



**HAL**  
open science

# Dynamic Evolution and Optimisation of an Urban Collective Taxis Systems by Discrete-Event Simulation

Jennie Lioris, Guy Cohen, Régine Seidowsky, Habib Haj Salem

► **To cite this version:**

Jennie Lioris, Guy Cohen, Régine Seidowsky, Habib Haj Salem. Dynamic Evolution and Optimisation of an Urban Collective Taxis Systems by Discrete-Event Simulation. ITS World Congress 2016, Oct 2016, Melbourne, Australia. 10p. hal-01579724

**HAL Id: hal-01579724**

**<https://hal.science/hal-01579724v1>**

Submitted on 31 Aug 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Dynamic Evaluation and Optimisation of an Urban Collective Taxis System by Discrete-Event Simulation

Jennie Lioris  
IFSTAR  
(on leave from CERMICS-ENPC)  
Paris, France  
jennie.lioris@cermics.enpc.fr

Guy Cohen  
Researcher Emeritus at CERMICS  
École des Ponts-ParisTech  
Marne la Vallée, France  
guy.cohen@mail.enpc.fr

Régine Seidowsky, Habib Haj-Salem  
UPE-IFSTAR, Paris, France  
regine.seidowsky@ifsttar.fr;  
habib.haj-salem@ifsttar.fr

**Abstract**—We are interested in an independent, affordable demand responsive transportation structure involving high quality of service and comfort similar to individual cars. The reduced fares will be achieved by raising the number of car passengers while keeping vehicles as busy as possible. The high standard of service will be ensured by a smart allocation of passengers to vehicles whilst optimal car itineraries will be defined according to their present state and the current traffic conditions. Moreover, the detours which clients can tolerate will be controlled and client waiting times will be taken into consideration. The system covers an entire urban area (including the suburbs), ensuring autonomous door-to-door services and flexible operational modes while it is destined for all classes of commuters. Various strategies on the system management and dimensioning will be examined and each resulting performance will be appraised (in balance with the pending costs). Metrics on client detours, client waiting times, vehicle occupancy etc. are provided. Optimisation of all the real-time controls governing the system (e.g. client acceptance, vehicle itinerary, idle vehicle management etc.) will be achieved within a virtual but reliable environment by a made-to-measure discrete event decision tool.

**Keywords**—Demand responsive transport; travel demand management; discrete-event simulation; Monte Carlo simulation; Poisson arrival processes; routing algorithms; Pareto optimality; performance evaluation; parameter optimisation;

## I. INTRODUCTION

### A. Principal Arguments for Collective Taxis

Efficient transport systems can be regarded as essential to the requirements of urban development and productivity. Tentative proposals such as car sharing or pooling have been put forward to reduce the dependency of individuals on their automobiles or imposing deterrents such as vehicle prohibition and tolls for entering city centres. All these initiatives, mostly remain partial solutions, often encouraging the driver willingness to bear additional costs to ensure comfort, safety and convenience. In this paper, an innovative transportation mode is proposed, called “collective or shared taxis”. More precisely, spare vehicle capacity will be used in an optimal way with the purpose of providing similar quality of service to classical taxis while associated with reduced fares (“sharing” being the means of reducing the fares). The idea

of associating more than one passenger with each vehicle already exists, but often various limitations are encountered (fixed routes [1], advanced booking [2]), services often at the disposal of particular groups of citizens, etc. The present research is an extension of a study initially proposed by [3] in which the authors were interested in providing a complete alternative to individual vehicles covering a city centre and its suburbs whilst releasing clients from many constraints: no prior reservations (decentralised approach) nor fixed pick-up points (where the term “innovative” transportation mode) but dynamically constructed itineraries depending upon the passenger destinations.

### B. Operating Modes

The behaviour of the “Collective Taxi” system will be appraised according to the following three possible approaches:

- **the decentralised management** where no pre-requisite seat reservation is involved
- **the centralised management** dealing only with clients who have previously confirm a seat reservation
- **the mixed management** where we are dealing with both types of clients.

We can then proceed to a comparison of the resulting performance of each mode, checking the related benefits and drawbacks and consequently select the most favourable approach. Although both models are introduced in this paper, evaluation of the the decentralised approach is developed.

## II. MODEL OVERVIEW

The “collective taxi” transportation structure forms a dynamic complex system, presenting asynchronous behaviour while uncertainty, synchronisation and concurrence elements are encountered. It is unnatural and inefficient to consider a particular time step to govern the system state transitions. An event-driven approach is considered specifying the occurrence of events as the only responsible entities for driving all dynamics.

In a discrete event model, the system evolution is represented as a chronological sequence of the form  $\{\dots, s_i, e_i, s_{i+1}, e_{i+1}, \dots\}$ ,  $i = 0, 1, \dots$

In such a formulation, the system state at time  $t_i$  is  $s_i$  and  $e_i$  is the event occurring at time  $t_i$  bringing the system to the new state  $s_{i+1}$  and so forth.

### A. Modelling Framework

Approaches based on Petri nets, dioid algebras, Perturbation Analysis, simulation techniques etc. can be utilised for the study of event-driven structures.

In the considered “collective taxi” model, where most of the ruling constraints have been removed in order to increase its *flexibility*, it is difficult or almost impossible a precise comprehension without a reliable virtual representation of the system behaviour. Direct experience on a “trial and error basis” would imply risking unnecessarily high financial costs and possibly irreparable damage to customer relations due to client dissatisfaction. A combination of mathematical methods such as Optimisation, Operational Research and advanced computer science techniques, is required for a refined model representation and evaluation.

