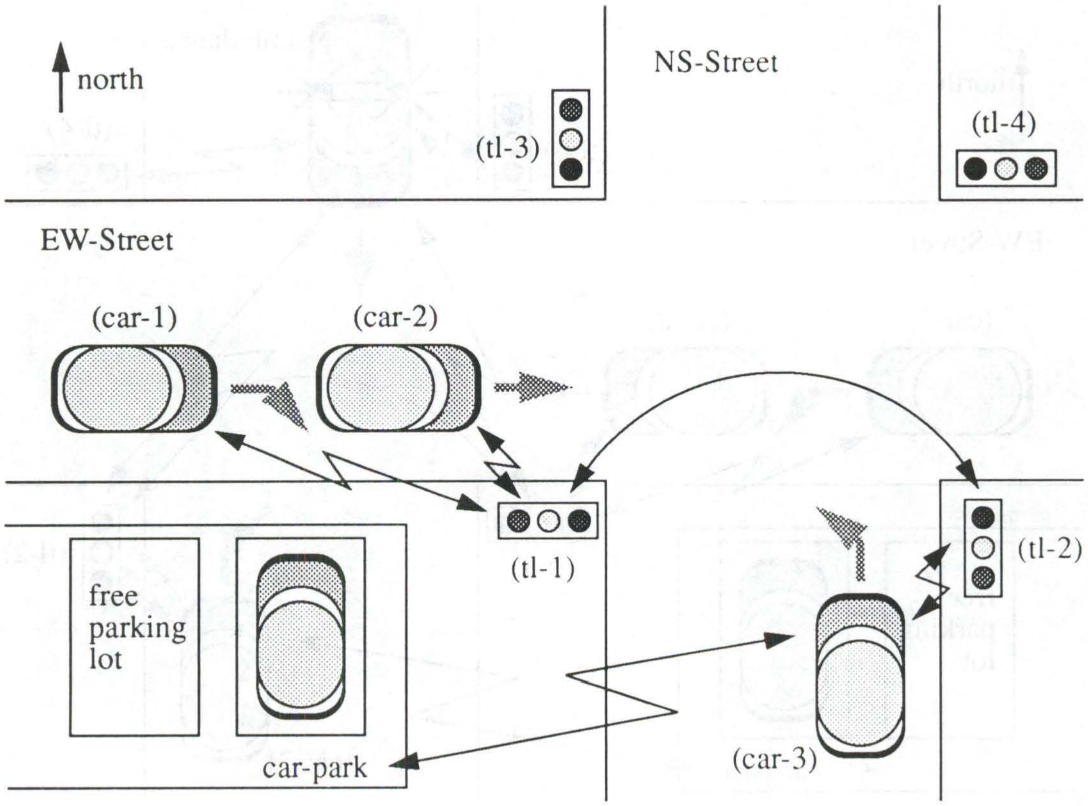


Figure 3.1: Crossroads Scenario

Figure 3.2 is a variation on the previous example. An additional car agent, `ambulance`, has been added to the scene. It intends to cross `EW-Street` and wants to take precedence over all other cars currently approaching the intersection. The `ambulance` introduces an exception to the standard operations of the traffic-lights: through communication with the traffic-lights the ambulance gives the order to immediately block all directions.

This exemplifies the way how certain agents may override standard operations, at least in a limited region, due to their higher authority.

3.2 Multi-Agent Scenario: Car Approaching a City

The example depicted in Figure 3.3 gives an overview of what happens when a mobile agent approaches a city. A prerequisite to this example is a standard mechanism for enabling the mobile agent to contact the traffic control agents.

As Figure 3.3 suggests, a hierarchical organization of the traffic control agents is assumed which results in the grouping of subordinate agents under their superiors.[1] Not only the immobile agents are part of that hierarchy but also pedestrians and all kinds of vehicles. As opposed to the immobile agents they are only temporarily assigned a place in the hierarchy. As they move, their placement in the agent hierarchy changes, e.g. they leave town, become immobile while parking, or pass from one district to another.

While approaching the outskirts of the city, the car in Figure 3.3 has to register with the `traffic flow control` agent. This agent samples data about all mobile agents entering or leaving the urban region. Each of them has to declare its destination(s), preferred route(s), speed, estimated time of

[1] The hierarchy shown in Figure 3.3 is for illustration purposes only and is not considered to be a realistic model. In particular a MAS-hierarchy for a realistic town is much too complex to be depicted clearly within the limited space of this paper.

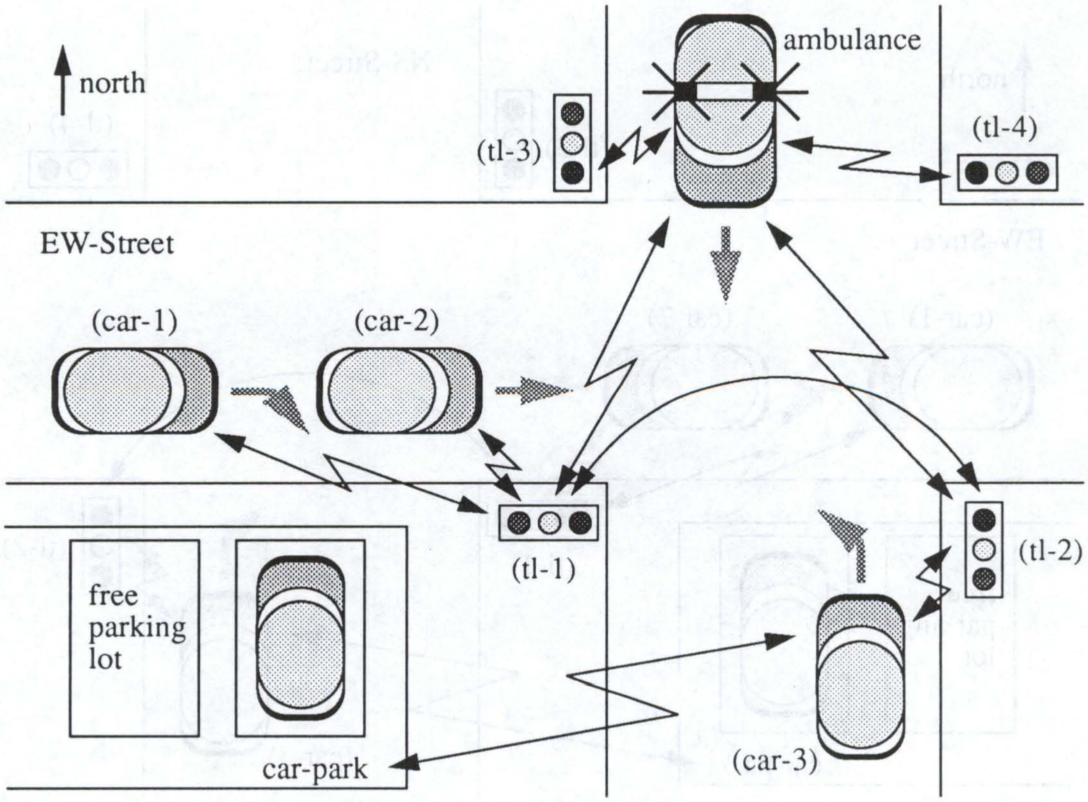


Figure 3.2: Crossroads with approaching Ambulance

departure etc. The **traffic flow control** agents calculate a route for the approaching agent based on their current knowledge about the (estimated) traffic volume and parking space occupation.

