Disturbance Management in Distributed Travel Information System

M. F. Feki*. S. Hammadi**

*Central School of Lille
France (e-mail: firas.feki@ec-lille.fr).
***Central School of Lille
France (e-mail: slim.hammadi@ec-lille.fr).

Abstract: To generate a route using several transport operators, we have to check different Internet websites. This complex planning task can be very difficult and does not guarantee the optimality of the selected route. Moreover, when a perturbation occurs, the need to the multimodal best path grows and consulting several websites becomes harder.

We are therefore interested in designing a support information system capable of providing multimodal information linking several transport operators. The system in question must be able to assist the user in the planning phase and also during his trip by informing him in case of disturbance and by proposing alternatives routes if necessary. To facilitate the information optimization in a distributed environment, this work is based on Multi-Agent systems and the O-MaSE methodology.

Keywords: Transport perturbation management, Multi-agent, O-MaSE, information systems, multimodal information, public transport.

1. INTRODUCTION

Today, we face a huge demand for the use of public transport. This is due to several factors. For example the growing interest in preserving the environment particularly to reduce greenhouse gases emissions.

Limiting the use of personal vehicles remains a key point to achieve this goal. In addition, the population increase causes insufficient parking spaces and, in rush hour, waiting time lengthens in traffic jams.

All these factors discourage the use of personal vehicles for a significant increase in demand for public transport.

We propose in this work an agent based information System to improve the quality of public transport service by helping users planning their trip and assisting them in case of a disturbance.

The route planned could passe through several operators and different transport modes so the disturbance could occur on any operator or on any line. The perturbation managing system (DPM) should be able to communicate with the different Transport Operator Information System (TOIS) to detect the disturbance, calculate an alternative path and inform the user.

2. PUBLIC TRANSPORTATION DISTURBANCE

The travel information becomes very important when a perturbation occurs. In fact, users try to find alternative routes using all available transport modes.

In this paragraph we present some perturbation aspects and causes.

Many factors may produce disturbance in public transport. Duc (DUC, 2007) classifies the transport perturbation depending on the external and internal factors as shown in Fig 1.

![Fig 1 - Perturbation classification](image)

All the listed factors are the causes of one or more of the following possible disturbances:

- Interruption of one line or more.
- A modification in arrival times.
- A station no longer served by a line.

To provide a better quality of server to the traveller, it is important to inform him as soon as possible of any disturbance and provide him with an alternative path.
3. DISTRIBUTED SYSTEMS

3.1 Distributed travel systems

The definition of the distributed system proposed by Wang and Kaempke (Wang & Kaempke, 2006) is as follows: A distributed information system (or distributed system) is defined as a system that is composed of a set of autonomous subsystems and a central computing server. The subsystems may overlap at some places called intersections. There is no central database that aggregates all the data of the subsystems since such a database is neither operationally feasible nor economical.

An example of such a system, a travel information system that provides customers with suggestions of travelling paths. This travel information system involves various companies (Transport Operators) each operator manages various means of transportation (bus, train, subway ...). The transport operator networks overlap at the transferring points (intersections).

The operators represent the sub-systems, each of them stores and maintains its data and the central computing server is the information system proposed in this paper. It computes the global best path by communication with the Operators through their TOIS.

3.2 Fastest paths in distributed systems

In distributed system, it is not possible to use time-dependent fastest paths algorithm like Chibani’s one (Chabini, A New Short Paths Algorithm for Discrete Dynamic Networks, 1997) (Chabini, Discrete dynamic shortest path problems in transportation applications : Complexity and algorithms with optimal run time, 1998) due to the data location.

In 2004 Wang and Kaempke (Wang & Kaempke, 2004) proposed the first polynomial algorithm that guarantees to find the shortest route in distributed systems. Their approach consists in building a virtual complete intersection graph (a skeleton of the global path) and then filling it by the contribution of the different sub-systems (local path of each sub-system).

We proposed in previous work (Fekiri & Hammadi, 2009) an algorithm that computes shortest paths in dynamic distributed system. We will use this algorithm in the module dealing with helping users planning their trip.

4. MULTI-AGENT SYSTEM FOR DISTURBANCE MANAGEMENT IN PUBLIC TRANSPORT (DPM)

4.1 Agent methodologies and O-MaSE

Several researches were made on multi-agent methodologies. Bauer and Mueller (Bauer & Mueller, 2004) proposed a classification table where the methodologies are differentiated according to their basis, phases, application area and agency support.

Three main groups are distinguished as following:

- Methodologies based on Agent-Orientation.
- Methodologies based on Object-Orientation
- Methodologies based on Knowledge Engineering.

