Agent-based coordination model for designing transportation applications

Fabien Badeig\textsuperscript{1,2}, Flavien Balbo\textsuperscript{1,2}, Gérard Scemama\textsuperscript{1}, Mahdi Zargayouna\textsuperscript{1,2}

\textsuperscript{1} INRETS - GRETIA team, avenue du Général Malleret-Joinville, F-94114 Arcueil Cedex, France.
\textsuperscript{2} University Paris-Dauphine - LAMSADE, Place du Maréchal de Lattre de Tassigny, F-75775 Paris 16 Cedex, France.
\{badeig,zargayou,scemama\}@inrets.fr balbo@lamsade.dauphine.fr

Abstract—This paper presents an environment-centered approach to design multi-agent solutions to transportation problems. Based on the Property-based Coordination Principle (PbC), the objective of our approach is to solve three recurrent issues in the design of these solutions: the knowledge problem, the space-time dimension and the dynamics of the real environment. To demonstrate the benefits of our approach, two completely different applications, a demand-responsive transportation system and a simulator for crisis management, both based on this principle are presented. For each of them, we show how these recurrent issues have been solved.

I. INTRODUCTION

Several researches in the multi-agent community have been done in the transportation domain. The multi-agent approach deals with systems consisting of many physically and/or logically distributed interacting components that possess some level of autonomy. These components are able to perceive their environment and to react to changes in that environment as well, in accordance to their goals. Making a strong use of bottom-up paradigm as an approach to system design, makes it easier to comprehend a complex reality, by the reification of the components of the system to manage. In the transportation domain, the objective of many applications is the management of distributed entities and in this context, the multi-agent approach simplifies an approach by analogy.

In our previous developments of transportation applications [2], [13], we have identified three recurrent issues. The first issue is related to the knowledge processing. In many problems such as urban network regulation [2], the knowledge is uncomplete or is only owned by human experts. The knowledge issue is also related to the coordination of many information sources like in the traffic regulation problem [8]. The second issue is the space-time dimension of the problems. For instance, the traveler information systems have to provide the right information and/or services at the right time and the right place [9], [13]. The last issue is the dynamics of the real environment, which impacts the quality of the information, makes direct communications difficult and costly and/or implies that mobile entities appear and disappear (e.g. Automatic Guided Vehicles management [12]).

In this paper, we argue that the design of a multi-agent environment compliant to the Property based Coordination (PbC) principle is a solution to these issues. PbC promotes an effective separation between an observable description of the abstract and real MAS components and the use by the agents of these descriptions to coordinate them. The multi-agent environment contains the recorded descriptions and supports their processing. We describe two applications that are based on an environment modeling that is compliant to the PbC principle. For each of them, we have focused on the solutions to the transportation issues defined above.

The remainder of this paper is organized as follows: section II presents the role of the environment in the design of transportation applications and how it is a support for the PbC principle. Section III details a demand-responsive transportation system. Section IV describes a simulator for crisis management. We finally draw general conclusions and perspectives in section V.

II. ENVIRONMENT COMPLIANT TO THE PbC PRINCIPLE

A. Environment role

The environment has responsibilities [11] that have successfully been applied in the design of transportation applications. The environment being a shared space for the agents, resources and services, its first responsibility is the structuring of the MAS. The environment modeling is a solution to give a space-time referential to a transportation application. For example, the transportation intelligent information systems or more generally location-based services exploit the information about the location of the mobile users in order to adapt their processing. There is a system which has to manage the relation between the mobile user’s location and the local services. In the multi-agent solutions, this task can be achieved by the environment. The environment either centralizes the user agent locations and matches against the available information/services [9] or is distributed and gives a homogeneous support to the service design [6]. Moreover, its privileged intermediary role makes the environment a good candidate to support spatially and temporally decoupled coordination models and therefore it is appropriate for applications taking into account dynamic real environment. In [12], the environment contains fields that are propagated in the environment in a certain range and used by the agents to organize the task assignment for Automatic Guided Vehicles. The MAS environment becomes the common referential that enables agents to adapt their behavior according to the dynamics of the real environment.