Computer simulations are often used where deterministic solutions are intractable. They form a viable and effective research tool allowing the many aspects of complex problems to be explored, reproducing precisely all stochastic features which inevitably randomly occur. Answers to all “what-if” questions, such as “how much time does the system spent at a particular time”, “how soon can a particular state be reached”, etc. can be provided within secure environment and master the system without the consequences of poor decision making. Moreover, simulation is the only mean for reproducing various scenarios with just a single factor modified at each run. This is a crucial condition for tuning optimal parameter values.

### B. Designing the Simulator

With the purpose of providing the best policy for any demand level and geometry, various operational schemes and control algorithms will be experimented. Their manifested influence on the system behaviour will be studied, even within the most complex conditions, not simply under ideal or trivial situations.

With this aspect, the simulation program is split into two parts depending upon the nature of the tasks (see Figure 1). These two parts are independent but closely interacting with each other.

- **The mechanical part** forms a virtual representation of the real system. It receives two types of entry deterministic and stochastic parameters (demand, traffic condition etc.) and controls ruling the system, in the form of a feedback.
- **The real time management** corresponds to the system intelligence. It is comprised of all the decision algorithms responding the enquiries made by the mechanical part. In addition to that, design and dimensioning

decisions are being considered at the stage of data input, at every run of the simulator.

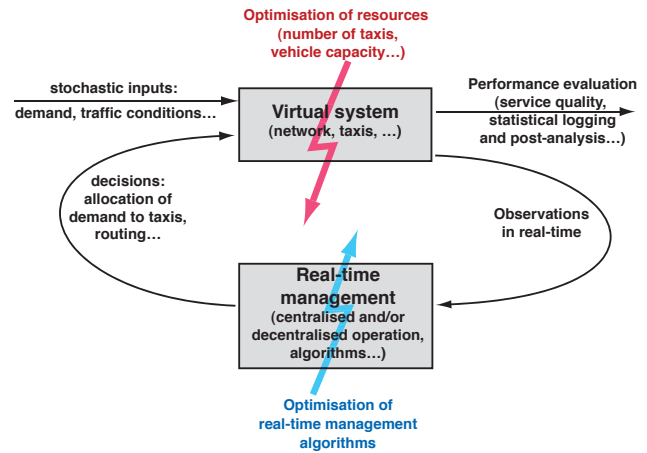


Figure 1. The simulator architecture

### C. Formalising the System Behaviour

In this section, it is described how state transitions are generated and the interaction of the system entities (clients, vehicles, ...) is specified. Each modelling event is comprised of the time at which it occurs, its type and the system entities involved (clients, vehicles, network nodes, etc). For each type of event, a specific procedure is defined related to its treatment. In order to get empowering simulation results, a sophisticated model closely representing the real world system is necessary. However, if too many simplifications are taken into consideration, we will end up with unrealistic situations whilst, conversely, the use of a complex model will lead to extremely complicated results, unlikely to be studied and understood.

A brief description of the modelling events is presented.

#### 1) Events generated by decentralised clients

**Client appearance.** A new client arriving somewhere in the network, wishes an immediate departure. If an available vehicle is found a dialogue begins. Otherwise, the client decides whether and for how long he/she will wait.

**Client enters the vehicle.** At this time and after a positive answer concerning the client acceptance to a given vehicle, the client enters it.

**Client abandons the system.** If no vehicle is found before the max waiting period of the client is reached, the client quits the system (and the structure is penalised).

#### 2) Events generated by centralised clients

**Client makes a request.** This event represents the client behaviour when he/she joins the dispatching center aiming at a seat booking. If an available

dispatcher/server is immediately found an information exchange begins; otherwise the customer decides whether and for how long he will wait.

**Client abandons the dispatching center.** If no controller has taken the client request by this time, the client quits the system without being served.

**Client arrives at meeting node.** At this time the client arrives at the pickup point and searches for the associated vehicle. In case of an absent absent, the client decides whether and for how long to wait.

**Client abandons waiting at the meeting node.** At this moment, the client quits his/her origin node.

**Reservation cancellation.** It is possible that, a customer desires to cancel a previously confirmed reservation. If a dispatcher (or server) is found, a dialogue starts; otherwise the client decides whether and for how long he will wait.

**Clients enters the vehicle.** At this time, one or more passengers enter(s) the vehicle at the pick up point. The car may depart or wait for other customers to arrive, according the defined vehicle max waiting period.

### 3) Events initiated by vehicles

**Vehicle starts its service.** From this time, the vehicle is at the client disposal.

**Vehicle finishes its service.** At this moment, the vehicle ceases its service. This event takes place only if the vehicle has completed all its tasks.

**Vehicle arrives at a node.** This event describes the vehicle operations every time it joins a node.

**Vehicle completes the dialogue with a decentralised type of client.** At this point, the response for the client acceptance or refusal is obtained.

**Clients disembark the car.** The current position of the vehicle is the destination of one or more of its passengers.

**Empty vehicle considers alternative parking.** Every time a vehicle has an empty itinerary, a node is chosen to park during a specified time. If by this moment the vehicle has found no client, it requests for its future planning.

**Vehicle ceases waiting for missing centralised clients.** At this moment the vehicle quits waiting for absent clients with whom an appointment is planned.

### 4) Events generated by the dispatchers or servers

**Dispatcher is on-duty.** From this time, the associated dispatcher or server is at the client disposal.

**Dispatcher is off-duty.** From now on, the controller is out of service. This event will take place only if the dispatcher is not occupied with a call.

**Dispatcher finishes a call.** At this moment, the answer for the client acceptance/refusal is provided. In case of acceptance, the corresponding meeting information is given (vehicle id, pick up point and time period etc.).

Potential vehicle updates may be required in this case.

### D. Simulator Engine-Exploitative Possibilities of the Simulator

According to the needs of the study, the following simulation options are available.

- Either a *new simulation* will be initiated. In this case the initial state of the system is empty and vehicles must be placed at nodes and clients have to appear so as their interaction will begin.
- Alternatively, a previously completed simulation can be extended (e.g. case when the duration of the initial simulation is not enough to draw conclusions, etc.).