The most used methodologies are still those based on object-orientation such as MaSE (DeLoach, 1999) Prometheus (Padgham & Winikoff, 2002). In fact, these methodologies cover the analysis phase, the conception phase and also implementation phase in some cases (using specific tools).

Many studies developed multi-agent systems (MAS) for travel-problems and public transport in multimodal networks. Among these works, we can find the Information System proposed by Balachandran (Balachandran & Enkhsaikhan, 2006) who chose MaSE methodology because of its simplicity. We are also interested in the use of this methodology because of its complete-life cycle. In fact, MaSE proposes tasks for analyzing, designing and developing MAS.

The only limitation that has an impact on our system is that MaSE do not have the ability to handle dynamic systems. In fact, as Scott said, “MaSE do not propose a possible model to entrance and leaving agent during system execution” (DeLoach S., 2006). Or in our system, a disturbance may appear and disappear dynamically. So depending on how we model the disturbance, we should have the possibility to make agents appear and disappear also when the system is running.

Further research shows that an extension of MaSE methodology called O-MaSE (DeLoach S. A., 2005) exists and takes into account dynamic systems.

So, we choose the Organization-based Multi-agent System Engineering (O-MaSE) to develop our System.

This method conceives a MAS as an agent organization where agents are the members. Each agent plays a specific role according to his ability to get its objectives. Therefore, the purpose of this method is to construct an organizational agent society based on the meta-models of the organization.

4.2. Modeling the system

To develop our MAS, we use the O-MASE methodology as mentioned in the previous paragraph.

We start by the problem description: the main objective is to guide travellers from a place A to a place B in normal traffic and also when a perturbation occurs.

One of the difficulties lays in the separation of transport operators’ database. In fact, every operator controls his own database and it is not possible to merge all databases in a huge centralized one due to economic and political reasons (Kamoun, 2007).

It is important to take into account this constraint in the requirements engineering activity of the initiation phase.
To build the goal model diagram, we use the “AND/OR” decomposition method. As indicated by its name, this method proposes two refinement ways. The “AND” refinement way used in conjunction subgoals (to satisfy a parent goal, all child goals have to be satisfied). And the “OR” refinement way which is used for disjunction subgoals (to satisfy a parent goal, it is sufficient to satisfy one child goals).

In this work, our main goal is guiding the traveller even in case of transportation perturbation (Goal0). The goal0 depends on achieving two sub-goals which are “routes search” (Goal1) for traveller assistance and “Perturbation management”. The (Goal1) is decomposed into tree sub-goals. Indeed, to build a route, we start by reading the transportation data (Goal1.1). Next, we calculate the global path using optimized algorithms (Goal1.2). Finally, formatting and sending the response to the passenger (on a PC or PDA ...) (Goal1.3). At this level, we must take into account the data localization. Indeed, as previously described, every transport operator manages his own database. So to compose a global path passing through several operators, we have to limit our research only on the potential operators offering a section of the final path. This task can be done by decomposing the (Goal1.1) into (Goal1.1.1) and (Goal1.1.2).

We first identify the transport operators involved in the global path search (Goal1.1.1). Then, we extract the information from the concerned operators (Goal1.1.2).

The (Goal2) "disturbance management" is composed of two sub-goals. Indeed, to manage a transport perturbation, we have to detect the disturbance (Goal2.1) which triggers the (Goal 2.2) "traveller orientation". To orientate a passenger in case of perturbation (Goal 2.2), we have to determining the impact of the disturbance on the proposed itinerary (Goal2.2.1), inform the traveller about the details of the perturbation (Goal2.2.2) and propose alternative route (Goal2.2.3). See Fig 2.

In the role diagram we describe the roles that can be played by DPM’s agents. For each goal / sub-goal identified above, we must create a role that can achieve it.

A role may achieve several goals simultaneously.

Agent diagram
This diagram is created so that there is at least one agent with all the needed capabilities in order to play every role.

Each class of agent represents a template for a type of agent that can be instantiated several times depending on the system requirements.

The O-MaSE philosophy provides good design flexibility. In fact, it separates roles and agents. By this way, we can use a minimum number of agents to ensure the system functionalities. Each agent plays different roles depending on the situation. Thus, we optimize the memory use.

For a good execution of our system, at least three agents must work simultaneously: A first agent (DPM_1) will play the following roles: (route composer), (Impact disturbance calculator) (Informing traveller about disturbance), (Response format). A second agent (DPM_2) will play the roles: (limit search zone). Finally, a third agent (DPM_3) will play the roles (information extractor) and (perturbation detector).

Fig 2 - DPM goal diagram
A plan model is represented as finite state automata. It is designed to achieve a specific goal. As described in Fig 3 we need eight plan models. Because of limited space we do not present any of them.

