



The 4th International Conference on Ambient Systems, Networks and Technologies  
(ANT 2013)

## Estimating scalability issues while finding an optimal assignment for carpooling<sup>☆</sup>

Luk Knapen<sup>a,\*</sup>, Daniel Keren<sup>b</sup>, Ansar-Ul-Haque Yasar<sup>a</sup>, Sungjin Cho<sup>a</sup>, Tom Bellemans<sup>a</sup>, Davy Janssens<sup>a</sup>, Geert Wets<sup>a</sup>

<sup>a</sup>Hasselt University, Wetenschapspark 5, B3950 Diepenbeek (Belgium)

<sup>b</sup>Department of Computer Science, Haifa University, Haifa 31905 (Israel)

---

### Abstract

An automatic service to match commuting trips has been designed. Candidate carpoolers register their personal profile and a set of periodically recurring trips. The *Global CarPooling Matching Service (GCPMS)* shall advise registered candidates on how to combine their commuting trips by carpooling. Planned periodic trips correspond to nodes in a graph; the edges are labeled with the probability for negotiation success while trying to merge planned trips by carpooling. The probability values are calculated by a learning mechanism using on one hand the registered person and trip characteristics and on the other hand the negotiation feedback. The *GCPMS* provides advice by maximizing the expected value for negotiation success. This paper describes possible ways to determine the optimal advice and estimates computational scalability using real data for Flanders.

© 2013 Published by Elsevier Ltd.

*Keywords:* Graph theory, Agent-based modeling, Scalability, Dynamic networks, Learning

---

### 1. Problem context

An advisory service for carpooling while commuting is to be built. People will register their *periodic commuting trips*: the base period typically is one week i.e. a specific pattern valid for working days is repeated after every seventh day. Considering one week periods accommodates for most situations (including part-time workers). People who are able to fulfill all their carpooling needs within their own social network (*local exploration*) of acquaintances, are assumed not to need the advisor service. Others will need to explore the set of yet unknown carpooling candidates (*global exploration*). The matching service shall determine which trips are best suited to be combined for carpooling and provide advice by suggesting people to start a negotiation with respect to a specific periodically executed trip.

---

\*Corresponding author: Tel.: +32-11-269-126; Fax: +32-11-269-199  
E-mail: [luk.knapen@uhasselt.be](mailto:luk.knapen@uhasselt.be)

## 2. Advisor Model

### 2.1. Principle of Operation

In order to find carpooling companions, people who did not find a suitable partner by exploring their private network, register themselves with the *GCPMS*. Registration implies first posting some descriptive characteristics like age, gender, education level, special interests (like music style preferences), job category, driver license availability, etc. Those qualifiers are used because it is known that continued successful cooperation between people requires a minimal level of similarity.

Secondly, people post information about each trip they periodically plan to execute: those data consist of origin and destination locations, earliest and latest departure and arrival times, the maximal detour distance that is acceptable and the availability of a car (possibility to drive). Note that a particular driver license owner can be unavailable for driving on a specific day of the week because the family car on that day is in use by her/his partner.

*Periodic trip executions (PTE)* need to be matched, not people. A periodic trip on Wednesday from A to B leaving at about 08:30h needs to be matched with another one having similar characteristics. Of course, the people involved shall be mutually compatible but they are not the primary subject of matching. A particular individual can periodically carpool with several people for different trips in the week (on Monday with colleague A, on Tuesday with neighbour B who differs from A). Periodic trip execution is abbreviated by *PTE* in the remainder of the text.

A *pooled trip execution* is the cooperative execution of a set of trips using a single car and a single driver. As a consequence, the route for each passenger shall be embedded in the route of the driver (*single driver constraint*).

After having found a good match (details on how to do so will be explained in section 2.3) the matcher conveys its advice to the candidates involved (the owners of the matched *PTE*); they evaluate the proposal, negotiate about carpooling and possibly agree to cooperate. Note that this negotiation is not guaranteed to succeed. One of the reasons is that the individuals dispose of more information during the *negotiation process* than the service does during the *matching process*. Therefore, the candidates convey the negotiation result back to the matcher service. This paper assumes that sufficient (financial) incentives are in place in order to make this happen.