Based on the calculated route, the approaching agent is brought in contact with the subordinate agents responsible for the districts the agent will pass through.

Assuming that each agent gives honest information about its intentions, the consequences for traffic flow control may be far reaching.[2] Based on this information, traffic volumes can be forecast and traffic flow can be controlled so that it comes close to an optimum with respect to global throughput and traffic-jam avoidance. In particular, exceptional events, e.g. road works, can be taken into consideration when calculating routes for mobile agents.

Furthermore, foreign road users, not familiar with their route, may be continuously guided by the traffic flow control system bringing them directly to their destinations.

Even accidents will not unconditionally lead to traffic-jams. The continuous supervision of mobile agents in the proximity of the accident allows calculating individual detours for these agents mitigating the effects on traffic flow.

The benefits of a MAS organization of the UTS for traffic flow control seem to be enormous but a complete treatise goes far beyond the scope of this paper. In the next section the individual entities of the UTS are investigated in detail with respect to the MAS aspects.

3.3 Agents in the Urban Traffic Scenario

In the examples given above, some of the entities in the UTS have been described as agents. This section summarizes the results and investigates each type of agent individually.

[2]The author assumes benevolent agents. For some domains this might be an unrealistic assumption, but it is an appropriate or even necessary assumption in scenarios where reliable forecasts are to be made.

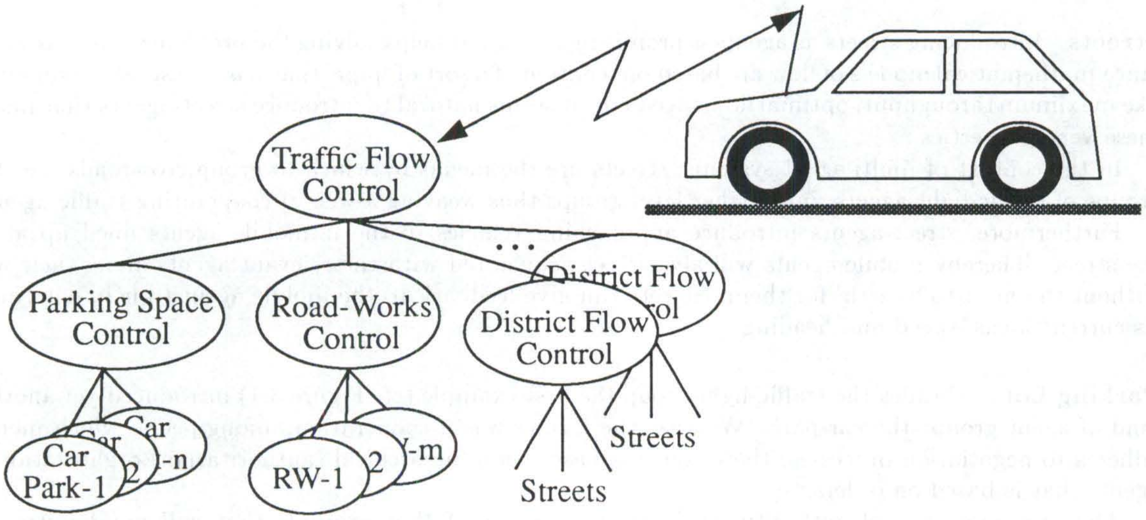


Figure 3.3: Car approaching a City

Vehicles Vehicles, i.e. any mobile agent except pedestrians, are agents that incorporate the necessary computational power and telecommunication facilities to establish and sustain contact with a considerable number of agents in their environment. The speed with which vehicles move requires not only that they be able to handle a great number of communications in parallel, but also to frequently make new 'acquaintances' and terminate communications rapidly.

Since vehicles rely on guiding information from their environment, i.e. mostly from the immobile agents, they are to some extent subordinate to these. It is a matter of future research how to properly adjust the authority levels in order to not over-constrain the vehicles.

Pedestrians For pedestrians to become part of the MAS-world in the same way as described for vehicles some technological progress is required. Each person would have to carry a sort of pocket computer (whose computational power is beyond what is commercially available at present) that makes its owner an agent in the MAS (cf. Section 2.4). Aside from the political and social discussions about supervision of individuals that such a device would trigger, it may be claimed that, at least for the purpose of guiding people to a destination, it is a valuable concept. Many more advantageous properties of this device are conceivable, e.g. benefits for handicapped persons.

Since such a device would turn out to be a universal communication facility, it allows its owner not only to interact with the immediate environment, performed mostly automatically by the device, but also provides access to information systems at any time at any place. For instance schedules of public transport systems may be accessed and reservations could be made instantaneously with the help of this device.

Note that equipping persons with a portable 'user agent device' is optional in the traffic control scenario.

Traffic-Lights The roles of traffic-light agents have already been described in the above examples. The task of a singleton traffic-light is rather simple. It has to recognize (i.e. count) approaching vehicles (and pedestrians). Based on their estimated arrival times the timing of the light sequence at the crossroads can be adjusted appropriately. Since, in most cases, a traffic-light belongs to a group it can not alter the light sequence timing on its own but has to negotiate with its partners. Each traffic-light agent knows about the traffic volume approaching from the direction it is controlling. The integration of the knowledge of all members of the group allows to calculate the timing of light sequences that is optimal with respect to overall throughput.

Unfortunately, the local optimizations at each crossroad will not lead to globally optimal results. Therefore, the dependencies that exist between sequences of crossroads (lined up along streets) have to be taken into consideration.

Streets Introducing streets as agents is promising because it helps solving the problems of flow control. Since mathematical models of flow are based on a notion of a sort of 'pipe' that has measurable properties like maximum throughput, optimal flow speed etc. it seems natural to introduce street-agents that model these very properties.

In the context of multi agent systems, streets are the means by which to group crossroads, i.e. the groups of traffic-light agents, into higher level groups thus weaving a web of cooperating traffic agents.

Furthermore, street-agents introduce approaching vehicles to the immobile agents lined up along the street. Thereby mobile agents will always get acquainted with the relevant agents along their way without the need to 'search' for them. Streets can give feedback to the mobile agent to help determine its current locus, speed and heading.

Parking Lots Besides the traffic-light group the first example (cf. Figure 3.1) introduced yet another kind of agent group: the car-park. Whereas the former was a cooperation among peers, which merely adheres to negotiation processes, the latter was based on a hierarchical (authoritarian) organization of agents that is based on orders.[3]

The pros and cons of authority levels are one aspect of that example that will not be stressed here since it depends on the application at hand. A more general aspect is the way how hierarchical grouping allows for hiding the details of group organization. A client to the group need not even know that he is actually interacting with a group. Furthermore, authoritarian group organization reduces communication overhead between group members and promises timely and consistent behavior of the group. In particular predetermined group configurations that are not likely to be reconfigured and that operate mostly on standard situations (e.g. the traffic-lights) are candidates for authoritarian group organization.

