Multiagent Systems in Travel Planning

Diego Fialho Rodrigues*, Heber Amaral*, Alcione de Paiva Oliveira* and Simone Dornelas Costa†

*Departamento de Informática
Universidade Federal de Viçosa - UFV
Viçosa, MG - Brazil
diego.fialho@ufv.br, heberfa@gmail.com, alcione@dpi.ufv.br

†Departamento de Computação
Universidade Federal do Espírito Santo - UFES
Alegre, ES - Brazil
sidornellas@gmail.com

***Abstract*—Currently, as a result of globalization and other factors, the tourism market has remained high. Assisting the customer in order to satisfy their desires and constraints is a major challenge in this sector. This paper presents an agent-based system for developing a travel plan. In the system, according to the preferences informed by tourists, several agents cooperate in order to construct the travel plan that best suits the client needs. A tourism domain ontology was developed to enable communication between the agents. The system relies on the assistance of the greedy algorithm to trace the routes. A heuristic function was designed to ensure the fulfillment of the customer goals.**

Keywords—multiagent; ontology; Travel Planning

I. Introduction

It's known that building a travel packet that satisfies the majority of desires and constraints of the clients is a tough task. There are several decisions to be taken and restrictions to be satisfied such as: displacements, schedules, time and cost constraints, accommodation restrictions, tourist attractions to be visited, etc. Therefore, a growing interest for automated systems that supports travel packets construction can be noticed nowadays. The agent-based systems approach can be helpful in this domain due to its characteristics: many autonomous agents (representatives from airlines, hotels, bus companies, tourist attractions) with their own goals interact to build a travel plan. To implement such a system is necessary to first establish a common vocabulary, using a lightweight ontology, so that agents can communicate. It is then necessary to design agents, the roles to be played and the interaction between agents in order to produce a travel plan that meets the users needs. This paper presents a Multiagent system (MAS) [1] for the travel planning problem. A tourism domain ontology was developed to enable communication between the agents. The system relies on the assistance of the greedy algorithm to trace the routes and the information required for the plan creation is located in a relational database. A heuristic function was designed to ensure the fulfillment of the customer goals. As related work, Lopez and Bustos [2] used a multiagent systems based on mobile devices to build a trip plan for a specific date. Schiafino and Amandi [3], presented an expert system, named *Traveller*, used for user assistance in solving travel problems.

II. Problem description

The problem to be addressed lies in the travel planning domain and the solution presented can be used to assist travel agencies. The problem consists in the construction of a travel plan that best satisfies the client needs, considering not only particular restrictions of the travel packet problem, but also clients constraints, like: time, money, preferences, and so on. The MAS approach is proposed in order to treat such a problem. The agents play roles in the system and interact in a collaborative way inside the environment. Each agent has its particular goals and the union of all agents' goals together reaches efficiently the general goal of building a travel packet. The Table I presents the roles in the system.

It is important to notice that each agent has an individual goal, which is related with the role played. The behavior of all agents and all their interactions, both with the environment and between the agents, shape the general behavior of the system.

The problem has some constraints that must be taken into account during the travel plan construction:

- The planning must fit in a time range (minimum and maximum date).
- There must be a minimum and a maximum number of cities to be visited.
- The travel time must be minimized.
- There must be a threshold cost.
- The travel must start in and return to the same place inside a period of time.

TABLE I. Roles occurring in the system

Role	Description
Planner Agent	It is the agent responsible for the development of the sequence of steps that make up the travel plan.
Tourist Agent	The agent that asks for the travel plan creation.
Travel Agent	The agent responsible for informing means of transport, accommodations, and interesting places to be visited.
Other Agents	Agents that represent airlines, bus companies, hostels, etc. These agents are responsible for informing prices, places available, and travel time.

- One city must not be visited more than once, except in the case it is used to return to another city.
- A person can visit the same kind of tourist attraction, in one or more cities, but not the same tourist attraction.

III. The system

To model the requirements of this project Tropos methodology [4] was used to help improve understanding of roles and goals of each agent. The main agents of the system are: *Planner*, *Tourist*, *Travel*, *Transport*, *Events Promoter* and *Lodging*.

The *Tourist* agent depends on the *Travel* agent to achieve the following objectives: (a) Perform Travel, (b) Cost Limited, (c) Limited Time, and (d) Minimum Transit Time. The objective (b) is related to financial constraints of the *Tourist* agent. The goal (c) is related to the time available to the *Tourist* agent to make the trip. The objective (d) is related to the *Tourist* agent desire to spend as little time as possible on transportation vehicles.

The *Tourist* agent also has a soft-goal, i.e., a goal that is important to be reached but it is not top priority, namely: *Enjoy the Journey*. This agent requires the asset: *travel plan*, which will be assembled by the *Planner* agent.