After trip execution, users can qualify each other. The *GCPMS* allows for controlled mutual evaluation of individuals with respect to timeliness and safety. Only individuals cooperating in an agreement can qualify each other. The negotiation and qualification feedback is used by a learning mechanism incorporated in the matching service. After receiving the feedback, the matching service disposes of (i) data describing the *PTE* and their owners (individuals) as well as of (ii) the negotiation result; those are used to train a logistic regression based predictor. Please refer to fig. 1 for an high level overview of dataflows, relations and method activation.

The model used for matching consists of a directed graph (see figure 2 left part); by convention, each edge points to the *PTE* whose owner will be the driver. Each vertex corresponds to a *PTE*. A vertex for which the owner is unable to become the driver, never can be a target edge (its *indegree* equals zero). Two vertices are connected by an edge if and only if it is worth to advise the *PTE* owners to start negotiating. Every edge is labeled with the estimated probability for the negotiation to succeed. Note that

1. the set of vertices evolves over time because people register and withdraw *PTE* as time evolves and because people join and leave the carpooling candidates society (removing all their *PTE* in the latter case).
2. edges emerge as soon as the estimated negotiation success probability exceeds a given threshold; this can be caused by changes in the *PTE* (e.g. by relaxing the time constraints) and people characteristics respectively (e.g. by reputation changes (see below)).
3. probability estimates can change over time by re-training the predictor. Note that this can cause threshold crossing and hence edge creation or deletion.

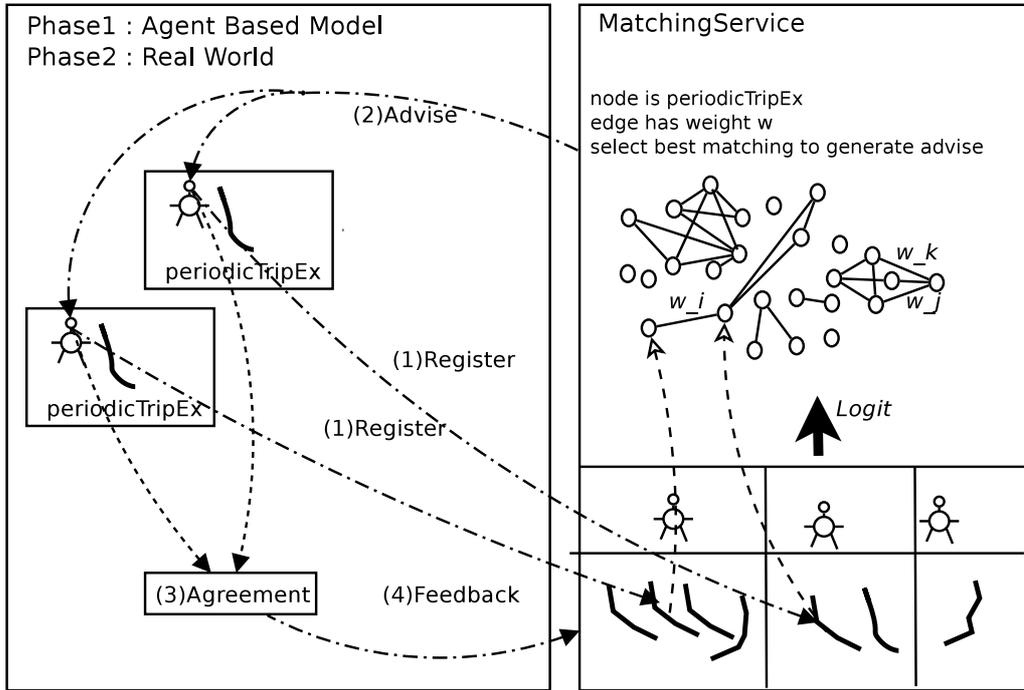


Fig. 1. Application context: the right hand side shows the *matcher service*. People register some descriptive data (lower part) about themselves and their trips to be executed periodically (*PTE*). Those constitute a graph (upper part): the edges are labeled with the probability that negotiation will succeed when the trip owners are advised to carpool. Negotiation result is fed back to train the *logit* predictor. The left hand side shows the entities exercising the *matcher service* in consecutive phases.