The second environment responsibility is to maintain its own dynamics. Following its structuring responsibility, it
can also manage the dynamics of the real transportation environment, ensuring the coherence of the MAS. In [12] for instance, the environment ensures the propagation of the fields. Moreover, the environment can ensure services that are not at the agent level or simplify the agent design. In a traffic light control system [3], the environment, that has a global point of view, gives rewards or penalties to self-interested agents according to their local decision.

Because the environment with its own dynamics can control the shared space, its third responsibility is to define rules for the multi-agent system. In a bus network simulation [7], the main role of the environment is to constrain perceptions and interactions of agents. For transportation applications that have an incomplete knowledge, it simplifies the design of the MAS by a clear separation between the roles of the agents and their organization. In the coordinated monitoring of traffic jams application [5], the environment provides organizations, which dynamically evolves according to the current context.

Because the agents are "users" of the services of the environment and in order to really create a common knowledge, the last responsibility of the environment is to make observable and accessible its own structure.

### B. Property based Coordination principle

The Property based Coordination (PbC) principle is based on the observability of the environment. We propose the following definition for PbC: *The Property-based Coordination principle is to design multi-agent components by observable Symbolic Components (SC) and to manage their processing for coordination purposes.*

There are two categories of SC. The first category is the symbolic description (SD) of a real component of the multi-agent system: agent, message, or object. The descriptions are symbolic because they are a representation of the real components. The data structure that has been chosen to give the symbolic description is a set of property-value pairs. For example, an agent has its own process and knowledge, and a SD for it is recorded in the environment. Only the descriptions are observable and a control can be applied by the environment. For example, in the figure 1, the environment contains the SD of two agents, two messages, and three objects.

The second category is related to the abstract MAS components and especially to the coordination component. The objective is to manage how the agents link the environment content to their behavior. Each link represents a need of an agent or a category of agents and is reified by a Selective Object (SO) (five SO, figure 1). A SO is a set of constraints on the properties of the SD that defines an agent need. For example, if the SD of messages or vehicles are recorded in the environment, the following needs could be represented: “I need to send a message to the fastest vehicle” or “when I’m close to a vehicle, I need to slow down”.

The management of the SC processing deals with the interaction between the agents and the environment. It takes into account the modification environment (e.g., the SD update process) done by the agents and how the agents link the environment content to their behavior. The same SO can be used by several agents if they have the same need, in figure 1, SO1 is used for the agents A and B (alternatives 1, 2). These SO are added (or removed) by the agents or by the designer of the environment. The dynamic of the SO management ensures the autonomy of the agents. The agents adapts their interaction with the environment by their management of their SO. The common process is a matching algorithm that matches SO against the SD. When a SO is triggered, this activates an action (message reception, agent activation (simulation), object perception) for an agent to perform. The figure 1 gives some examples of combinations of the descriptions. Thanks to the action that is related to the SO, different coordination models can be simultaneously supported. For example, if the coordination is based on a communication model, then the messages have a SD, the SO reifies the interests of the agents for them and the matching process applied by the environment gives for each agents the messages that are related to it, figure 1 alternatives 1-4.

The context is the state of the environment that is composed of the SD and the behavior of the agents is adapted according to this information. In that way, the reification by the SO of the agent needs is the starting point for contextual processing of the agent behavior. Agents condition their behavior according to their current context including their own state. Figure 1, the alternative 6 is an example of contextual activation that takes into account the state of the agents A, B and the object O1.

To be compliant to the PbC principle, the environment has to give at least the services for the SC management and a matching algorithm. Their are no assumptions on the modeling of the SC and the matching algorithm. The two following transportation applications are based on an environment that is compliant to the PbC principle.

### III. COORDINATION ENVIRONMENT FOR DEMAND-RESPONSIVE TRANSPORTATION SYSTEMS

We have proposed a demand-responsive transportation system (DRTS) as a MAS in which the coordination of agents’ activities is fulfilled through the environment, following the PbC principle.