4.3. System functioning

First, we should remind that we have a "client side" vision on the problem. We are trying to optimize the service quality provided to passengers. But we cannot act on the number of buses in transit or their frequency.

In fact, this latter kind of regulation is in the responsibility of transport operators. Several studies focused on this issue and proposed algorithms (strategies) to improve traffic flow after a perturbation. Among these works we can find those of ZIDI (Zidi, Système Interactif d’Aide au Déplacement Multimodal (SIADM), 2006), Borne et al (Borne, Fayech, Hammadi, & Maouche, 2003).

In our case, we try to help travellers in normal traffic and after perturbations without acting on the network architecture or the transport frequencies.

We proceed in an informative way by providing optimized routes to passengers in normal and perturbed situations.

System Startup

At system startup, for each TOIS available, an agent will be created and associated to it. Each one of these agents will play the roles (Extract information) and (Disturbance detection) and will have all the necessary tools to consult the TOIS.

If a TOIS is disconnected, the associated agent will die and if a new TOIS appears, a new agent will be created.

Moreover, when the system startup an agent playing the role (Zone limitation) will be created. This agent represents a kind of directory maintaining the list of agents in the system, their addresses and the services they offer. This agent directory is updated when a new agent is created or when an agent dies. Fig 4.

Running the system

When a user X logs in, an agent AUx will be created. This agent will play the roles (Compose path, Compute impact perturbation, Inform travellers about the perturbation) and will be associated to this user (the user who just logged in).

After being logged in, the user X makes his request (by choosing the parameters: departure station, arrival station, departure time) and indicates the mean with which he wishes to be informed in case of perturbation (SMS or e-mail). These data will be stored in the agent memory. So (AUx) will save the route selected by the user and a mean to contact him in case of perturbation (sms or e-mail).

To generate a global path, the AUx interact with the agent playing the role (Zone limitation). This latter identify the search area using an inundation algorithm (Kamoun, 2007) and returns a list combining TOIS linking the departure station to the arrival one without detailing the means used in each local path.

After that, the agent playing the role (Compose path) will calculate the global route using a dynamic distributed path search algorithm (Feki & Hammadi, 2009).

The execution of this algorithm begins by constituting an intersection graph presenting the exchange poles between the different operators.

Then, we subsequently build a global path by asking the distant Information Systems IS (in our case the TOIS) to retrieve the best offer between intersection nodes. In each step, we select a TOIS to ask based on a modified Dijkstra algorithm.

Fig 3 - DPM role diagram
To ask TOIS about the best offers, the agent playing the role (Compose path) communicates with the agent playing role (Extract information). This latter is able to respond to simple requests like “what is the best offer to go from A to B” (A and B station of the associated TOIS) (FEKI, KAMOUN, & HAMMADI, 2008).

After the global path calculation, the agent playing the role (Compose path) will change his role to format the calculated path and transmit it to the user.

Operating after a perturbation event

If an agent playing the role (Disturbance detection) detects a perturbation in its associated TOIS, it begins by requesting the addresses of all agents playing the role (Compute impact perturbation) from the “Directory” agent (the agent playing the role “Zone limitation”). Then, it transmits the disturbance information to them.

Each one of these agents estimates the impact of the disturbance on the routes previously proposed to users. After this task, four scenarios are possible:

- The perturbation had no impact on the proposed route. Then the agent ignores the information.
- The perturbation has an impact on the route, but the section concerned by this perturbation was already travelled by the user. The agent ignores the information of disturbance.
- The perturbation impacts the current segment or impacts the next section and the user is travelling. As an example, the Fig 5 shows a traveller going from station A to station B but when he reaches the segment [X, Y], a perturbation occurs on the segment [Y, Z]. In this case, the agent playing the role (Compute impact perturbation) uses the dynamic distributed algorithm to look for a route starting from the next station of the current transport mode (station Y) to reach the final destination (station B).
- If it finds a new possible path (Y, P, M, B), it sends it to the user by e-mail or sms (according to the user's choice).

6. CONCLUSIONS

We propose in this work an agent based information system capable of providing comodal information linking several transport operators and informing users in case of perturbation. We chose O-MaSE methodology to model this system. For a good functioning, at least three agents must exist simultaneously to ensure a minimum number of roles (Extract information) from existing TOIS, (Zone search) to define a research area reducing the number of TOIS. (Compose path) to find a global path passing through different operators.

Depending on the situation, we may add other roles like (Disturbance detection) or (Compute impact perturbation) to identify a disturbance and estimate its impact on a proposed itinerary.

In next work, we will present the implementation aspects of this system and its integration in a J2EE platform.
REFERENCES