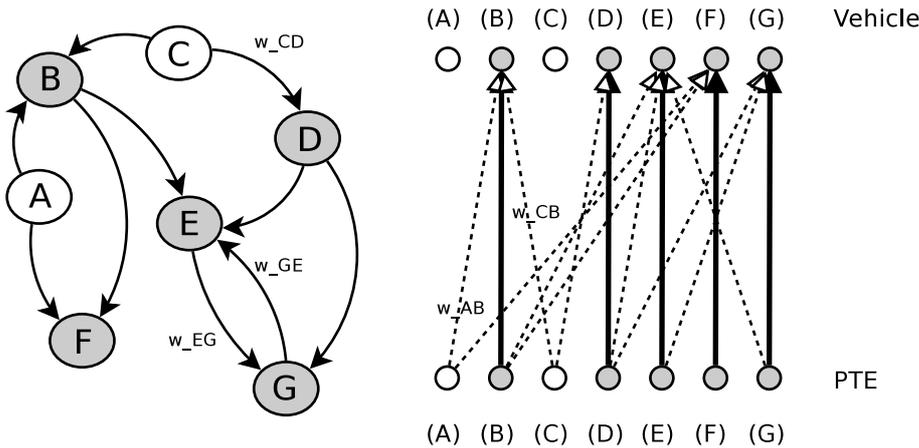


Fig. 2. The leftmost diagram shows the graph where vertices correspond to *PTE* (*periodic trip execution*) and the edges are labeled with the success probability for the negotiation (if that is sufficiently large). The rightmost part shows the same information in bipartite graph showing both *PTE* and *vehicles*. Continuous line arcs connect an *PTE* to the vehicle of its owner; dashed lines show potential participation as a passenger. Grey vertices correspond to *PTE* where the owner is prepared to drive. Some, but not all of the edges have been labeled with their weights.

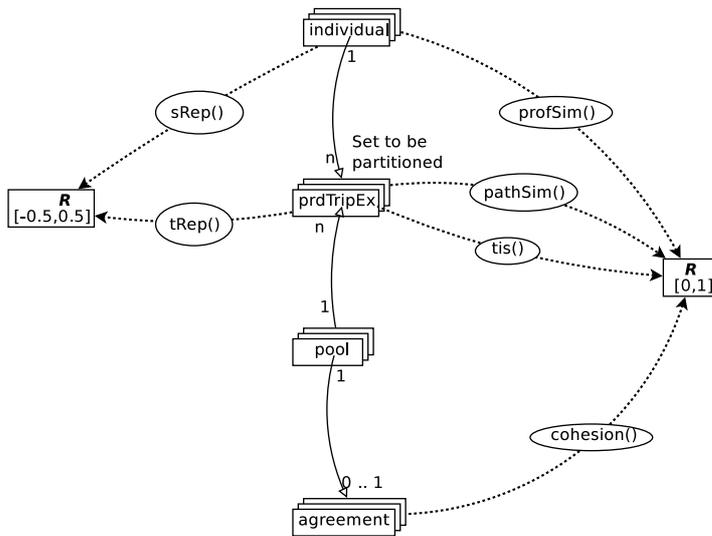


Fig. 3. Overview of functions defined on the sets (*individual*, *PTE* and *agreement*) used in the model and functions those sets are involved in. Continuous lines represent references, dashed lines represent functions.

Finally the problem size can grow large when a nation-wide service is considered. Large scale deployment probably is a necessary condition for both effective operation (delivery of advice that has a high success probability) and economic viability. The matcher needs to cope with large networks whose topology and edge weights evolve in time.

This represents a complex problem and hence thorough evaluation before deployment. An agent based model simulating the actual population behavior, will be used to exercise the matching service for several reasons. First, performance and effectiveness need to be evaluated on a running system since they are very difficult to predict from design data only. Second, deploying such system should go flawlessly because lost customers will be reluctant to return. Finally, the system behavior during the startup transient when only few customers already registered, is difficult to predict and hence observations made are difficult to interpret; simulation can support learning about the overall system behaviour.