For each of these two possibilities, the choice of creating new demand or utilising a previously generated one is possible, (for instance, when analysing how the same clients behave under different management strategies).

### E. Brief Description of Simulation Data.

Hereafter, the necessary input data required by the simulator are briefly described.

**The network** in which the collective taxis system will be applied: its topology comprised of nodes and edges (directed graph) and the probability laws defining the travel times at different periods of the simulation duration (representing congestions, off peak hours etc.).

**The demand** is composed of probability distributions for :

- the client appearance on the network (stated for all types of clients with or without reservations),
- the client destination,
- the duration of the client maximal waiting time at each node, before the final abandonment.

The demand is characterised by:

- a vector  $\lambda$  defining the intensity of client arrival at nodes. The client appearance follows a Poisson process ( $\lambda_i$  is the average number of clients appearing per time unit at node  $i$ );
- the origin-destination matrix  $M$ , defining the geometry of the demand. The entry  $M_{ij}$  indicates the probability of choosing node  $j$  as destination when the origin is node  $i$ .

Multiple types of demand geometry are explored in association with various levels of demand intensity. It worths to highlight the *centripetal* geometry of demand in which clients move mostly from the outskirts to the centre of the network, the *centrifugal* one when the periphery is more attractive than the town centre and finally the *balanced* demand where the client attraction is equal to the client emission at each node. Combination of *centripetal* and *centrifugal* demand allow the study of the system behaviour during an entire day (early in the morning when a large majority of travels correspond to people going to work and later in the evening when the population comes back home).

Moreover, the following information needs to be modelled:

- the service duration of all resources (vehicles, dispatchers or servers etc.). These values can be dynamically decided during the simulation, depending upon the current system state. Potentially, these durations can also be pre-defined and provided as input data
- the duration of various operations (dialogues, client embarking/disembarking etc).

#### F. System Performance and Statistics.

#### G. Simulation Results Treatment

All proceeded events are recorded in a database the analysis of which is performed at a second stage after the completion of the implementations.

- **A Micro analysis** allows an individual examination of all agents (nodes, clients, vehicles, edges etc.).
- **A Macro analysis** provides global statistics such as mean values, variances, histograms etc. on vehicles, customers, node statistics and so forth.

### III. DECISION ALGORITHM FOR CLIENT ACCEPTANCE IN THE DECENTRALISED APPROACH

Among the crucial algorithms composing the decision part of the simulator are those relative to the acceptance of clients to vehicles in the decentralised mode, as well as the ones choosing the best suited vehicle for a centralised type of client. Hereafter, an algorithm for the decentralised client acceptance to a given vehicle is presented. However, one can consider a whole set of algorithms and this is where one can see some of the simulation strong points. Each version of each algorithm can be tried and tested so as to choose the one providing the most desirable performance.

We are dealing with the following problem: a vehicle with a certain number of passengers on board having a given itinerary comprised of destination nodes arranged in an optimal way, meets a prospective client with a given destination (possibly already included in the itinerary of the vehicle). A binary decision has to be taken as to whether this potential client will be accepted or refused and in case of acceptance the vehicle itinerary has to be redefined. A possible rejection of the candidate customer is due to the impossibility of meeting the constraint of a “maximal detour” tolerated by each passenger, including the candidate himself. In case of acceptance, the chosen itinerary will be the one optimising a given cost function related to the arrival times of the passengers prioritising the nodes with the greater number of disembarking passengers. In order to mathematically formulate this problem, we first introduce some notation.

- $t_0$  present (meeting) time;
- $n_0$  present node;
- $n_i^o$  origin of passenger  $i$  (already on board);

- $n_i^d$  destination of passenger  $i$ ;
- $c$  index of candidate to getting on board;
- $n_c^d$  destination of this candidate (his origin is of course  $n_0$ );
- $L = \{n_1, n_2, \dots, n_m\}$  sorted list of destination nodes of passengers according to the planned itinerary (prior to acceptance of the candidate);
- $\ell = L \cup \{n_c^d\}$  unsorted list of destinations in case the candidate would be accepted (one part of the decision consists in finding the best order with which to visit those nodes);
- $J$  index set of elements in  $\ell$ ; namely,  $J = \{1, \dots, M\}$  where  $M$  is the length of  $\ell$  (indeed  $M = m$  or  $m + 1$ , according to whether  $n_c$  was already present in  $L$  or not);
- $I$  index set of vehicle passengers and candidate  $c$  ( $I = I \cup \{c\}$ );
- $d : I \rightarrow J$  mapping which provides the destination nodes of clients, that is,  $\forall i \in I, n_i^d = n_{d(i)} \in \ell$ ;
- $p(j)$  for  $j \in J$  is the number of passengers whose destination is node  $j$ , that is,  $p(j)$  is the size of  $d^{-1}(j)$ ;
- $\delta(a, b)$  duration (in time units) of the direct trip from node  $a$  to node  $b$  through the graph by the shortest path using average travel times (this matrix is pre-computed and part of the data);
- $t_i^o$  time at which passenger  $i$  got on board (for  $i = c, t_c^o = t_0$ );
- $t_j^p$  forecasted arrival time at destination  $n_j \in L$  prior to acceptance of the candidate:

$$t_j^p = t_0 + \delta(n_0, n_1) + \sum_{\substack{n_k \in L \\ k=1, \dots, j-1}} \delta(n_k, n_{k+1}) ;$$

- $s$  the diversion threshold borne by each client with respect to his direct travel. This is a parameter to be optimised as we shall see below;
- $t_j^{\text{lim}}$  deadline for arrival time at node  $n_j \in \ell$ :

$$t_j^{\text{lim}}(j) = \begin{cases} \max \left( t_j^p, \right. \\ \quad \left. \min_{i \in d^{-1}(j)} (t_i^o + s \times \delta(n_i^o, n_i^d)) \right) & \text{if } n_j \in L, \\ t_0 + s \times \delta(n_0, n_c^d) & \text{if } n_j = n_c^d \text{ and if } n_c^d \notin L. \end{cases} \quad (1)$$

Equation 1 is underlying that the serve time of each passenger must not exceed the accepted ratio of detour in proportion of his direct travel (that which would result of the use of a classical taxi). This constraint however may be impossible to be satisfied due to reasons such as past stochastic travel times, delays etc. Consequently, in the above definition, the maximum between the predicted arrival time (before any new passenger is accepted, updated according to the know conditions at  $t_0$ ) and the allowed

arrival time taking into account the maximal detour ratio is considered.