#### A. Demand-Responsive Transportation Systems

A DRTS is a system designed to answer online customers that desire to be transported from a point in the network to
another one. Customers specify a time window associated with each point (departure and arrival) inside which they want to be visited. The criteria to evaluate the efficiency of a DRTS is, first, to minimize the number of used vehicles to serve all the customers, and then to minimize of the total routes’ lengths.

A DRTS has to deal with the three issues given in the introduction in that way. First, The knowledge issue is related to the fact that all the customers are not known before the start of the system execution. In our system, the available information are put in common in the environment as soon as they become available. Second, the space-time dimension of the problem is naturally important, since the space-time distribution of the customers and the mean width of their time windows condition the behavior of the vehicles and their coordination, and determine the quality of the solutions offered by the system. The environment in our application structures the MAS components temporally and spatially, so that the interaction between agents is guided by their perception on it. The interaction between customers and vehicles is guided by their space-time positions, and the environment is modeled to allow it. The last issue is the dynamics of the environment. Indeed, when modeled as a MAS, DRTS are open MAS, since agents (e.g. customers and vehicles) join and leave the system freely. In such a dynamic environment, limiting the communications is very important, since the broadcast of all the available information is very costly. We use a coordination model adhering to PbC, which allows to palliate the loss in information in dynamic environments. More precisely, the MAS environment is a persistent data repository, in which data is maintained by the context, the use of common shared environment compliant PbC, helps us devising a new method to materialize the VAs that could be interested by its insertion, without knowing them a priori, and on the other hand it circumscribes the communication in the system to the only agents that can reach an agreement (an insertion in the route of a vehicle). It is worth noting that the VAs that don’t perceive a CA can use their time to be candidates for the insertion of other customers.

The protocol followed in the MAS is a negotiation mechanism between CAs and VAs. When a new customer connects to the system, a CA is created, and is perceived by the available VAs (that is, which are not already involved in the insertion of another customer). Each VA computes an insertion price for the insertion of this customer, and proposes it to the CA. The CA finally chooses the VA proposing the lowest price. This protocol is a distributed version of the well-known insertion heuristics [10], which are the fastest heuristics since an insertion decision is not reconsidered afterwards. However, these heuristics are known to be myopic [4], because the computed price is generally function of the additional distance for the vehicle resulting of the insertion of the considered customer. The future is then not taken into account in the price calculation. We choose to keep the negotiation protocol, but we propose a new method to compute the insertion price of a new customer.

C. Insertion price calculation

Instead of focusing on the distance traveled by the VAs in order to compute the insertion price of a customer, we choose to focus on the future availability of the VAs. In this context, the use of common shared environment compliant to PbC, helps us devising a new method to materialize the VAs’ availability. Indeed, the MAS environment in a DRTS can be modeled as a space-time network, in which there are not simply nodes, but node-time pairs. We call a perception
field of a VA the space-time nodes that it can visit: a set of nodes it can visit together with the corresponding times, given the customers already inserted in the route of the VA. The insertion of a new customer in the route of a VA would imply that there will be several nodes that it wouldn’t be able to visit anymore, at least during a certain period. These would be as many chances to participate in future insertion negotiations that it would loose. As a consequence, the more the perception field of a VA is wide, the more it has chances to be a candidate for future customers’ insertion (in the absence of other information, we assume a uniform distribution of customers).

We illustrate the notion of a perception field in the figure 3 with an Euclidean problem, where the network is a plane. The environment of the MAS is illustrated as a cube: it represents all the virtually possible space-time positions of all the vehicles in the network. The cone in the left figure represents the perception field of a VA located in \((x_0, y_0)\) at the moment \(e_0\). This VA cannot be a candidate for customers that have a departure or an arrival node with a corresponding time that is outside the cone (because, even if the vehicle leaves at \(e_0\), it won’t be able to serve it, given the Euclidean metric). In the right figure, we illustrate the effect, on the perception field of a VA, of the insertion of a node in its route. All the space-time zone that is outside the bold shape and that is inside the initial cone, is the loss in the perception of the VA, resulting from the insertion of this node.