## 2.2. Similarity, Reputation and Cohesion

The functions used to determine the input variables for the logit based negotiation success probability estimator are briefly explained in this section. Details have been left out due lack of space. Fig. 3 shows an overview.

1. Path similarity  $\text{pathSim}()$  is a value in  $[0, 1]$  assigned to an *ordered* pair  $(pte_0, pte_1)$  of *PTE* that indicates to what extent the OD (Origin, Destination) pairs involved in the respective trips, are compatible for carpooling in case the owner of  $pte_0$  is assigned to be the driver. Path similarity defines a function of *PTE* that is not symmetric in its arguments. This is easily seen because the distance driven depends on the driver selection; the driver needs a detour to pick up passengers.
2. Profile similarity  $\text{profSim}()$  is a value in  $[0, 1]$  assigned to a pair of individuals that indicates to what extent the individuals are compatible for carpooling (homophily concept).
3. Time interval similarity  $\text{tis}()$  is a value in  $[0, 1]$  assigned to an ordered pair  $(pte_0, pte_1)$  of *PTE* having identical origins and identical destinations; it indicates to what extent the time intervals involved are compatible for carpooling. Compatibility for carpooling requires a minimal amount of intervals overlap (see fig. 4). Time interval similarity can be calculated only for a pair consisting of the passenger trip and the part of the driver's trip for which the route coincides with the passenger trip route (because  $\text{tis}()$  applies to trips having identical origins and destinations).

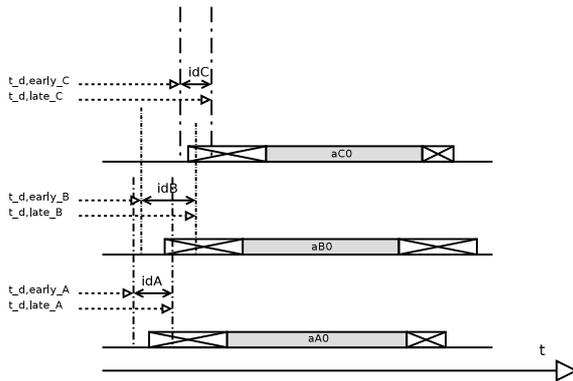


Fig. 4. Activities ( $a_{A,0}$ ,  $a_{B,0}$ ,  $a_{C,0}$ ) for individuals A, B and C and the associated trips (crossed rectangles). The valid departure intervals  $i_{d,A}$ ,  $i_{d,B}$ ,  $i_{d,C}$  are shown. Note that B can choose to co-travel with A or C but A and C cannot co-travel.

4. Safety reputation  $sRep()$  of a driver is a value in  $[0, 1]$ . Each individual has an *sReputation* value that evolves over time due to qualification by passengers (i.e. individuals who participated in an agreement where the person being evaluated was the driver). Notifications received are registered in a personal *qualifications list* with the individual they apply to; for each issuer, only the most recent qualification is kept. The *sReputation* is calculated as a weighted average of the values posted in the qualification list: the weight decreases with age of the notification and increases with the duration of the cooperation.
5. Timeliness reputation  $tRep()$  (or accuracy reputation) is a value in the range  $[-0.5, 0.5]$  assigned (by the co-travellers) to a *PTE* in an *agreement*: it indicates to what measure the owning individual respects the timing when executing the periodic trip in the *agreement*. *tReputation* is defined for both drivers and passengers. *tReputation* has been defined as a characteristic of a tuple (*PTE, agreement*) and not as a characteristic of an *individual* or of a *PTE* because an individual can behave differently on a specific *PTE*  $pte_0$  in different agreement contexts (pools).
6. Cohesion qualifies the strength of an *agreement* using a value in  $[0, 1]$  that is a function of attributes of the *agreement* only. It is a measure for the resistance to break an existing pool in case some participants receive a better offer.

### 2.3. Weights Determination - Learning Mechanism

The weights used to label the edges in the graph, are probability values associated with the success of the negotiation process between individuals. Those probabilities are calculated by means of *logistic regression* (logit) fed by results of negotiations who have been advised by the *carpoolMatcher*.