[3]Note that the choice of describing traffic-lights as peers and parking-lots as subordinates of a car-park agent is only for illustrating some of the different options in the organization of agents. There is no reason not to introduce a crossroad-agent that gives orders to its subordinate traffic-lights or to organize a car-park based on peer-to-peer cooperation of parking lots.

Chapter 4

Research Projects in the Field of Traffic Control

In the past a variety of traffic control systems have been developed and installed that strive to improve traffic situations with respect to one or more of the following general issues:

Security - E.g. notification of road users of weather and street conditions ahead, improvement of vehicle technology.

Flow - E.g. anticipation of jams, coping with increase in traffic volume in general.

Ecological damage - E.g. improvement of efficiency of drive-train technology, homogeneous traffic flow, avoidance of unnecessary (de-) tours, promotion of the use of public transportation systems.

Any traffic control system is concerned with a collection of these issues, but their developers weight them differently:

The major goal of the older systems is to cope with almost unlimited increase in traffic volume and to enforce security for these volumes of private traffic by means of road construction.

Emphasis has shifted in the recent years. Security issues have become increasingly important in the public opinion, even though acceptance of the necessary steps, such as speed limits, is still not very high. The ecological consequences of private mass transportation have become publicly noticed recently, but yet only a minority is willing to take the appropriate steps.

This ongoing shift has consequences for the development of future traffic systems. Since security and ecological issues are considered more important now, completely different systems will emerge; some of their typical characteristics are [6]:

- A general shift from private to *public transportation means* will occur. Beginning with the (short distance) urban traffic systems acceptance of public transportation means will increase also for long distances.

- As opposed to the older systems that cope with road users as an anonymous mass, future systems will be much more concerned with the *individual* needs of customers. Therefore, the integration of private and public transportation means has to be enforced (e.g. park-and-ride systems) along with a general shift towards public systems. These will have to offer services customized to the individual needs of each user.

- Future traffic systems will be *information processing and transmission* systems. As was pointed out in the previous chapters, intelligent networking and cooperative interaction among the components of the traffic systems will be crucial for the performance of these systems.

As was pointed out in the previous chapters, the MAS approach to traffic control exhibits these very characteristics.

Traffic control systems are embedded in a network of interests. Future developments will have to account for them in order to be publicly accepted:

Social interests - Concerning business as well as spare time activities.

Economical interests - Such as unrestricted pursuance of one's profession.

Ecological interests - Avoidance of exhausts and noise, saving resources.

Aesthetical interests - Preserving the quality of residential areas, parks etc.

The hypothesis is that all these interests can be reconciled in the future by employing cooperatively working, interactive systems. Here, MAS technology provides a promising prospective.

Since a comprehensive treatise goes beyond the scope of this paper, only two of the contemporary (German and European) research projects are sketched here. Namely the projects PROMETHEUS and LISB will be described, representing vehicular traffic control projects in general.

4.1 PROMETHEUS

The project PROMETHEUS (**PRO**graM**m**e for an **E**uropean **T**raffic with **H**ighest **E**fficiency and **U**nprecedenced **S**afety) [4] [5] is a joint research effort of the Western-European automobile industries and research institutes. It is one of the largest EUREKA projects.

The main objectives are improving safety and reducing ecological effects of the transport systems. PROMETHEUS relies mostly on improvement of vehicle technology. Development of intelligent vehicles is pursued, but minor emphasis is put on improvement of the infrastructure. The overall project goals are a sequence of steps:

1. cars that give information,
2. cars that emit warnings,
3. cars that give instructions,
4. cars that correct the driver's actions,
5. cars that steer themselves.

PROMETHEUS is a compound of seven research areas that are assigned to the partners. Industrial research encompasses the following three areas:

PRO-ROAD Development of communication systems linking infrastructure and on-board computers that allow for decentralized traffic control.

PRO-NET Development of a communication system linking vehicles in order to increase the driver's perception range and to guarantee safe and harmonized traffic flow.

PRO-CAR Development of an intelligent on-board system that informs and actively supports the driver.

Basic research, done at universities, encompasses the following four areas:

PRO-GEN Conceiving traffic scenarios that allow analysis and evaluation of the systems developed.

PRO-COM Development of standards and protocols that enable communication among on-board computers of vehicles and between vehicles and the infrastructure.

PRO-CHIP Implementation of the required computer hardware.

PRO-ART Development of procedures and algorithms that allow utilization of artificial intelligence in future traffic systems.

The common property of all research done in project PROMETHEUS aims at improving private transportation means. Integration of public and private means is of minor concern. In particular PROMETHEUS does not pursue reduction of private traffic volumes.

4.2 LISB

LISB (Leit und InformationsSystem Berlin) [5] is a field study carried out in Berlin. The prototypical system is a navigational aid that pursues traffic control tasks by individually guiding anonymous road users. A similar system, called AUTOGUIDE, will be installed in London.

LISB is comprised of stationary sender/receiver units that are distributed all over the town. Vehicles continuously transmit sensory data such as the time required to move along a road to the system. LISB samples the data and updates its centralized database. Every 5 to 10 minutes the latest information about the current traffic situation is broadcast.

Vehicles continuously listen to these broadcasts. Their on-board navigation computer filters and processes the informations with respect to the vehicle's destination. It derives guiding hints and presents them to the driver by means of a display and natural language utterances.

Based on the centralized processing of up-to-date traffic volume data individual guidance can be provided with respect to the current traffic situations. Note that all road-users stay anonymous. LISB is based on a central database which is continuously updated by the incoming vehicle data and which continuously generates the latest traffic guidance information. Furthermore, this database allows to deal with:

- traffic management according to given policies,
- traffic light control,
- radio traffic services,
- park-and-ride systems, parking space occupancy control,
- route and travel planning,
- commercial fleet management,
- supervision of dangerous goods transportation,
- and traffic planning.

Project LISB also encompasses investigations concerning the acceptance of such a system by the users. It has revealed that on routes which the road user does not know in advance, acceptance of the guiding informations is very high, on known routes it is high. On 53 percent of the routes taken by private persons guiding hints were always obeyed. On 41 percent of routes taken by business vehicles guiding hints were always obeyed.

Chapter 5

Conclusion

The examples in the first three chapters have revealed that the MAS paradigm, when employed in the urban traffic scenario, leads to an highly integrated system that allows to account for virtually all relevant aspects of traffic control.

Furthermore, as Chapter 4 has pointed out, the adoption of the MAS paradigm does not entail neglect of other approaches to traffic control problems; it rather allows to integrate them. The rationalizations presented in the first three chapters have been presented without referring to the state of the art in traffic control research projects. After looking at the projects presented in Chapter 4, it turned out that the 'MAS view' on traffic control is an obviously natural one that is not only compatible but even complementary to these projects.

Therefore, we may conclude that the MAS approach to the UTS is a promising way to organize a broad variety of already existing and of upcoming technologies according to one common paradigm. MAS research is able to make viable contributions in the domains of communication, cooperation, distributed information processing and control.