In the negotiation process, the price that is proposed to the CA is the difference between the old perception field (i.e. before the insertion) and the new one (i.e. after the insertion). The main goal of the negotiation process becomes the optimization of the perception fields of the agents, with an objective: the minimization of the number of mobilized vehicles to service all the customers. In [14], we report preliminary results that show that the use of this protocol behaves better than the traditional insertion heuristics with respect to the number of used vehicles, but traditional heuristics behave better with respect to the total distance traveled by all the vehicles. In addition, the experiments demonstrate the relevance of the use of the environment in the limitation of the communication costs, since the gain in terms of exchanged messages is more than proportional to the number of customers connected to the system. That means that the more agents we have in the MAS, the more we save network costs with respect to a solution based on a broadcast of all the customers to all the vehicles.

In this application, the use of MAS environment as a materialization of the PbC principle has a twofold benefit. On the one hand, it structures the agents interaction and coordination, and makes it more efficient to interact in a dynamic environment, where agents appear, disappear without maintaining a knowledge about others and where communications can be disturbed, costly etc. On the other hand, it allows for the definition of a new objective: the optimization of agents’ perception fields on the environment, that showed to be relevant in the context of DRTS, and is meant to be generalized to certain categories of transportation applications to be defined.

IV. INTERACTIONAL ENVIRONMENT FOR CRISIS MANAGEMENT

In this section, we discuss how the environment-centered approach compliant to the PbC principle enables to solve the difficulties presented above in the case of agent-based simulation of transportation crisis.

A. The crisis context

The knowledge related to the management of a crisis is incomplete, because each crisis situation is complex and singular. Indeed, a crisis situation is a dynamic phenomenon defined by the initial situation that depends on the place and of the time, and by the impacts on the population and on the infrastructures. To deal with the traffic crisis situations, traffic management plans are developed. One plan defines the set of the involved agencies and describes the actions protocol for each agency. The efficiency of a plan depends on the organizations efficiency of the agencies involved in the transportation crisis. Our proposal is to design a decision support system for the evaluation and validation of the traffic management plan.

One issue is that the agencies have their own communication and action processes depending on their own goals. An agency respects a local hierarchical organization that constrains its interaction protocol. Consequently, an agency can be seen as a knowledge source with its own organization and an efficient global organization has to be found. An agency is composed of decisional and operational stakeholders. The decisional stakeholders adapt the decisions process to the perception of the crisis situation and the operational ones apply the actions involved by the decisions process. The operational level gives the informations about the situation to enable the decisional level to build a consistent perception of the situation. According to its agency, its role and its goal, the stakeholders have a selective perception of their environment. Indeed the crisis perception of a stakeholder is goal-oriented.

From a multi-agent point of view, an agent corresponds to a stakeholders and the two functional kinds of stakeholders mean that the simulation has to take into account the cognitive agents and the reactive agent in the same simulation. Following the PbC principle, each agent keep updated its symbolic description (SD) in the environment, and its perception is ensured thanks to the selective objects (SO). A detailed description of the symbolic components SC...
specifications is available in [1]. PbC is used to reproduce the partial perception of the real environment.

The knowledge of the crisis situation is split up in the time, in the space and between the agencies. Consequently, the difficulty is to represent its space-time dimension. By definition of an agent, the environment is a common space and naturally, the environment becomes the space-time referential. About the space referential, the reactive agents are located in the spatial environment and the environment ensures the validity of the agent actions. About the time referential, the simulator needs a global clock to synchronize the agents and the environment ensures the processing of the global time. Consequently, the space-time information are shared between the components of the simulation and they appear in the SD of the agents. Thus, they are used to compute the context of an agent and to constraint communication and action of the agents in the definition of the SO. For instance, an agent can add a SO that matches the agent context against the SD of a resource in order to perform an action, where the context expresses the constraint about the proximity of the resource to the agent.

The last difficulty is to simulate the dynamic of the real environment in which a crisis evolves and which impacts the stakeholders behaviors. The environment is used to constraint the behavior of the agents. Then, the environment has to manage the simulation process which gathers the services do not depend on the agents. Thus, the environment applies the simulation scenario that is tested (see next section), and thus the scenario can impact the interactional process. For instance, the designer can specify a failure probability of a communication canal. Consequently, the designer can easily modify the simulation behavior in playing with the scenario definition.