Fig. 5 summarizes the data dependencies relevant to edge weight determination. From the point of view of the matcher service, the outcome of a negotiation process is a discrete variable with values: *success* (yes) and *failure* (no). Independent variables influencing the negotiation are continuous: *profSim*, *pathSim*, *tis*, *cohesion* and *sReputation*. A *logit* model will be used to predict the negotiation outcome. Negotiation results fed back to the *Global CarPooling Matching Service (GCPMS)* are used to determine the coefficients for the *logit* model by linear regression.

## 3. Optimisation problem

The *GCPMS* aim is to maximize carpooling. Hence, the advice is based on maximizing the expected value for the negotiation outcomes.

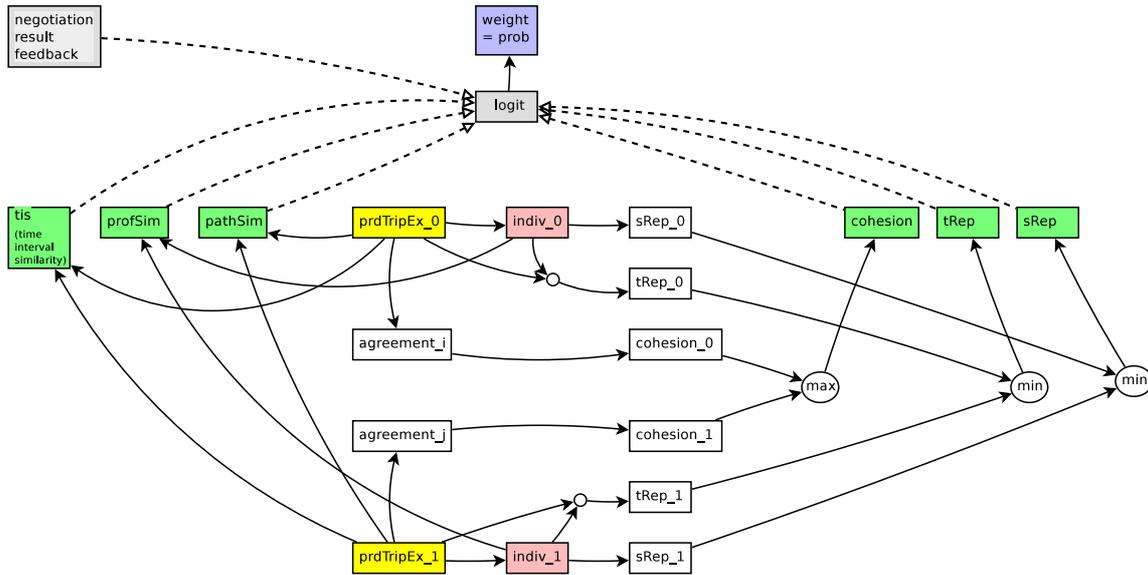


Fig. 5. Dependencies between concepts used to calculate the weight for a edge connecting two *periodicTripEx*'s

### 3.1. Graph theory based solution

1. Consider the rightmost part figure 2. Assume that the graph is connected (partitioning has already been done).
2. First, one can try to reduce the problem using following steps:
  - (a) Edges emerging from a vertex at the lower side of the diagram, are mutually exclusive. If and only if a *PTE*  $pte_i$  does not point to any other *Vehicle* than the one owned by its owner, then the edge  $pte_i, veh_i$  shall be selected.
  - (b) If a subset  $V_{nc}$  of vehicles exists so that none of its members is critical with regard to capacity, then the constraint specified in equation 4 always is met. In such cases, as soon as the subset  $V_u \subseteq V_{nc}$  of vehicles to be used is known, the *PTE* can be assigned to the vehicles in arbitrary order and hence for all the edges involved the one having maximal weight can be chosen. Note that the number of combinations to investigate equals  $2^{|V_{nc}|}$  and can be large.
3. Note that the problem sketched no longer is a *matching problem in a regular graph* (where each edge has exactly 2 vertices). The so-called *Hungarian method* (which comes down to path finding) to find a maximum matching cannot be used.
4. We conjecture that describing the problem as a directed hypergraph (in which an edge can have an arbitrary number  $\geq 2$  of vertices), can lead to an *Hungarian Method* based problem solution. [1]