A mathematical formulation of the decision problem as a dynamic programming problem (this is inspired from [4]) is going to be presented. The purpose consists in finding an order for visiting the nodes in  $\ell$  such that the deadline of each node is respected whilst a certain objective function (defined hereafter) is minimised. If no feasible solution is found, the candidate must be rejected (and the planned taxi itinerary is not changed). Otherwise, a new itinerary is defined by this order (including the possible additional stop  $n_c$  in  $\ell$ ).

The whole itinerary requires  $M$  moves between the  $M + 1$  nodes in  $\ell \cup \{n_0\}$ ; those stages are indexed by  $k$  from 0 to  $M$ . Define:

- $x(k), k = 1, \dots, M$ , a (considered) sorted sequence of nodes in  $\ell$ ; in addition, set  $x(0) = n_0$ ;
- $u(k), k = 0, \dots, M - 1$ , choice of the next node to visit once  $x(k)$  has been reached; thus  $x(k + 1) = u(k)$ ;
- $\mathcal{E}(k)$  set of nodes already visited at stage  $k$ ; consider that  $\mathcal{E}(0) = \emptyset$ ; of course  $\mathcal{E}(k + 1) = \mathcal{E}(k) \cup \{u(k)\}$ ; in order that all nodes be visited after  $M$  moves, we require that  $\forall k, u(k) \notin \mathcal{E}(k)$ ;
- $t(k)$  arrival time at  $x(k)$ ;  $t(0) = t_0$  and  $t(k + 1) = t(k) + \delta(x(k), u(k))$ .

Finally, we aim at solving the following problem:

$$\begin{aligned} & \min_{u(\cdot)} \sum_{k=1}^M p(x(k)) t(k) \\ \text{s.t. } & x(k + 1) = u(k), \quad k = 0, \dots, M - 1, \quad x(0) = n_0, \\ & \mathcal{E}(k + 1) = \mathcal{E}(k) \cup \{u(k)\}, \quad k = 0, \dots, M - 1, \\ & \mathcal{E}(0) = \emptyset, \\ & t(k + 1) = t(k) + \delta(x(k), u(k)), \quad k = 0, \dots, M - 1, \\ & t(0) = t_0, \\ & u(k) \notin \mathcal{E}(k), \quad k = 0, \dots, M - 1, \\ & t(k) \leq t^{\text{lim}}(x(k)), \quad k = 1, \dots, M. \end{aligned}$$

#### IV. MANAGEMENT OF IDLE VEHICLES

Each time a vehicle is left with an empty itinerary at some network node, a station location will have to be assigned.

The designated node is defined according to a probability law constructed by taking into consideration both the distance of the present vehicle location from the parking node and also the client arrival intensity of the candidate node. In other words, an empty vehicle has a preference to chose nodes to park where the chances of finding prospective clients are greater, but preferably in the neighbourhood of its position, so as to avoid long journeys with no passengers on board. An empty vehicle parks at a node for a maximal period of time, and if no client was found within this time, the vehicle makes another request.

#### V. RUNNING THE SIMULATOR

In this section, we are not demonstrating a thorough optimisation procedure. We are simply aiming at showing the enormous variety of information that a single simulation can provide. At this stage, we explore the decentralised management in which clients appear randomly on the network looking for a vehicle for an immediate departure.

##### A. Case study

The data utilised during the studied implementations are going to be introduced. At present, results employing real data could not be published, consequently we had to construct fictitious but sufficiently realistic ones so as to proceed to the system assessment for the applied policy. We wish to make it clear that the data are merely inputs for the simulator and in no case they compromise the presented methodology evaluating the system performance.

1) *Network*: A network inspired by the Paris metro plan (288 nodes and 674 edges — see Figure 2) is employed. The travel time at each edge follows a shifted log-normal distribution (the shift ensures a minimal nonzero travel duration). On each edge, the shift represents 80% of its mean travel time. The mean travel time is proportional to the length of the edge.

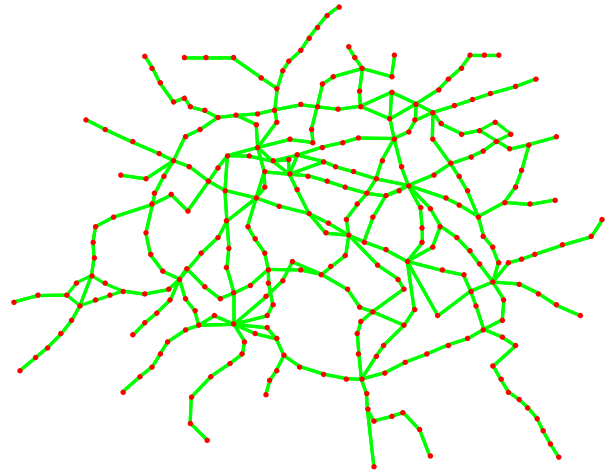


Figure 2. The network

2) *Demand*: For this run, in average 15,393 clients per hour are generated on the entire network, according to a predominantly *centripetal* movement.