B. Functional architecture of the simulator

In this section, we give a functional architecture of the crisis simulator that has been developed. Figure 4 depicts the functional architecture of the crisis simulator where the environment is the space-time referential and ensures the management of the interaction and execution process.

Our simulator is composed of the simulation components (agents, resources and messages), of the environment and of four modules. We identify the cognitive agent (decisional level) and the reactive agent (operational level). The agents are linked to the environment because they participate to the interaction process, and for the reactive agent the environment is the space referential with the graph of the transport network. The resources corresponds to the safety equipments, to the safety vehicles and to the infrastructures. The four modules are related to the processing of the simulation execution (Life Cycle Module (LCM)), to the processing of the traffic management plan (Traffic Management Plan Module (TMPM)), to the communication with a external data source (Interface Module (IM)) and to the observation of the simulation variables (Observation Module (OM)).

Moreover, the simulator parameters are defined by two inputs and the results of the simulation are given by one output. The first input corresponds to the traffic management plan (TMP) and it is loaded in the TMPM. The second input enables to initialize the simulator. It defines the initial situation (transportation network, traffic, ...) and the probabilities about the performed actions and communications. The output provides a simulation history with the observation of the interactions defined by the designer. Indeed, the evaluation criteria defined by the designer in the initialization file specifies the output results.

A simulation process is controlled by a life cycle that corresponds to an execution scenario. LCM defines the simulation parameter and it is linked to the environment to initialize the variables of the execution process. It is initialized by the input initialization file. TMPM manages the traduction of the traffic management plan for the cognitive agent corresponding to the state officer. The state officer is the responsable of the decisionnal level. This module represents a data source for the state officer. Thus, the state officer adapts the decision process according to the intervention plan and the crisis situation perception. The module is initialized by the input file TMP. IM corresponds to an interface between the the simulation platform and a external data source. Its role is to communicate and synchronize the spatial environment (i.e. the graph of transport network) with a data source. Thus, the transport network representation in the simulator is relevant and the traffic flow is realist. So our crisis simulator communicates with a macroscopic traffic simulator MAGISTER. MAGISTER is a traffic simulation tool built-up by the the french National Institute for Transport and Safety Research (INRETS). It implements several traffic flow models : : LW, LW, MET, ARZ.

The data transfer between the interface module of the crisis simulator and the simulator MAGISTER is ensured by the protocol TCP (Transmission Control Protocol). Then the module updates the traffic flow on the transport network of the spatial environment. OM is directly connected to the environment. It defines the variables that have to be observed, and the environment send it the change of the variables. Consequently, this module recovers that data about the scenario execution for a specific crisis situation, and a expert can analyze the results to study a specific point of the simulation.

C. Simulation model specifications

The agent interactions are directed by its local context, which need a local context computing analysis. This com-
puting analysis complicates the modification of the agent behavior and the passing from the designing to the implementation. Our objective is to separate the design process to the simulation process in analyzing the agents context inside the environment. Thus, the environment contains the interaction and simulation SO that are the link between a local agent context and a agent action. The interactions processing is unified and enables to situate easily in the same simulation cognitive agents and reactive agents. Moreover, the interactions processing inside the environment provide a new way to parameterize the simulation behavior.

The environment is the temporal reference for the simulation agents and each agent has its internal time. The environment time is discrete and is the global time of the simulation. At a given time cycle, the simulator ensures that the agents which are ready to act are activated; this is done only once per time cycle. This control is guaranteed by the environment thanks to the comparison of the global time with the internal time of the agent in each activation SO.

Therefore, the internal time of agents must be observable which implies the addition of the observable property time to the SO of the agent. The activation of the agents depends on the SO inside the environment. In order to ensure a default activation, a default SO is systematically added to the environment and its triggering depends on the comparison between the global time and the internal time of the agent. This modeling offers two advantages: (1) an agent can choose to be inactive for a period in the simulation, thus enabling a gain in run-time; (2) the same agent cannot be activated more than once in the same time cycle. Moreover, it can facilitate complex interaction protocol like overhearing. In that case, SO are added in the environment to enable the agents to receive messages that are send by agents close of their position. Based on the PbC principle, the coordination model unifies the framework of communication and activation to support the design of complex multi-agent simulations, and it facilitates the change of simulation parameters.