Acknowledgements

The concepts and ideas presented in this paper were developed in the DFKI Project KIK-TEAMWARE and in the Esprit II Project No. 5362, IMAGINE. Prototypical implementations of the UTS, as described in an associated paper [9], were part of the conceptual prototyping phase of IMAGINE and served to support the rationalizations presented here.

Thanks to the members of KIK-TEAMWARE and Hans Haugeneder, who contributed to the development of the urban traffic scenarios through fruitful discussions.

Appendix A

Towards an Implementation

This appendix outlines how the agents of the urban traffic scenario can be simplified considerably and can be modeled by means of the concepts of multi-agent systems technology. In contrast to the informal examples in the main part of this report, a more technical description of a basic set of agents is presented here.

In order to keep the example simple, only a restricted set of basic types of agents will be introduced. Compared to real world traffic scenarios the simulation will be be considerably less complex. So, instead of striving for a comprehensive simulation of reality, emphasis will be put on the applicability of the major multi-agent system aspects to the UTS domain. Correspondingly, the following model is based on the rationalizations of the author, and not on exhaustive nature investigation.

Three types of non-mobile agents are used to model a city map, i.e. the non-mobile parts of the scenario: STREETS[1], CROSSROADS (four streets joining) and T-JUNCTIONS (three streets joining). Note that, as opposed to its usual meaning, the term street denotes nothing more than a plain connection between adjacent crossroads and/or t-junctions. CROSSROADS and T-JUNCTIONS model the connection of STREETS with respect to 'flow' of mobile agents through them and at the same time include the functionality of the traffic-lights, i.e. of flow control.

Mobile agents are represented by the general agent type VEHICLE. (Since this paper describes a first approach to a MAS implementation of the UTS, pedestrians are not yet modeled.) These agents are the only auto motive parts in the UTS model. Their purpose is to move along STREETS, pass CROSSROADS and T-JUNCTIONS and thereby bring to life the simulation.[2]

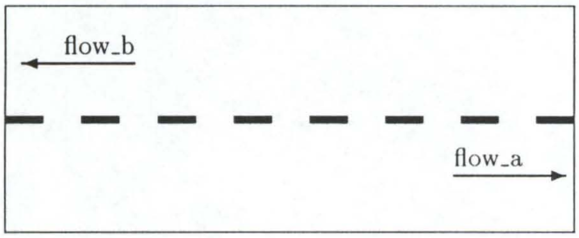
A.1 Street Agents

A STREET agent models the space between consecutive CROSSROADS or T-JUNCTIONS. To keep the examples simple, only streets with a single lane for each direction are considered. Definition 1 illustrates the major properties of a STREET agent.

A STREET is of a given length (*streetlength*). The mean length of a VEHICLE, including a minimal distance to the next VEHICLE, is called *carlength*. Based on these values the maximum *capacity* of one lane of the STREET can be computed as shown in Definition 1. Note that, in reality, the capacity depends on the speeds at which the vehicles are moving since the distances between (human controlled) vehicles must increase with speed. So the capacity of a street is at its maximum when all vehicles stand still and decreases as vehicles increase their speed. To keep the example simple, the capacity is treated as a constant value.

[1] Note the typeface of the word STREET. Whenever a particular type of agent of the model is referred, small capital letters are used. To denote the general meaning of the word normal typeface is used.

[2] In the future this model will be extended in order to include group agents, e.g. heterogeneous groups comprised of STREETS plus CROSSROADS and T-JUNCTIONS and homogeneous groups comprised of PARKING-LOTS which in turn form CAR-PARKS. Furthermore, a sort of register agent is necessary in order to model the process of mobile agents entering or leaving the scene.



Definitions:
let *carlength* be the mean length of a VEHICLE;
let *streetlength* be the length of the STREET;
the STREET has a maximum *capacity* for each lane:

$$capacity := \lfloor streetlength \div carlength \rfloor$$

the flow in a lane can not exceed the capacity:

$$\left. \begin{array}{l} flow\_a \\ flow\_b \end{array} \right\} \leq capacity$$

Definition 1: Properties of a STREET

Therefore, a STREET can be represented simply as a pair of queues.[3] Each queue is capable of holding a certain number of VEHICLES. The maximum queue length is *capacity*. The functionality of each queue is to accept as many VEHICLES as possible from its input-side. On the output-side the draining of VEHICLES is controlled by the next traffic-light. As long as this traffic-light (or whatever else) interrupts the draining away, the queue will not be able to accept more VEHICLES than up to its capacity limit.[4]

A.2 Crossroads and T-Junction Agents

Streets have been modeled in the previous section as bidirectional buffers. They exhibit a rather primitive behavior. Crossroads and t-junctions in conjunction with their associated traffic-lights provide a richer functionality. In particular, these strive to intelligently control the traffic flow. As opposed to the STREETS, traffic-lights and CROSSROADS or T-JUNCTIONS are active agents in the model. To keep the model simple, CROSSROADS or T-JUNCTIONS and their sets of associated traffic-lights are modeled as a single agent.

A.2.1 Crossroads Agent

Definition 2 sketches the main properties of a CROSSROADS agent concerning the aspects of traffic flow. Conforming with the simple model of streets given above, intersecting streets do not fork off right-turn and/or left-turn lanes. This restriction allows to do without traffic-lights that show arrows for the different directions of traffic flow and simplifies light sequence control considerably.

Even in this simplified crossroads model, light sequences of the individual traffic-lights are strongly interdependent. Opposite directions always switch in synchrony. That is, `flow_a_in` and `flow_b_in` are controlled by exactly the same sequence. The same holds for `flow_c_in` and `flow_d_in`. Furthermore these two sequences are complementary to each other: while the a and b lights show green the c and d lights must show red and vice verse.

The main functionality of the CROSSROADS is implemented by its associated set of traffic-lights. Their task is to control the flow of traffic so that all directions are served equally. Based on the simple model sketched above, Algorithm 1 describes the behavior of a set of traffic-lights at a CROSSROADS.

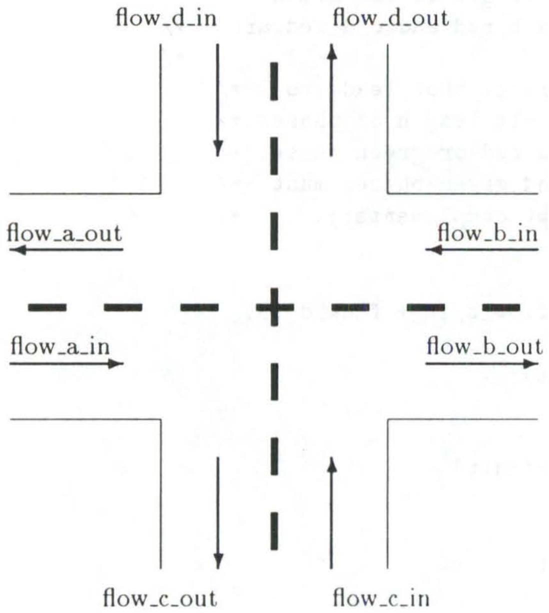
The constant value `default`, which is arbitrarily set to 15 seconds in Algorithm 1, determines the standard length of red and green phases. As long as no significant differences in traffic flow volumes are detected, the traffic-lights adhere to this default value. The constant `threshold` determines how significant the differences in traffic flow volumes must be in order to deviate from default timing.

