Smart Ride Seeker (SRS) An Introductory Plan

Sameh Abdel-Naby [a,1] and Paolo Giorgini [a]
[a] *University of Trento, Department of Informatics and Telecommunications (DIT)*

Abstract. Based on the use of location and available car seats, Car sharing systems allowed a substantial number of people to share car rides.This paper proposes the initial phase of Smart Ride Seeker (SRS), which is a Car Pooling-like technique for distributing resources among a community. This paper develops the SRS technique through a mobile-based application that allows the mapping of ride seekers' locations along with the locations of available cars on a graphical interface/map, giving the possibility to calculate the optimal path for both the ride giver and seeker to fulfill their demands.

Keywords. Car Pooling, Agents, Mobile Communications, Pollution

1. Research and Motivation

Car Pooling Problem is NP-Hard [1]. A branch and cut algorithm for the vehicle routing problem [2], and in our framework a vehicle routing algorithm is also applied for a certain unit customer demand. Therefore, SRS will be involving NP-Hard-*like* algorithm. For purposes of simplicity and practicality, places for ride seekers to meet ride givers are assumed to be fixed. Initially, the system will recognize only 3 stops, each of which acts as a pickup/dropping point. We assume that the system will automatically assign one of the pre-defined fixed pickup/drop stops (PDS) to ride givers (RG) or ride seeker (RS) according to their home or work locations. A common problem for routing the requests within a Car Pooling system is handling a multipoint trip. Calculating a distance between two points is mostly constrained by the time limit. So finally the system avoids having any ride givers that are going to take more than the ride seeker's preferred time. But sometimes the opposite takes place, which will let us understand that the ride giver is taking a different and longer path or maybe stopping between the two points.In SRS, we introduce the notion of mapping all PDSs on a graphical interface that acts as a territory map, reflecting the real world actual distances. Accordingly, the system recognizes whether the ride giver is taking the appropriate path to reach the destination. In the second phase, the system recommends a certain request to a specific situation. On a later stage of application, proper routing messages can be reached by linking the system to maps or addresses database that would help giving up the use of fixed PDSs gradually and rely completely on mapping users on graphical interface and calculating the distances needed

[1]Correspondence to: Sameh Abdel-Naby, via Sommarive 14, I-38050 Povo, Italy. Tel.: +39 0461 881508-2020; Fax: +39 0461 883964; E-mail: sameh@dit.unitn.it.

to accomplish more reliable requests routing. At the SRS initial phase, system participants are well determined, two system actors will be located in the centralized focus of the scheme: the *ride seeker* and the *ride giver*. A car sharing system will be basically concerned with fulfilling these two entities demands as well as processing their requests. System inputs and outputs will be exchanged between these two actors and, in turn, the connection between system, seeker and rider should be well-established. Standard mobile phones relations are taken as a method of communication. But autonomous agents will be negotiating the ride details and finally communicating the final agreement with the mobile user. Similar architecture - *selling used books for student* - was described by scholars [3]. Toothagent: a Multi-Agent System for Virtual Communities Support.). Our suggested scheme will be using an any-time algorithm suggested by scholars [4] that is expected to help us in solving the problem of tasks allocation (care ride requests routing) within a community of autonomous agents, and as a result, agents are expected to shape an alliance to better perform a certain car *ride seeker* request between multiple points.

2. SRS Scenario

The RS will be going through an initial phase to select the application interface language and the action to be taken (for example, seeking a ride or offering one). By selecting the "seeking ride" option, the system will turn the RS to the next step: selection of the nearest PDS as well as the PDS at the final destination. The last step will be selecting the ride date and time within a flexible time range. All these selections will be saved in the mobile-based interface, and then sent to a managing server, either by short message service (SMS) or by Bluetooth, whatever appropriate. Then the user will wait for a reply containing the ride details. The RG will be going through two possible situations: the first is when s/he would like to offer a new ride service within a specific time range, and the second is when s/he is asked about the possibility to give a ride. In the first situation, upon entering his/her offer, the status of the RG will be always Pending during this time and s/he will be receiving messages directly from the system. In the second situation, when a ride seeker requests a ride at a certain time and between two PDSs that were common to be offered by a certain RG, the system would automatically send this request to the RG regardless of his/her status (for example, Pending) in the system. To perform this function, the SRS has to maintain a logging and communication history, and save the users' destination and time records. At this point, a technique similar to Agents Cloning [5] can be used to facilitate recommendations of requests and referral routing.

The SRS will be supervising four major tasks: 1) managing the request routing process and demand matching; 2) running a reputation system by asking feedback from ride takers; 3) managing a crediting system, in which a ride seeker or giver will get upon registering in the crediting system, presumably, five credits for free; to move from a place to another, the seeker will donate two credits to the ride giver; finally, at a certain time the ride seeker will have to choose either to start to offer rides to others to collect credits or to buy them; and 4) supervising methods of communication between system agents and actors.

References

[1] Araque, J. R., T. L. Morin, J. F. Pekny. 1994. A branch and cut algorithm for the vehicle routing problem. *Ann. Oper. Res*. 50 37-59

[2] Letchford, A. N., R. W. Eglese, J. Lysgaard. 2002. Multistars, partial multistars and the capacitated vehicle routing problem. *Math. Programming* 94 21-40.

[3] Bryl, V., Giorgini, P., Fante, S. 2005. Toothagent: a Multi-Agent System for Virtual Communities Support. , *Informatics. Telecomunication*. Technical Report DIT-05-064

[4] O. Shehory and S. Kraus Task Allocation via Coalition Formation Among Autonomous Agents, Proc. of IJCAI-95, pages 655-661, Montreal, August 1995.

[5] Shehory, O., Sycara, K., Chalasani, P,. and Somesh Jha. “Agent cloning”. The Robotics Institute, Carnegie Mellon University, Pittsburgh.