### 3.2. Specification as a linear problem

1. The linear optimisation problem can easily be derived from the rightmost graph in figure 2 as follows. Let  $G(V, E)$  with vertex set  $V = V_p \cup V_v$  and edge set  $E$  denote the bipartite graph of figure 2.  $V_v$  denotes the set of vehicle vertices,  $V_p$  denotes the set of *PTE* vertices. A variable  $x_e$  is associated with each edge  $e$ . Value *one* (*zero*) means that the edge has (not) been selected.  $P(e)$  denotes the *PTE* (source vertex) for the edge;  $Veh(e)$  denotes the vehicle (target vertex).  $cap(v)$  denotes the vehicle capacity (defined as the number seats, including the driver seat). Let  $Veh(p)$  denote the vehicle owned by the owner of the *PTE*  $p$ . The weight associated to each edge that links a *PTE* to the owners vehicle, is set

to zero because those links do not contribute to the objective of maximizing the number of succeeded negotiations:  $\forall e : Veh(e) = Veh(P(e)) \Rightarrow w_e = 0$ . Then the problem statement is:

$$\text{maximize } \sum_{e \in E} w_e \cdot x_e \quad (1)$$

subject to

$$\forall e \in E : x_e \leq 1 \quad (2)$$

$$\forall p \in V_p : \sum_{\{e \in E | P(e)=p\}} x_e = 1 \quad (3)$$

$$\forall v \in V_v : \sum_{\{e \in E | V(e)=v\}} x_e \leq \text{cap}(v) \quad (4)$$

$$\forall v \in V_v : \forall \{e \in E | V(e) = v\} : x_e \leq x_f | V(x_f) = v \wedge Veh(P(x_f)) = v \quad (5)$$

Equation 2 limits the range of the (boolean) variables. Equation 3 requires that each *PTE* shall be assigned to exactly one vehicle (i.e. the trip shall be executed). Equation 4 states the limited capacity for each vehicle. Equation 5 follows from the requirement that each car be driven by its owner only.

$$\forall v \in V : (\exists e \in E | (Veh(e) = v) \wedge (x_e = 1)) \Rightarrow (\exists f \in E | (Veh(P(f)) = v) \wedge (x_f = 1)) \quad (6)$$

Equation 6 says that if there is a passenger in the car who is not the car owner, then the car owner is in the car too. By numbering both the *PTE* and vehicles from 0 to  $N - 1$ , the edges can be identified by  $x_{i,j}$  where  $i$  identifies a *PTE* and  $j$  identifies a vehicle.

$$\forall i, j \in [0, N - 1] : x_{i,j} \leq 1 \quad (7)$$

$$\forall i \in [0, N - 1] : \sum_{j \in [0, N - 1]} x_{i,j} = 1 \quad (8)$$

$$\forall j \in [0, N - 1] : \sum_{i \in [0, N - 1]} x_{i,j} \leq \text{cap}(v_j) \quad (9)$$

$$\forall i, j \in [0, N - 1], i \neq j : x_{i,j} - x_{i,i} \leq 0 \quad (10)$$

2. This in general, cannot be solved using linear programming. Note that all coefficients in the matrix equal either  $-1$ ,  $0$  or  $1$ ; this holds for the coefficients of the  $x$  unknowns in 7, 8, 9, 10 and for the *slack variables* originating from the inequalities. The special case where capacity constraints are redundant (inequalities 9 do not generate polytope faces) can be proved to correspond to an integer polyhedron and hence the simplex method leads to an integer optimum. Additional research is required to determine the amount of cases where inequalities 9 are not redundant and generate polytope faces (and hence non-integer vertices) for graphs derived from realistic data.