The algorithm is comprised of an infinite loop. Each iteration starts with gathering up-to-date information about the current traffic volumes. That is, the actual values of `flow_a_in` ... `flow_d_in` have to be determined.[5] Based on these values the difference in flow volumes (`diff`) is calculated. As

[3]Note that in order to keep the example simple, streets have only one lane for each direction; no left-turn or right-turn lanes and no one-way streets are considered.

[4]The simplicity of this model is intentional. It is obvious that a model of a real traffic scenario will be far more complex with respect to the mathematical model describing flows and capacities. In particular, the complex interdependencies between consecutive streets and crossroads (the net-effects) would be of great significance in a more realistic model.

[5]Flows are non-negative integer values.



Definitions:
the sum of all input flows and output flows must be equal to zero:

$$\sum_{\nu=a}^{d}(flow\_\nu\_in - flow\_\nu\_out) = 0;$$

CROSSROADS have no *capacity*.

Definition 2: Properties of a CROSSROADS

long as `diff` does not exceed `threshold`, the traffic-lights adhere to the default timing. Otherwise, the timing is adjusted by prolonging the green phase for the high volumes direction(s). Prolongation is limited to at most three times the `default`.[6]

A.2.2 T-Junction Agent

An important variant of a CROSSROADS is the t-shaped junction of only three roads. Definition 3 sketches the major properties of a T-JUNCTION agent. All remarks concerning a CROSSROADS are valid for the T-JUNCTION agent as well.

In the T-JUNCTION model, light sequences of the individual traffic-lights are strongly interdependent. Opposite directions always switch in synchrony. That is, `flow_a_in` and `flow_b_in` are controlled by exactly the same sequence. This sequence and the sequence controlling `flow_c_in` are complementary to each other: while the a and b lights show green the c lights must show red and vice verse. Algorithm 2 is similar to the previous algorithm. It illustrates the adjustment of light sequences at the T-JUNCTION due to the traffic volumes. Note that the directions a and b are given preference over c assuming that usually the main traffic volumes `are flow_a_in` and/or `flow_b_in`. Therefore, if traffic volume `flow_c_in` raises above the given `threshold` its green phase may be prolongated up to five times its default as opposed to at most three times for the directions a and b.

In a MAS implementation of the traffic scenario each individual junction or crossroad will be tuned by adjusting the default timing of light sequences individually to the typical traffic volumes. Tuning may be done à priori by the implementor or through a continuous learning process.

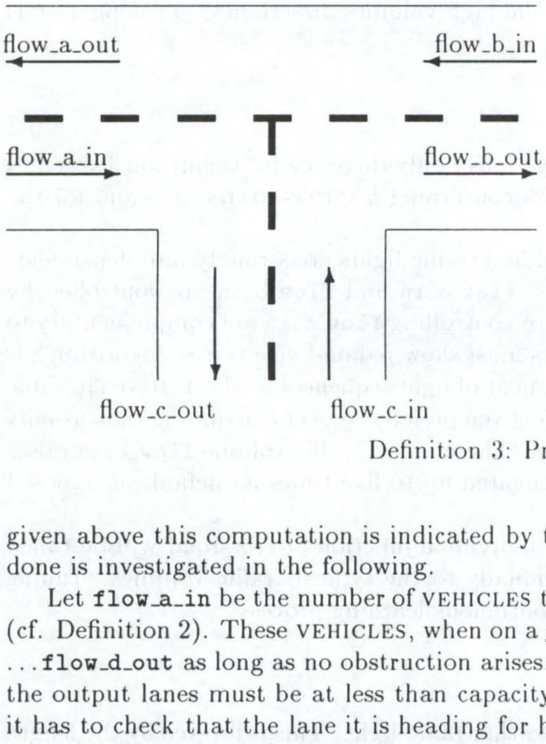
A.2.3 Computing Flow Volumes

During the green phases a certain number of VEHICLES pass the traffic-light. This is limited by the length of the green phase and may be further reduced if the draining away of VEHICLES is obstructed. The actual number of VEHICLES that passed a traffic-light during the last green phase has to be determined in order to compute the (next set of) up-to-date values for `flow_a_in` ... `flow_d_in`. In the algorithms

[6]All values are chosen arbitrarily. In an implementation of the UTS simulation they have to be adjusted appropriately for each individual agent.

```
/* Let a_b_green be the length of the green phase for flow_a_in */
/* and flow_b_in. Let c_d_green be the length of the green      */
/* phase for flow_c_in and flow_d_in. a_b_red and c_d_red are   */
/* the respective red phases.                                   */
threshold := 10  /* Minimal flow difference that leads to      */
                 /* changes of the default length of phases.*/
default := 15    /* Default length of a red or green phase. */
ASSERT (c_d_red == a_b_green) /* Red and green phases must    */
ASSERT (a_b_red == c_d_green) /* be kept complementary.       */
LOOP
   /* Determine current flows. */
   diff := (flow_a_in + flow_b_in) - (flow_c_in + flow_d_in)
   IF (diff > threshold) THEN
      a_b_green += MAX (diff, 3*default)
      c_d_green := default
   ELSEIF (diff < (-threshold)) THEN
      c_d_green += MAX (ABS(diff), 3*default)
      a_b_green := default
   ELSE
      a_b_green := c_d_green := default
   ENDIF
ENDLOOP
```

Algorithm 1: Adjustment of Light Sequences at a CROSSROADS



Definitions:
the sum of all input flows and output flows must be equal to zero:

$$\sum_{\nu=a}^{c}(flow\_\nu\_in - flow\_\nu\_out) = 0;$$

a T-JUNCTION has no *capacity*.

Definition 3: Properties of a T-JUNCTION

given above this computation is indicated by the comment 'determine current flows'. How this can be done is investigated in the following.

Let **flow_a_in** be the number of VEHICLES that comprise the current traffic volume at a CROSSROADS (cf. Definition 2). These VEHICLES, when on a green light, will distribute to the output flows **flow_b_out** ...**flow_d_out** as long as no obstruction arises. Since accidents are ignored here, this requires only that the output lanes must be at less than capacity. Therefore, before a mobile agent enters the crossroads it has to check that the lane it is heading for has enough free space left. Not only does this conform to the traffic regulations, it is also a requirement arising from the definition of the model which states that CROSSROADS and T-JUNCTIONS have no capacity.