The *Travel* Agent is the only agent that communicates with all other agents to get the following information from them: (a) *Tourist* agent preferences, (b) *Transportation*, (c) *Attractions*, and (d) *Lodging*. The *Travel* Agent needs this information to achieve its goal *Sell Travel Package*. This agent has as soft-goal to *Attend well Tourist* agent. It also has the task: *Assemble Travel Plan*.

The *Planner* agent needs several resources, which are available for this agent by communicating with the *Travel* agent, namely: (a) *Tourist* agent preferences, (b) *Lodging*, (c) *Transportation*, (d) Pre-defined Plan from *Tourism* agent, (e) *initial City*, and (f) *Attractions*.

The agents *Transport*, *Lodging* and *Event Promoter* are only responsible for providing the resources: *Transportation*, *Lodging* and *Attraction*, respectively, for the *Travel* agent. Fig. 1 shows through an use case diagram the actors and their functional dependencies.

In order to enable communication between the agents was necessary to develop an ontology for the travel domain. Although the ontology was created using OWL tools, it was stored as relational database due to its capabilities for fast retrieval of the data. This was an important feature as there were a great deal of instances (cities, attractions an hotels) to stores and manipulate. Fig. 2 shows the database schema used to store the ontology instances.

Most of the classes are self-explained. The class *Attribute* has a single property name, which stores the description of the attribute including it in a theme or a facet of the related classes. The attributes have many-to-many: with classes: *Accomodation*, *Attraction*, *Event*. For instance: (a) a rock concert, which is an event, can have the attribute “music” among others; (b) a resort have rooms that can have the attribute “whirlpool”. The *Accomodation* class store information for

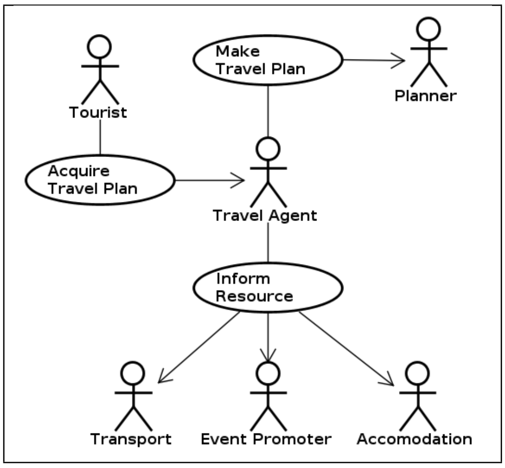

Fig. 1. System use case diagram.

inns, hotels and hostels. The property *kind* stores the type of accommodation. The *Transportation* Class represents the various means of transportation such as: bus, plane, ship, etc. The routes are specified from the *Link* class. Each route has the city of origin, destination, and a means of transportation. The associated class *LinkHasTransportation* specify the price and travel time between the cities for a particular mode of transport. These properties are important for assembling the travel plan.

A. The Planning Algorithm

In this work we adopted a greedy search for the assembly of the trip itinerary. In the greedy method decisions are made in an isolated way and in every step its decision is based on a heuristic function. The purpose of this function is to guide the greedy algorithm in the search for the most interesting attractions, i.e. those that meet the highest amount of desires and constraints of the tourists. Thus, we specified the

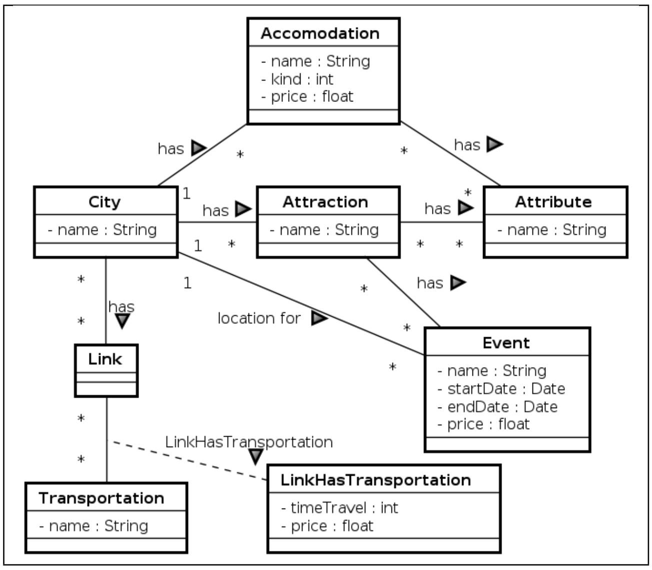

Fig. 2. The database schema.