3) *Duration of operations*: The duration of each dialogue between a client and a vehicle is of 30 seconds independently of a potential acceptance or refusal. The maximum waiting time of each client at the origin node is 10 minutes and the maximum parking duration of an idle vehicle is of 15 minutes. We have allowed 10 seconds for each client embarking/disembarking.

4) *Resources*: We are employing 13 taxis per node (3,744 vehicles in total) initially uniformly allocated, but this distribution rapidly changes as soon as the simulation starts. Each vehicle has 5 available passenger seats.

5) *Diversion Threshold*: The acceptable detour threshold of a client, regarding his direct travel, is  $s = 1.9$ .

6) *Simulation Duration*: Simulation duration is of 8 hours (real time) and after multiple verifications, this duration is proved to be long enough to provide statistical results with an insignificant variance (stochastic model).

**B. Some Statistical Observations for a Single Run**

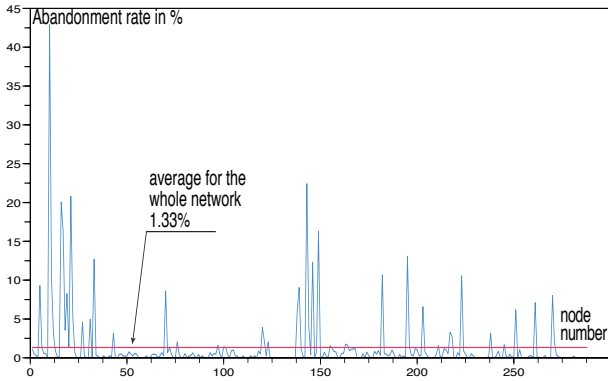


Figure 3. Abandonment rate at each node

1) *Client Abandonment Rate and Waiting Time*: Figure 3 represents the mean abandonment rate at each node. The average value is of 1.33% for the entire network.

Furthermore it would be of interest to know the client waiting time at each node of the network.

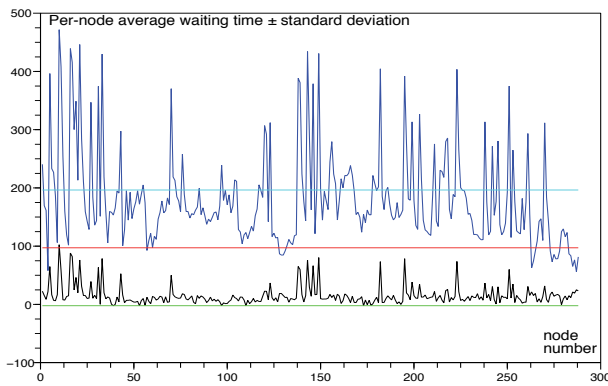


Figure 4. Mean client waiting time at each node

Figure 4 shows the mean client waiting time (plus/minus the standard deviation) at each node whilst Figure 5 is colour coordinated in which darker shades represent higher values of waiting time. As one can see, it is mostly the outskirts

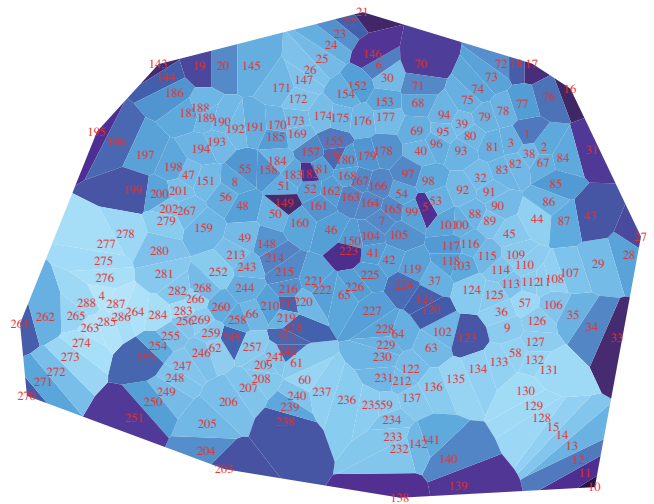


Figure 5. Map of mean client waiting time at each node

of the town which present increased values of client waiting times. However, some very dark shades are located close to the town center. This phenomenon is not normal for a centripetal movement where most of the entities (clients, vehicles) move towards the city center.

Let us consider the dark coloured node 149 and explain the reasons causing this result. Examination of the corresponding client abandonment rate implies increased values as well. We proceed to a *micro analysis* for this node.

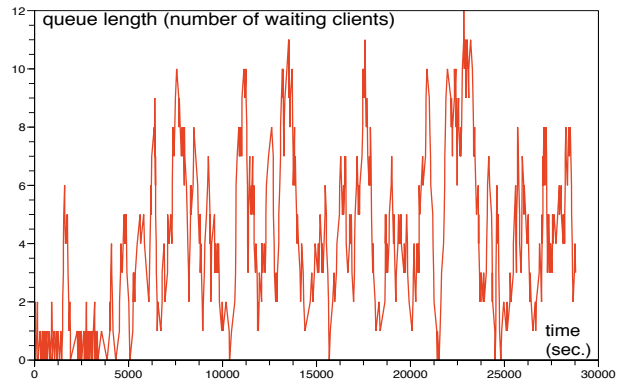


Figure 6. Evolution of client queue at node 149

Figure 6 shows the evolution of the client queue length at node 149 during the entire simulation duration while Figure 7 illustrates the histogram depicting the percentage of time with  $n$  clients in the queue. One observes that the client queue size frequently reaches high values at this node. After a deeper examination, we came to the conclusion that this is mainly due to the geometry of the network. In Figure 8, we show nodes in the vicinity of node 149