To determine the latest flow values requires counting the VEHICLES that left the CROSSROADS or T-JUNCTION through the out flows during the last green phase.

```
/* Let a_b_green be the length of the green phase for flow_a_in    */
/* and flow_b_in. Let c_green be the length of the green phase     */
/* for flow_c_in. a_b_red and c_red are the respective red phases.*/
threshold := 15      /* Minimal flow difference that leads to      */
                     /* changes of the default length of phases.   */
default_a_b := 15    /* Default length of green phase for a and b. */
default_c   := 10    /* Default length of green phase for c.       */
ASSERT (a_b_green == c_red)  /* Red and green phases must be kept */
ASSERT (a_b_red == c_green)  /* complementary.                     */
LOOP
   /* Determine current flows. */
   diff := ((flow_a_in + flow_b_in) / 2) - flow_c_in
   IF (diff > threshold) THEN
      a_b_green += MAX (diff, 3*default_a_b)
      c_green := default_c
   ELSEIF (diff < (-threshold)) THEN
      c_green += MAX (ABS(diff), 5*default_c)
      a_b_green := default_a_b
   ELSE
      a_b_green := default_a_b
      c_green := default_c
   ENDIF
ENDLOOP
```

Algorithm 2: Adjustment of Light Sequences at a T-JUNCTION

For a complete light sequence at a CROSSROADS the computations are as follows:[7]

$$flow\_a\_in' := flow\_a\_in - (flow\_b\_out_a + flow\_c\_out_a + flow\_d\_out_a)$$

$$flow\_b\_in' := flow\_b\_in - (flow\_a\_out_b + flow\_c\_out_b + flow\_d\_out_b)$$

$$flow\_c\_in' := flow\_c\_in - (flow\_a\_out_c + flow\_b\_out_c + flow\_d\_out_c)$$

$$flow\_d\_in' := flow\_d\_in - (flow\_a\_out_d + flow\_b\_out_d + flow\_c\_out_d)$$

And the overall out flow i ($i \in \{a \dots d\}$) during a complete light sequence is the sum of all partial flows originating from all directions but i:

$$flow\_i\_out' := \sum_{\nu=a,\nu\neq i}^{d} flow\_i\_out_\nu$$

For a complete light sequence at a T-JUNCTION computations are similar:

$$flow\_a\_in' := flow\_a\_in - (flow\_b\_out_a + flow\_c\_out_a)$$

$$flow\_b\_in' := flow\_b\_in - (flow\_a\_out_b + flow\_c\_out_b)$$

$$flow\_c\_in' := flow\_c\_in - (flow\_a\_out_c + flow\_b\_out_c)$$

The overall out flow i ($i \in \{a \dots c\}$) during a complete light sequence is the sum of all partial flows originating from all directions but i:

$$flow\_i\_out' := \sum_{\nu=a,\nu\neq i}^{c} flow\_i\_out_\nu$$

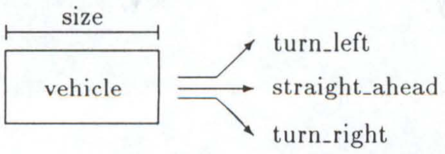
These formulae have to be integrated into the respective algorithms (see above) in order to complete the algorithmic description of the behavior of CROSSROADS and T-JUNCTION agents.

---

[7] The indices of the out flows indicate the source. For example $flow\_a\_out_b$ is the number of VEHICLES that leave the CROSSROADS via lane a coming from b.

A.3 Vehicle Agents

The main action of mobile agents in the UTS is to move around in the scenario. Moving along a street is almost completely modeled by the STREET lanes which organize VEHICLES in queues. The much more interesting event, from the viewpoint of simulating an UTS, is a VEHICLE passing a traffic-light. In the following, the model of mobile agents will focus on this issue.



Definitions:
the actual length of a VEHICLE is an integral multiple of the basic unit *carlength*:

$$size = n * carlength$$

the direction of the VEHICLE's next move is:

$$direction \in \{turn\_left, turn\_right, straight\_ahead\}$$

Definition 4: Properties of a VEHICLE

Definition 4 depicts the major properties of a VEHICLE agent. For queuing up VEHICLES along a lane the length of each VEHICLE is important. To facilitate modeling it will be assumed that all VEHICLES are either of equal length (cf. Definition 1: *carlength*) or an integral multiple thereof. A VEHICLE must communicate its *size* to its host STREET in order to enable the STREET to compute the current load.

```
/* Let SELF be the identification of the mobile agent,      */
/* SIZE its length, S, S1, S2 identifications of streets */
/* and T the identification of a traffic light.           */

/* entering a street S: */
   WHILE NOT capacity_available (S, SIZE) DO
      /* do nothing, wait until street is free */
   ENDWHILE
   register (S, SELF, SIZE)

/* leaving a street S: */
   unregister (S, SELF, SIZE)

/* SELF is the next vehicle on S1 to pass T wanting to */
/* enter street S2. S2 is determined by calling step() */
/* which tells the destination street of the next step */
/* in the plan. So S2 is the street forking off at T   */
/* which is reachable from S1 when making the 'next()' */
/* step of the plan.                                    */
   S2 := step(T, S1, next())
   WHILE green_phase(T, S1) DO
      IF capacity_available (S2, SIZE) THEN
         unregister (S1, SELF, SIZE)
         register (S2, SELF, SIZE)
         tell (T, S1, S2)
         EXIT_WHILE
      ELSE
         /* do not move, i.e. block the lane */
      ENDIF
   ENDWHILE
```

Algorithm 3: Actions of a VEHICLE Agent

Another important property is the direction the VEHICLE wants to take at the next CROSSROADS or

T-JUNCTION. We assume that mobile agents have their given plans determining their route. The plans may be generated in advance or may evolve dynamically. Guided by its plan, the VEHICLE finds its way through a town. An enumeration function **next** (c.f. Algorithm 3) maps the plan into its single steps. Upon each call it returns the direction of the next step in turn. So, for the mobile agent to find its way, it has to call **next** at each CROSSROADS or T-JUNCTION and, when given way, turn to the direction determined by **next**.

Algorithm 3 lists three main functions of a mobile agent. The first one is called whenever the agent wants to enter a street. It then tries to register with the STREET agent **S** by telling its identification **SELF** and its **SIZE**. If the STREET agent is used up to its capacity (filled with VEHICLES) the VEHICLE is caused to wait until enough space becomes available.

The complementary function is to leave a STREET which means the leaving VEHICLE un-registers with the STREET agent and frees the space it had occupied.

The most complex action sequence models the passing of a traffic-light which is a combination of leaving one STREET (S1) and entering another (S2). This action sequence is initially triggered when the VEHICLE becomes the next one to pass the traffic-light T (i.e. is the first one in the S1 queue) and stays active during the green phase for S1. To determine which of the two or three STREETS reachable from S1 will be entered next, the VEHICLE's plan has to be interrogated. Applying the function **next** to the route plan tells which direction to take. So, being in street S1 at traffic-light T and knowing the direction of the next step suffices to determine which STREET to enter next. Let this STREET be S2. As described above, this succeeds only if enough space is available at S2. Then the VEHICLE leaves S1, enters S2 and notifies the traffic-light which way it took. Otherwise the VEHICLE blocks its lane until S2 has enough free capacity. Recall that CROSSROADS or T-JUNCTIONS have no *capacity*.

Explicitly notifying the traffic-light is required, since this enables it to count in and out flows and compute the current flows at the beginning of the next light sequence (cf. Section A.2.3).