criteria that influence, positively and negatively in the decision to participate in one attraction, namely: (a) the number of attributes that match the tourists preferences (b) the attraction cost, and this comprises the attraction cost plus lodging (it depends on the number of stay days), (c) the transportation cost, (d) and the transportation time to the city of the attraction. This is expressed the following function:

$$\begin{aligned} f(a) &= n(a) * k_1 \\ &\quad -(c(a) + cl(a) * s(ca)) * k_2 \\ &\quad -(tt(ca) * ct(ca) * k_3) \end{aligned} \tag{1}$$

Where: a denotes the attraction; ca stands for the city of the attraction; $f(a)$ is the heuristic value of the attraction; $n(a)$ is the number of attributes satisfied by the attraction; $c(a)$ is the cost of the attraction (ticket and other costs of the attraction); $cl(a)$ is the cost of accommodation in the city of the attraction; $s(ca)$ is the number of days staying in the city of the attraction; $tt(ca)$ is the time spent traveling to the city of the attraction; $ct(ca)$ is the cost of traveling to the city of the attraction; k_1, k_2 and k_3, are constant for adjustment of the equation. The great weight of the portion $(tt(ca) * ct(ca) * k_3)$ aims to make the algorithm select attractions in cities located close to each other. The algorithm also takes into account the financial cost of the return to hometown (co). The algorithm has addressed these constraints through the function $r(co)$ (cost of returning to co):

$$r(co) = (c(a) + cl(a) * s(ca)) - (cv(co, c)) \tag{2}$$

Where: co denotes the hometown; $cv(co, c)$ is the cost of returning hometown from the attractions' city. The value of $r(co)$ must always be more than 0.

IV. Results

Fig. 3 shows the output generated by the system for a simulation where the client requests a travel itinerary for a tourist region well known in Brazil, called "Serras Gauchas".

```
Event  1
Name = Festa do Boi
city = caxias do sul
Accomodation = hotel de caxias
Number of Days = 2
Event Price = 8.0
Accomodation Price=250.0
Start Date = Sun Mar 04 00:00:00 BRT 2012
Event  2
Name = Festa da cultura gaucha farropilha
city = porto alegre
Accomodation = porto do sol
Number of Days = 6
Event Price = 12.0
Accomodation Price=120.0
Start Date = Tue Feb 21 00:00:00 BRST 2012
Event  3
Name = V Encontro do chimarrão
city = passo fundo
Accomodation = passo fundo grande hotel
Number of Days = 5
Event Price = 9.0
Accomodation Price=150.0
Start Date = Wed Mar 07 00:00:00 BRT 2012
Event  4
Name = Festival do churrasco
city = gramado
Accomodation = grama palace hotel
Number of Days = 4
Event Price = 11.0
Accomodation Price=200.0
```

Fig. 3. Itinerary generated for "Serras Gauchas".

V. Conclusions

This paper presented a MAS in order to assemble tour packages according to the preferences and limitations of the tourists. In the MAS agents were used to search for information such as: accommodation, attractions, events, cities, transport, etc. A specific planner agent was used to assemble the travel plan from the information provided by other agents.

The greedy algorithm was used to assist in the assembly of the travel plan. The choice of cities was done in an iterative manner, being influenced by criteria: (a) positive: attributes of tourist interest, and (b) negative: cost and travel time to the next town. Because of this, it can be observed that the choice of cities have a behavior prone to formation of clusters, i.e. where it tends to group cities relatively close if they meet the interests of the visitor. The solution proved to be feasible in the tests with a small number of cities. Tests with larger number of cities need to be done.

As future work it would be interesting to use Internet to assemble tour packages. So it could seek the route information as distance, travel time between cities, etc., using Google Maps for instance. It would be interesting also to evaluate other search algorithms such as A*, which has the property to obtain optimal solutions for admissible heuristics. The system could also use a database of small pre-built routes to be dynamically joined.

Acknowledgment

This work is financed by funding agencies FAPEMIG, CNPq, FUNARBE and by the Gapso Company.

References

[1] M. Wooldridge, *An Introduction to MultiAgent Systems*. John Wiley & Sons Inc, 2002.

[2] J. S. Lopez and F. A. Bustos, "Multiagent tourism system: An agent application on the tourism industry," in *Proceedings of the International Joint Conference IBERAMIA/SBIA/SBR*, Brazil, 2006.

[3] S. Schiafino and A. Amandi, "Building an expert travel agent as a software agent," *Expert Systems with Applications*, vol. 36, pp. 1291–1299, 2009.

[4] P. Bresciani, P. Giordini, F. Giunchiglia, J. Mylopoulos, and A. Perini, "Tropos: An agent oriented software development methodology," *Journal of of Autonomous Agents and MultiAgent Systems*, vol. 8, no. 3, pp. 203–236, May 2004.