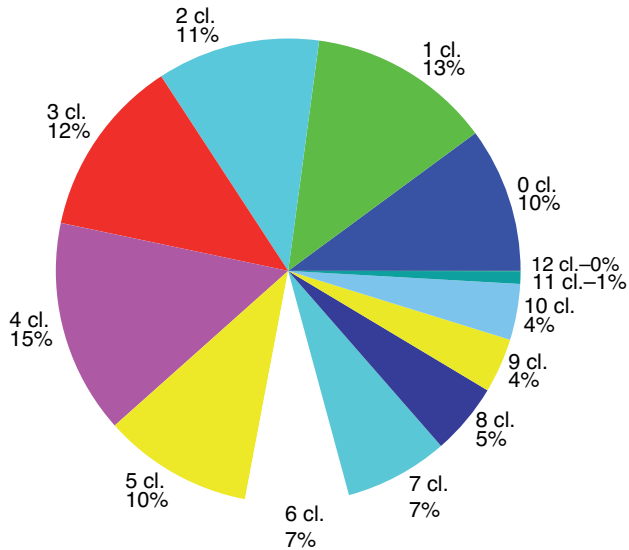


Figure 7. Histogram of client queue length at node 149

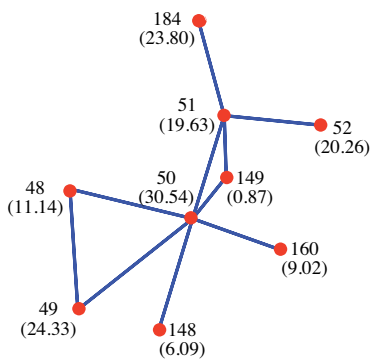


Figure 8. Frequency of vehicle passage per minute at the neighbourhood of node 149

with their node number and (in parenthesis) the frequency of vehicle passages for each node per minute. We see that there are only 0.87 vehicles passing through node 149 per minute. As Figure 8 shows, there are two possibilities for going from node 51 to 50: either by choosing the direct edge (most popular choice) or by passing by the intermediate node 149. Moreover links joining these nodes (51, 50 and 149) are mostly one-way, so vehicles can not pass by easily. Finally we conclude that the high abandonment rate at this node is mostly due to its inaccessibility. This example manifests how a refined and consequently efficacious analysis can detect even minor negative features and potentially improve the system behaviour.

2) *Probability of client acceptance by vehicles:* In Figure 9, we present the probability that during a dialogue between a vehicle and a candidate client, there are 0, 1, ..., 4 passengers on board (the vehicle capacity is 5 passenger

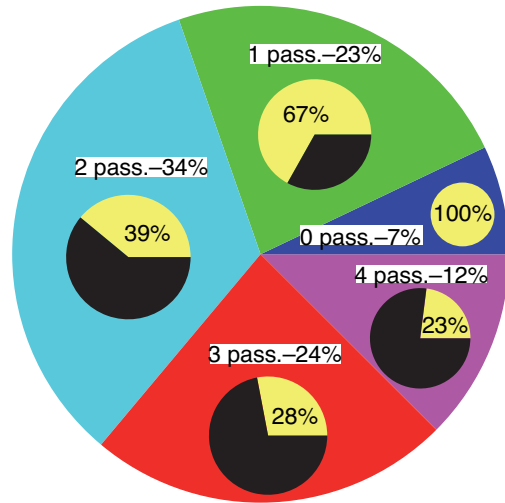


Figure 9. Conditional probability of client acceptance

seats) and the conditional probability of the client to be accepted for each of these cases. For example the conditional probability of a client to be accepted when there are 0 passengers on board is naturally 100%, and this case has a 0.07 probability to be realised.

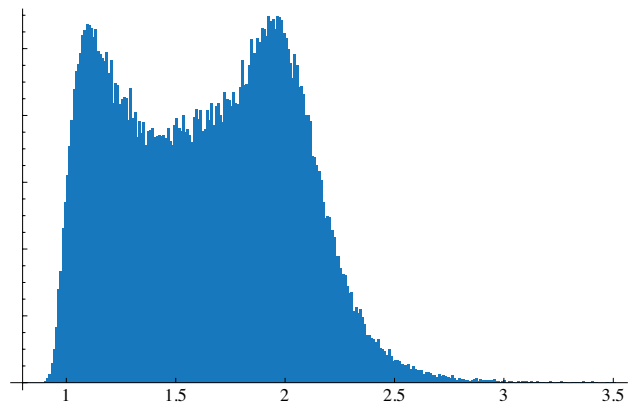


Figure 10. Total detour histogram

3) *Service Quality and System Reliability:* The provided quality of service as well as the system reliability can be measured by the *total detour*, characterising all passengers brought to their destination. It is defined as the ratio of the sum of the client initial waiting time (if any) plus his effective trip duration over his direct trip duration (the one he would have made if he had chosen a classical taxi or a private vehicle).

Figure 10 displays the histogram of the total diversion ratio. One observes that for the majority of passengers their detour tolerance has been respected (recall that the diversion threshold for this run is  $s = 1.9$ ) with a mean total detour



value approximately equal to 1.64. There are even some clients who have a total detour ratio less than 1. This is due to the fact that travel times are stochastic: those clients probably had a zero waiting time and they were lucky enough during their trip. Nevertheless, one also observes that there are clients having total diversion ratios greater than 2. Going along a tentative to explain the reasons of this result, we observed that these are mostly clients having very short trip durations (perhaps they should not consider taking a taxi for such a short distance).