A.4 Conclusion

Thus we have seen how the processes of the various agents involved in the UTS may be modelled independently of each other, yet interact cooperatively to guarantee an efficient and smooth flow of traffic.

The first prototypical urban traffic scenarios have already been implemented and not only show feasibility of our approaches but promises to completely fulfill our expectations [9].

Bibliography

[1] Scientific American, Special Issue: *Communications, Computers and Networks*, Sept. 1991.

[2] A. H. Bond, L. Gasser: *Readings in Distributed Artificial Intelligence*, Morgan Kaufman Publishers, San Mateo, CA., 1988.

[3] C. Rich, R. C. Waters (eds.): *Artificial Intelligence and Software Engineering*, Morgan Kaufman Publishers, San Mateo, CA., 1986.

[4] U. Franke: *PROMETHEUS: Wissensbasierte Systeme eroeffnen neue Perspektiven im Strassenverkehr*. in: W. Brauer, C. Freska (eds.): Wissensbasierte Systeme, 3. Internationaler GI-Kongress München, Springer Verlag, 1989, pp. 363-376.

[5] Forschungsgesellschaft fuer Straßen- und Verkehrswesen, Köln: *Verkehrsbeeinflussung*, Kolloquium, 25. Januar, 1990, Bonn.

[6] H.H. Topp (Hrsg.): *Verkehr aktuell: CATS Computer-Aided Traffic Systems* Vortragsreihe Wintersemester 1990/91, Fachgebiet Verkehrswesen, Universität Kaiserslautern

[7] A. Lux, F. Bomarius, D. Steiner: *A Model for Supporting Human Computer Cooperation*, AAAI Workshop on Cooperation among Heterogeneous Intelligent Systems, San Jose, CA. July 14, 1992.

[8] D. Steiner, D. Mahling, H. Haugeneder: *Human Computer Cooperative Work*, in: Proceedings of the 10th International Workshop on Distributed Artificial Intelligence, Bandera, Texas, Oct. 23-27, 1990.

[9] F. Bomarius: *Using the MECCA System to implement Urban Traffic Scenarios*, DFKI Technical Memo TM-92-07, September 1992.

Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

DFKI
-Bibliothek-
PF 2080
D-6750 Kaiserslautern
FRG

DFKI Publikationen

Die folgenden DFKI Veröffentlichungen sowie die aktuelle Liste von allen bisher erschienenen Publikationen können von der oben angegebenen Adresse bezogen werden.
Die Berichte werden, wenn nicht anders gekennzeichnet, kostenlos abgegeben.

DFKI Publications

The following DFKI publications or the list of all publisched papers so far can be ordered from the above address.
The reports are distributed free of charge except if otherwise indicated.

DFKI Research Reports

RR-91-33
Franz Baader, Klaus Schulz:
Unification in the Union of Disjoint Equational Theories: Combining Decision Procedures
33 pages

RR-91-34
Bernhard Nebel, Christer Bäckström:
On the Computational Complexity of Temporal Projection and some related Problems
35 pages

RR-91-35
Winfried Graf, Wolfgang Maaß: Constraint-basierte Verarbeitung graphischen Wissens
14 Seiten

RR-92-01
Werner Nutt: Unification in Monoidal Theories is Solving Linear Equations over Semirings
57 pages

RR-92-02
Andreas Dengel, Rainer Bleisinger, Rainer Hoch, Frank Hönes, Frank Fein, Michael Malburg:
Π_{ODA}: The Paper Interface to ODA
53 pages

RR-92-03
Harold Boley:
Extended Logic-plus-Functional Programming
28 pages

RR-92-04
John Nerbonne: Feature-Based Lexicons: An Example and a Comparison to DATR
15 pages

RR-92-05
Ansgar Bernardi, Christoph Klauck, Ralf Legleitner, Michael Schulte, Rainer Stark:
Feature based Integration of CAD and CAPP
19 pages

RR-92-06
Achim Schupetea: Main Topics of DAI: A Review
38 pages

RR-92-07
Michael Beetz:
Decision-theoretic Transformational Planning
22 pages

RR-92-08
Gabriele Merziger: Approaches to Abductive Reasoning - An Overview -
46 pages

RR-92-09
Winfried Graf, Markus A. Thies:
Perspektiven zur Kombination von automatischem Animationsdesign und planbasierter Hilfe
15 Seiten

RR-92-10
M. Bauer: An Interval-based Temporal Logic in a Multivalued Setting
17 pages

RR-92-11
Susane Biundo, Dietmar Dengler, Jana Koehler:
Deductive Planning and Plan Reuse in a Command Language Environment
13 pages

RR-92-13
Markus A. Thies, Frank Berger:
Planbasierte graphische Hilfe in objektorientierten Benutzungsoberflächen
13 Seiten

RR-92-14
Intelligent User Support in Graphical User Interfaces:
1. InCome: A System to Navigate through Interactions and Plans
Thomas Fehrle, Markus A. Thies
2. Plan-Based Graphical Help in Object-Oriented User Interfaces
Markus A. Thies, Frank Berger
22 pages

RR-92-15
Winfried Graf: Constraint-Based Graphical Layout of Multimodal Presentations
23 pages

RR-92-16
Jochen Heinsohn, Daniel Kudenko, Berhard Nebel, Hans-Jürgen Profitlich: An Empirical Analysis of Terminological Representation Systems
38 pages

RR-92-17
Hassan Aït-Kaci, Andreas Podelski, Gert Smolka: A Feature-based Constraint System for Logic Programming with Entailment
23 pages

RR-92-18
John Nerbonne: Constraint-Based Semantics
21 pages

RR-92-19
Ralf Legleitner, Ansgar Bernardi, Christoph Klauck: PIM: Planning In Manufacturing using Skeletal Plans and Features
17 pages

RR-92-20
John Nerbonne: Representing Grammar, Meaning and Knowledge
18 pages

RR-92-21
Jörg-Peter Mohren, Jürgen Müller
Representing Spatial Relations (Part II) -The Geometrical Approach
25 pages

RR-92-22
Jörg Würtz: Unifying Cycles
24 pages

RR-92-23
Gert Smolka, Ralf Treinen:
Records for Logic Programming
38 pages

RR-92-24
Gabriele Schmidt: Knowledge Acquisition from Text in a Complex Domain
20 pages

RR-92-25
Franz Schmalhofer, Ralf Bergmann, Otto Kühn, Gabriele Schmidt: Using integrated knowledge acquisition to prepare sophisticated expert plans for their re-use in novel situations
12 pages

RR-92-26
Franz Schmalhofer, Thomas Reinartz, Bidjan Tschaitschian: Intelligent documentation as a catalyst for developing cooperative knowledge-based systems
16 pages

RR-92-27
Franz Schmalhofer, Jörg Thoben: The model-based construction of a case-oriented expert system
18 pages

RR-92-29
Zhaohur Wu, Ansgar Bernardi, Christoph Klauck: Skeletel Plans Reuse: A Restricted Conceptual Graph Classification Approach
13 pages

RR-92-30
Rolf Backofen, Gert Smolka
A Complete and Recursive Feature Theory
32 pages