The activity of an agent is modeled using an behaviors automaton. Each state of the automaton is a reference to a behavior which is a coherent sequence of actions. The transition from one state to another corresponds to the context. To build the behavior automaton of each agent, the designer first has to identify the activation contexts of each behavior, and then build the behavior library. This library reifies the link between the agent activation and its context thanks to SO.

A SO becomes active when the agent adds it to the environment, and is desactivated when the agent retracts it. The agent dynamically chooses the activation SO that it adds to the environment; each additional activation SO triggers a particular activation context for the agent. In this way of modeling the activation process, we put context analysis into the environment. The same behavior can be activated from various SO and thus in various contexts. Conversely, the same SO can be used for several behaviors, enabling an agent to modify its reaction to the same context. This approach facilitates the implementation of various scenarios. Figure 5 depicts the relation between the agent behaviors and its activation SO.

V. CONCLUSION

When designing transportation systems, we identify three recurrent difficulties. Although the multi-agent paradigm facilitates the modeling by analogy, it is not sufficient. We argue to use the environment like a common coordination medium enable to solve in the efficient way this difficulties. Thus the environment is not only a common space but it participates to the management of the system components. Our coordination model is applied to solve two transportation problems: the dial and ride problem and the crisis simulation.

REFERENCES

[1] F. Badeig and F. Balbo et al. Contextual activation for agent-based simulation. In ECMS07, pages 128–133, 2007.
[2] F. Balbo and S. Pinson. Dynamic modeling of a disturbance in a multi-agents system for traffic regulation. International Journal of Decision Support System, 41 (1):131–146, 2005.
[3] Ana L. C. Bazzan. A distributed approach for coordination of traffic signal agents. Autonomous Agents and Multiagent Systems, 10(1):131–164, March 2005.
[4] Marco Diana. The importance of information flows temporal attributes for the efficient scheduling of dynamic demand responsive transport services. Journal of advanced Transportation, 40(1):23–46, 2006.
[5] R. Haesevoets and B. V. Eynen et al. Context-driven dynamic organizations applied to coordinated monitoring of traffic jams. In D. Weyns, S. Brueckner, and Y. Demazeau, editors, EEMMS07, pages 126–143, 2007.
[6] J. I. Hong and J. A. Landay. An architecture for privacy-sensitive ubiquitous computing. In Mobisys’04, pages 177–189, June 2004.
[7] D. Meignan and O. Simonin et al. Adaptive traffic control with reinforcement learning. In ATM06, pages 50–56, 2006.
[8] S. Osowski and J. Z. Hernandez et al. Decision support for traffic management based on organisational and communicative multiagent abstractions. In Transportation Research part C, volume 13(4), pages 272–298. Elsevier, 2005.
[9] A. Repenning and A. Ioannidou. Mobility agents: guiding and tracking public transportation users. In AVI ’06, pages 127–134, New York, NY, USA, 2006. ACM Press.
[10] Marus M. Solomon. Algorithms for the vehicle routing and scheduling with time window constraints. Operations Research, 15:254–265, 1967.
[11] D. Weyns and H. V. D. Parunak et al. Environments for multiagent systems, state-of-the-art and research challenges. LNCS Series, 3374:2–52, 2005.
[12] D. Weyns and N. Boucké et al. Gradient field-based task assignment in an agy transportation system. In AAMAS’06, pages 842–849, New York, NY, USA, 2006. ACM.
[13] M. Zargayouna and F. Balbo et al. Agent information server: a middleware for traveler information. In ESAW’05, volume 3963 of LNAI, pages 3–16. Springer Verlag, 2005.
[14] Mahdi Zargayouna. Modèle et langage de coordination pour les systèmes multi-agents ouverts. Application au problème du transport à la demande. PhD dissertation, University of Paris-Dauphine, Paris (France), 2007. In French.