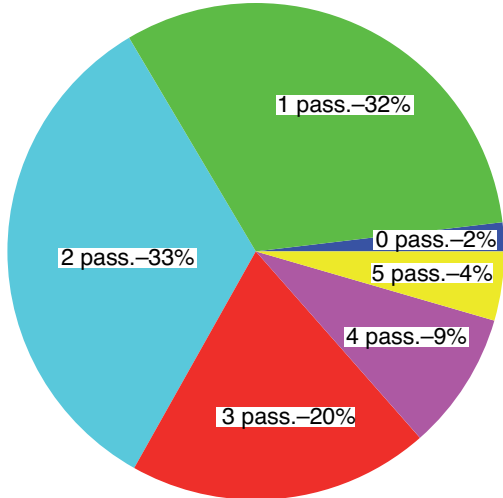


Figure 11. Percentage of time taxis travelled with 0 to 5 passengers

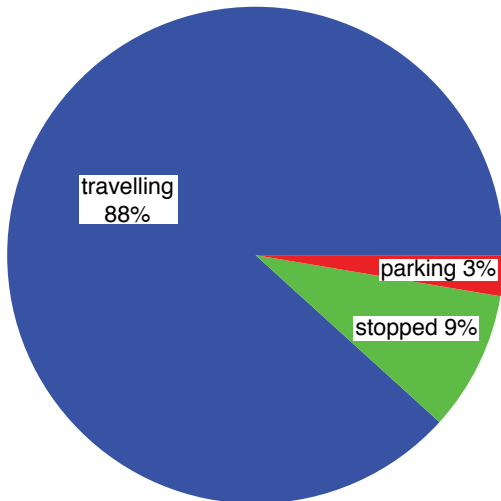


Figure 12. How busy taxis are

4) *How busy taxis are:* The histogram on Figure 11 represents the percentage of time with 0, 1, . . . , 5 passengers on board on all vehicles (while they travel on edges). Thus, during the 33% of the vehicle travelled time, there were two

passengers on board, while 4 passengers were during the 9% of the time. The vehicle travel time in an empty state is only during the 9% of the time.

In Figure 12, the blue (larger — 88%) area of the pie chart indicates the percentage of time spent by vehicles on travelling. The green part (medium area — 9%) shows the percentage of time during which vehicles were doing various operations at nodes (entering/disembarking passengers, examining candidate clients). Finally the red (smaller — 3%) area corresponds to the parking time of vehicles.

Obviously these statistics can be similarly calculated for each vehicle separately.

### C. Comparison Between Collective and Conventional Taxis

In this section, we are interested in examining a system of classical taxis and evaluating the related performance in comparison with a collective taxi system. In our model, classical taxis can be merely viewed as collective taxis with capacity 1. In other words, they always accept single clients whenever they are empty and cannot take any new trips before serving their unique passenger, since they are full (in real life, they may accept simultaneously a *group* of clients if those customers go together from the same origin to the same destination node. However, customer groups are not considered in the following implementations).

Our reference will be the simulation analysed at §V-A in which there were 3,744 taxis (of capacity 5) in service and we observed that the average number of passengers per taxi was 2.16. Therefore we are considering here exactly the *same sequence of generated clients* but with individual taxis ranging from 3,744 taxis in service to twice that number (7,488 and even three times more (11,232).

Table I summarises the results for those three cases and compare them with the case of 3,744 collective taxis in service of capacity 5. In terms of quality of service offered to clients (with respect to average waiting time or queue length, and abandonment rate), the only case of conventional taxis which can compete with the collective taxis structure is the case when we employ three times more vehicles in service (that is, 11,232 cars). Even in that case, the client abandonment rate is still almost three times larger. Of course, as one can expect it, the total diversion ratio is necessarily close to 1 with individual taxis (as long as the initial waiting time is not penalising this ratio heavily as this is the case with 3,744 taxis — second column — for example).

Hence, let us concentrate now on the comparison of a conventional taxi system (with three times more taxis in service) versus the collective taxi system studied earlier. Waiting times are comparable for both structures, abandonment rate is worse for classical taxis but total diversion ratio is necessarily better. Now, consider the activity of taxis and the corresponding costs (and consequently fares). In terms of the average number of clients transported per taxi (in 8 hours), this number is almost divided by 3 for the classical

Table I  
COMPARISON OF TAXIS WITH CAPACITY 5 VS. INDIVIDUAL TAXIS

	Collective taxis		Classical taxis	
Number of taxis in service	3,744	3,744	7,488	11,232
Abandonment rate	1.33%	28.95%	6.78%	3.53%
Mean client queue length	1.54	6.7	2.47	1.5
Mean client waiting time	97 sec	394 sec	135 sec	82 sec
Total diversion ratio	1.64	1.41	1.13	1.09
Total number of transported clients	112,883	82,628	109,481	113,494
Mean number of passengers per taxi	2.16	0.99	0.9	0.88
Mean number of transported clients per taxi (in 8 hours)	41	31	23	15

taxis (which is consistent with the total number of clients served, comparable in both cases, and the number of taxis in service).

*Remark 1.* One may wonder why the total number of transported clients in the simulation is slightly higher in the last column compared to the first column (since the sequence of generated clients is the same in all simulations and the abandonment rate is larger in the last column than in the first). This is because are considered as “transported clients” only those clients who effectively reached their destination *before* the end of the simulation. It is highly probable that there are more passengers still on board of taxis at the end of the simulation (and thus not counted as “transported”) with taxis of capacity 5 than with taxis of capacity 1.

(idle time) is only 3% of the total time of simulation whereas it jumps to 46% with conventional taxis (employing 11,232 cars).