RR-92-31
Wolfgang Wahlster
Automatic Design of Multimodal Presentations
17 pages

RR-92-33
Franz Baader:
Unification Theory
22 pages

RR-92-34
Philipp Hanschke:
Terminological Reasoning and Partial Inductive Definitions
23 pages

RR-92-35
Manfred Meyer:
Using Hierarchical Constraint Satisfaction for Lathe-Tool Selection in a CIM Environment
18 pages

RR-92-36
Franz Baader, Philipp Hanschke:
Extensions of Concept Languages for a Mechanical Engineering Application
15 pages

RR-92-37
Philipp Hanschke: Specifying Role Interaction in Concept Languages
26 pages

RR-92-38
Philipp Hanschke, Manfred Meyer:
An Alternative to Θ-Subsumption Based on Terminological Reasoning
9 pages

RR-92-41
Andreas Lux: A Multi-Agent Approach towards Group Scheduling
32 pages

RR-92-42
John Nerbonne:
A Feature-Based Syntax/Semantics Interface
19 pages

RR-92-43
Christoph Klauck, Jakob Mauss: A Heuristic driven Parser for Attributed Node Labeled Graph Grammars and its Application to Feature Recognition in CIM
17 pages

RR-92-44
Thomas Rist, Elisabeth André: Incorporating Graphics Design and Realization into the Multimodal Presentation System WIP
15 pages

RR-92-45
Elisabeth André, Thomas Rist: The Design of Illustrated Documents as a Planning Task
21 pages

RR-92-46
Elisabeth André, Wolfgang Finkler, Winfried Graf, Thomas Rist, Anne Schauder, Wolfgang Wahlster: WIP: The Automatic Synthesis of Multimodal Presentations
19 pages

RR-92-47
Frank Bomarius: A Multi-Agent Approach towards Modeling Urban Traffic Scenarios
24 pages

RR-92-48
Bernhard Nebel, Jana Koehler:
Plan Modifications versus Plan Generation: A Complexity-Theoretic Perspective
15 pages

RR-92-51
Hans-Jürgen Bürckert, Werner Nutt:
On Abduction and Answer Generation through Constrained Resolution
20 pages

RR-92-54
Harold Boley:
A Direkt Semantic Characterization of RELFUN
30 pages

DFKI Technical Memos

TM-91-13
Knut Hinkelmann:
Forward Logic Evaluation: Developing a Compiler from a Partially Evaluated Meta Interpreter
16 pages

TM-91-14
Rainer Bleisinger, Rainer Hoch, Andreas Dengel:
ODA-based modeling for document analysis
14 pages

TM-91-15
Stefan Bussmann: Prototypical Concept Formation An Alternative Approach to Knowledge Representation
28 pages

TM-92-01
Lijuan Zhang:
Entwurf und Implementierung eines Compilers zur Transformation von Werkstückrepräsentationen
34 Seiten

TM-92-02
Achim Schupeta: Organizing Communication and Introspection in a Multi-Agent Blocksworld
32 pages

TM-92-03
Mona Singh
A Cognitiv Analysis of Event Structure
21 pages

TM-92-04
Jürgen Müller, Jörg Müller, Markus Pischel, Ralf Scheidhauer:
On the Representation of Temporal Knowledge
61 pages

TM-92-05
Franz Schmalhofer, Christoph Globig, Jörg Thoben
The refitting of plans by a human expert
10 pages

TM-92-06
Otto Kühn, Franz Schmalhofer: Hierarchical skeletal plan refinement: Task- and inference structures
14 pages

TM-92-08
Anne Kilger: Realization of Tree Adjoining Grammars with Unification
27pages

DFKI Documents

D-92-06
Hans Werner Höper: Systematik zur Beschreibung von Werkstücken in der Terminologie der Featuresprache
392 Seiten

D-92-07
Susanne Biundo, Franz Schmalhofer (Eds.): Proceedings of the DFKI Workshop on Planning
65 pages

D-92-08
Jochen Heinsohn, Bernhard Hollunder (Eds.): DFKI Workshop on Taxonomic Reasoning Proceedings
56 pages

D-92-09
Gernod P. Laufkötter: Implementierungsmöglichkeiten der integrativen Wissensakquisitionsmethode des ARC-TEC-Projektes
86 Seiten

D-92-10
Jakob Mauss: Ein heuristisch gesteuerter Chart-Parser für attributierte Graph-Grammatiken
87 Seiten

D-92-11
Kerstin Becker: Möglichkeiten der Wissensmodel-lierung für technische Diagnose-Expertensysteme
92 Seiten

D-92-12
Otto Kühn, Franz Schmalhofer, Gabriele Schmidt: Integrated Knowledge Acquisition for Lathe Production Planning: a Picture Gallery (Integrierte Wissensakquisition zur Fertigungsplanung für Drehteile: eine Bildergalerie)
27 pages

D-92-13
Holger Peine: An Investigation of the Applicability of Terminological Reasoning to Application-Independent Software-Analysis
55 pages

D-92-14
Johannes Schwagereit: Integration von Graph-Grammatiken und Taxonomien zur Repräsentation von Features in CIM
98 Seiten

D-92-15
DFKI Wissenschaftlich-Technischer Jahresbericht 1991
130 Seiten

D-92-16
Judith Engelkamp (Hrsg.): Verzeichnis von Softwarekomponenten für natürlichsprachliche Systeme
189 Seiten

D-92-17
Elisabeth André, Robin Cohen, Winfried Graf, Bob Kass, Cécile Paris, Wolfgang Wahlster (Eds.): UM92: Third International Workshop on User Modeling, Proceedings
254 pages
Note: This document is available only for a nominal charge of 25 DM (or 15 US-$).

D-92-18
Klaus Becker: Verfahren der automatisierten Diagnose technischer Systeme
109 Seiten

D-92-19
Stefan Dittrich, Rainer Hoch: Automatische, Deskriptor-basierte Unterstützung der Dokumentanalyse zur Fokussierung und Klassifizierung von Geschäftsbriefen
107 Seiten

D-92-21
Anne Schauder: Incremental Syntactic Generation of Natural Language with Tree Adjoining Grammars
57 pages

D-92-23
Michael Herfert: Parsen und Generieren der Prolog-artigen Syntax von RELFUN
51 Seiten

D-92-24
Jürgen Müller, Donald Steiner (Hrsg.): Kooperierende Agenten
78 Seiten

D-92-25
Martin Buchheit: Klassische Kommunikations- und Koordinationsmodelle
31 Seiten

D-92-26
Enno Tolzmann:
Realisierung eines Werkzeugauswahlmoduls mit Hilfe des Constraint-Systems CONTAX
28 Seiten

D-92-27
Martin Harm, Knut Hinkelmann, Thomas Labisch: Integrating Top-down and Bottom-up Reasoning in COLAB
40 pages

D-92-28
Klaus-Peter Gores, Rainer Bleisinger: Ein Modell zur Repräsentation von Nachrichtentypen
56 Seiten

A Multi-Agent Approach towards Modeling Urban Traffic Scenarios

Frank Bomarius

RR-92-47

Research Report