#### 4. Data characteristics - Problem size

Before taking a final decision about the method to choose, we analyzed the characteristics of the graph in which to embed an optimal assignment. From the output of a FEATHERS activity based simulation run for Flanders, we filtered *home-work* trips starting in a given period of time. *Profile* similarity, *path* similarity and *time interval* similarity can be calculated since the origin, destination and start time follow from the FEATHERS predictions (9139001 episodes). In order to calculate time interval similarity we assume that people are prepared to shift the trip start within a time window  $[t_0 - 20[\text{min}], t_0 + 10[\text{min}]]$  where  $t_0$  denotes the originally planned time of departure. All home-work trips starting between 07:30h and 8:00h (143395 trips) were paired to calculate profile, path and timeInterval similarity. Since we do not know the relation between those quantities and the negotiation success probability, we assumed the probability to equal the

product of the similarities. For profile similarity, we considered the same attributes as [2]. Each trip pair for which the probability exceeded *minProb* was added to the graph. This was repeated for several *minProb* values. Results have been reported in table below showing characteristics of the resulting graphs.

minProb	nVertices	nEdges	nComponents	Compon.Size Largest	Compon.Size Average	nEdges/nVertices
0.80	102	58	44	5	2.32	0.57
0.75	719	492	247	13	2.91	0.68
0.70	3706	3062	763	427	4.86	0.83
0.65	13993	15478	1380	5223	10.14	1.11
0.60	37748	64737	1659	19601	22.75	1.71
0.55	74241	230731	1249	70452	59.44	3.11
0.50	109667	733098	451	108517	240.61	6.68

Note the difference between the *average* and *largest* component sizes in each case. For *minProb* = 0.70 and lower values, the networks consist of one or a few (very) large components and hundreds of small ones. In a second experiment, we used the average value of the similarity functions as an estimate for the probability (hence the edge weight) instead of their product. In that case, large components already occur for *minProb* = 0.80. We conclude that the problem size explosion (as expected) strongly depends on the *minProb* value and that we shall be prepared to solve the scalability problem.

## 5. Related work

[3] describes an agent-based model aiming to optimally combine demand and supply in an advisory system for *repeated ride-sharing*. The authors focus on the mechanisms required to model users cooperating on joint plans and focuses on the economic value of the shared plans; this research focuses on the fairness of the payment system but does not consider the rideshare demand and supply change in time. [4] focuses on dynamic non-recurring trips which is related to commuting carpooling but requires different solution concepts. Both maximal individual advantage and system wide optimum are considered. The paper presents a crisp problem specification. It suggests that integer programming optimisers can turn out to be insufficiently performant to solve practical problems. [5] derives *travel routine* from sets of GPS traces; the routines are matched to find an upper bound for possible carpooling. [2] investigates for the region of Toronto-Hamilton(Ca) what are the driving factors behind carpool formation: age, gender and income category are the only relevant factors. Recent carpool advisors (like <http://www.zimride.com/>) take additional factors into account for matching candidates (interests, music tastes) and allow for feedback to be posted.

## Acknowledgements

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement nr 270833.

## References

- [1] A. Schrijver, A Course in Combinatorial Optimization, CWI, Kruislaan 413, 1098 SJ Amsterdam, The Netherlands and Department of Mathematics, University of Amsterdam, Plantage Muidergracht 24, 1018 TV Amsterdam, The Netherlands., Amsterdam, 2012.
- [2] R. Buliung, K. Soltys, C. Habel, R. Lanyon, The Driving factors behind successful carpool formation and use, Transportation Research Record (2118).
- [3] E. Kamar, E. Horvitz, Collaboration and shared plans in the open world: Studies of ridesharing, in: Proceedings of the Twenty-First International Joint Conference on Artificial Intelligence, 2009.
- [4] N. Agatz, A. Erera, M. Savelsbergh, X. Wang, Sustainable passenger transportation: Dynamic Ride-Sharing, Research Paper ERIM Report Series Reference No. ERS-2010-010-LIS, Erasmus University of Rotterdam Erasmus Research Institute of Management (Mar. 2010).
- [5] R. Trasarti, F. Pinelli, M. Nanni, F. Giannotti, Mining mobility user profiles for car pooling, in: Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD '11, ACM, New York, NY, USA, 2011, p. 11901198. doi:<http://doi.acm.org/10.1145/2020408.2020591>. URL <http://doi.acm.org/10.1145/2020408.2020591>