## VI. CONCLUSIONS - FUTURE WORK

In this paper, a decentralised management of a “collective taxi” system is studied covering an entire urban area while offering door-to-door services. During this work, we have tried to preserve the provided quality of service compared to the one of classical taxis, maintaining reasonable diversions, and keeping low fares by smartly associating more than one passengers to each vehicle. The collective taxi system being a real world stochastic, complex dynamic system is heavily affected by a random behaviour, and difficult to be evaluated by real life applications. Discrete-event simulation is a promising mean allowing an accurate comprehension of the system behaviour at low costs and without any risk. After developing a decision tool specifically for the study of the “collective taxi” system, modelling each one of the three approaches (clients with and/or without reservations), we are able to construct multiple strategies, develop various control algorithms and appraise the resulting system performances. Thus, optimisation of all the controls ruling the system can be achieved and reasonable trade-offs between conflicting performance statistical indicators can be provided. The decentralised approach, discussed in this paper, consists of clients who randomly appear in the network, looking for a vehicle for an immediate departure. Every time a client meets a vehicle a decision algorithm responds whether the client can be accepted or refused, providing the corresponding optimal vehicle itinerary in case of a positive answer. The involved quality of service is measured in terms of client initial waiting time and detours. Metrics on the client abandonment rate are also studied. The system controls idle vehicles indicating station nodes with increased chances of finding new prospective clients. Furthermore, vehicle performance is examined in terms of occupancy, travel duration in empty state etc.

Comparison of the actual travel time and trip costs of the proposed “collective taxi” system versus other public transport modes in urban areas is an interesting element for characterising the urban mobility. Research in this direction

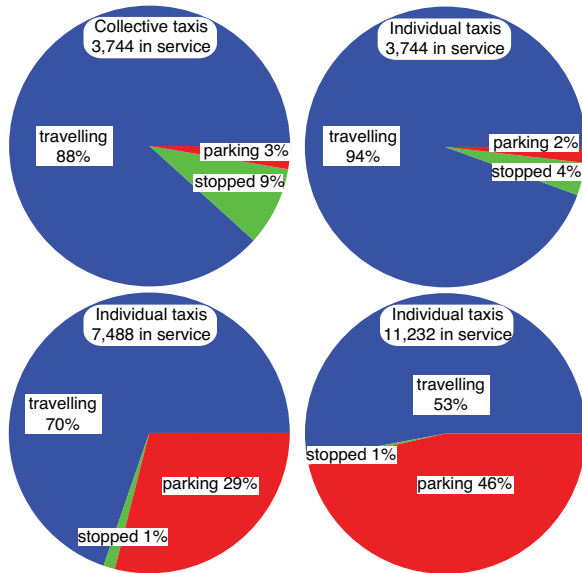


Figure 13. Collective and Conventional Taxis effectiveness

Figure 13 is also enlightening in understanding a fundamental difference in the behaviour of conventional and collective taxis. Considering the histograms of taxi activity corresponding to the first and last columns of the table (pie charts NW and SE in the figure), one can see that collective taxis are kept very busy all the time since the time of parking

is currently in progress. The development of more sophisticated decisions algorithms for the decentralised mode is already planned. Moreover, future work aims at the study of the centralised and mixed approaches. Measuring the benefits of each structure and consequently comparing the system performance with the aim to examine if and when settling up a dispatching center is worth the corresponding cost, is an interesting issue to be studied.

We believe that if a real application could be achieved, it would form an alternative to individual cars and classical taxis (without aiming at replacing these two transport modes).

#### REFERENCES

- [1] P. Lalos, A. Korres, C. K. Datsikas, G. Tombras, and K. Pappas, "A framework for dynamic car and taxi pools with the use of positioning systems," in *2009 Computation World*, Athens, Greece, November 15-20 2009.
- [2] C. Tao, "Dynamic taxi-sharing service using intelligent transportation system technologies," in *International Conference on Wireless Communications, Networking and Mobile Computing (WiCom 2007)*, Shanghai, China, September 21-25 2007, pp. 3209–3212.
- [3] P. Fargier and G. Cohen, "Study of a collective taxi system," in *Proceedings of the Fifth International Symposium on the Theory of Traffic Flow and Transportation*, G. Newell, Ed. University of California, Berkeley: American Elsevier, June 16–18 1971, pp. 361–376.
- [4] J. Tsitsiklis, "Special cases of travelling salesman and repairman problems with time windows," *Networks*, vol. 22, pp. 263–282, 1992.
- [5] F. Baccelli, G. Cohen, and B. Gaujal, "Recursive equations and basic properties of timed petri nets." in *Journal of Discrete Event Dynamic Systems*, 1992.
- [6] J. Xu and Z. Huang, "Autonomous dial-a-ride transit introductory overview," *Journal of Software*, vol. 4, no. 7, pp. 766–776, September 2009.
- [7] F. Ciari, M. Balmer, and K. Axhausen, "Large scale use of collective taxis: a multi-agent approach," in *12th International Conference on Travel Behaviour Research*, Jaipur, India, December 2009.
- [8] M. Diana and M. Dessouky, "A new regret insertion heuristic for solving large-scale dial-a-ride problems with time windows," in *Part B 38 (2004)*, 539557, 2004.
- [9] R. Dial, "Autonomous dial-a-ride transit introductory overview," *Transportation Research Part C*, vol. 5, pp. 261–275, 1995.
- [10] L. Fu, "A simulation model for evaluating advanced dial-a-ride para- transit systems," *Transportation Research Part A*, vol. 36, pp. 291–307, 2002.
- [11] S. Balemi, P. Kozak, and R. Smedinga, "Discrete event systems: Modeling and control." in *Birkhauser; Basel*, 1993.
- [12] G.J.Olsder, "On structural properties of min-max systems." in *In Guy Cohen and Jean-Pierre Quadrat, editors, 11th International Conference on Analysis and Optimization of Systems*, Springer Verlag, 1994.
- [13] R.Cieslak, C.Desclaux, A.Fawaz, and P.Varaiya, "Supervisory control of discrete-event processes with partial observations." in *IEEE Trans. Automatic Control*, 33(3):249-260, 1988.
- [14] F.Baccelli, G.Cohen, J.Olsder, and J.P.Quadrat, "Synchronization and linearity – an algebra for discrete event systems," in *Wiley, New-York*, 1992.
- [15] Y.C.Ho and X.Cao, "Perturbation analysis and optimization of queueing networks." in *Journal of optimization theory and Applications*, 19:559-582, 1991.