Using mobile phone data for Spatial Planning simulation and Optimization Technologies (SPOT)

Serigne Gueye[1], Babacar M. Ndiaye[3], Didier Josselin[3], Michael Poss[5], Roger M. Faye[2], Philippe Michelon[1], Cyrille Genre-Grandpierre[3], and Francesco Ciari[4]

[1] Laboratoire d'Informatique d'Avignon (LIA), Université d'Avignon, France
{serigne.gueye,philippe.michelon}@univ-avignon.fr
[2] Laboratoire de Traitement de l'Information (LTI),
ESP- Cheikh Anta Diop University, Dakar, Sénégal
roger.faye@ucad.edu.sn
[3] Laboratoire de Mathématiques de la Décision et d'Analyse Numérique (LMDAN)
FASEG-University of Cheikh Anta Diop, Sénégal
babacarm.ndiaye@ucad.edu.sn
[4] Institute for Transport Planning and Systems (IVT), Zurich, Switzerland
ciari@ivt.baug.ethz.ch
[5] UMR Etude des Structures, des Processus d'Adaptation et des Changements de l'Espace (ESPACE), Avignon, France
{didier.josselin,cyrille.genre-grandpierre}@univ-avignon.fr
[6] Heudyasic, UTC, France. mjposs@gmail.com

Abstract. We propose in this paper a methodology to find locations or relocations of some Dakar region amenities (home, shop, work, leisure places), that may reduce travel time or travel distance. The proposed methodology mixes multi-agent simulation with combinatorial optimization techniques; that is individual agent strategies versus global optimization using Geographical Information System. We use MATSim as a multi-agent simulator system, and need for that to generate agent plans. Some additional methods are thus proposed to generate representative agent plans from mobile phone data provided by Orange. Some preliminary numerical results are presented on the Dakar region showing the potential of the approach.

Keywords: amenities location, multi-agent simulations, combinatorial optimization, local search, clustering, GIS, planning

1 Introduction

Many urban areas in the world, especially in developing countries, are faced to a rapid population density increase, that generates a transport demand that cannot be supported by transport infrastructures. Between 1976 and 2005, the population in the Dakar region had been multiplied by approximately 6 [7] while

[7] source : Enquête ménage CAUS/2001/PDU Dakar horizon 2025

in the same time the transportation network and the urban design was not sufficiently adapted to this change. It leads to congestion problems and a reduction of the urban **accessibility** defined as the capacity to reach some given resources or activities, within a given time. As a quantitative measure of the accessibility in a time interval, we call **global accessibility** in an urban area, the sum of the whole travel times or distances (for all the people) between the urban amenities.

When thinking about suitable actions to improve the accessibility, two dimensions are usually taken into account by planners: the transportation network design and the location of amenities. Indeed, people uses the transportation network motivated by activity objectives and places located somewhere in the urban area. Thus, to improve the accessibility to the facilities to allow these activities, one should improve both dimensions of the problem.

In 2007, a planning of the Dakar urban areas over the horizon 2025 had been performed by GMAT (Groupe Métropolitain en Aménagement des Transports) and CETUD (Conseil Exécutif des Transports Urbains de Dakar) (see [6]). This study, called "Plan de Déplacement Urbain de l'agglomération de Dakar-Horizon 2025 (PDUD)" contains a series of futur projects or recommendations, concerning each of this dimensions. For instances, among a very large list of projects, let us cite the construction of the highway "Patte d'Oie - Diamniadio" opened in 2013, that strongly improves the transportation network, the Diamniadio urban pole (4000 ha) whose construction started in 2014, located at 30 km of Dakar downtown, the closure of an old important inter-regional bus (Gare Pompiers), relocated in a a new more suitable and non-occupied place (Baux maraichers) in the suburb of Pikine (10 km of dakar downtown). Notice that the new activities that should take place in the old location is (to our knowledge) not yet clearly defined.

We observe that another possible relocation decision may be, instead of relocating this station in a non-occupied place, to exchange it with another existing amenities, thus solving in the same time the question to know what activities should be carried out in the former inter-regional bus station. For instance, switching with a significant commercial or shopping amenity with the inter-regional bus station would be possible. One may also consider not only relocating a single amenity, but rather finding the "best" relocation decisions, according to an objective of global accessibility optimization, that is to say relocating several amenities, in various non-occupied or occupied sites. A simple method could be to analyze all possible relocation scenarios. Nevetheless, as the number of amenities linearly increases, the number of possible scenarios increases exponentially, making intractable such an approach. This paper proposes a methodology by which a very large set of relocation possibilities can be simulated, analyzed, and the "best" one can be found, according to some quantitative measures. The methodology was coded in a prototype software called SPOT, that originated from two projects DAMA [1] and ORTRANS [12], and which is operating as follows.

Finding good geographical locations of amenities that optimize the global accessibility measure, supposes to be able to foresee, as realistically as possible, the trip flows induced by the users moving on the transportation network, between all amenities. In this task, SPOT uses the multi-agent simulator MATSim (see Balmer et al [2]). In MATSim, the actors of the modelled system are the agents (i.e the city residents). The agents act according to given "realistic" rules. They try to perform some activities at different places and have learning capabilities. The overall traffic observed in the urban area emerges from the simulation as a consequence of **individual agents behaviour**, each pursuing his/her **individual interests**. MATsim basically needs three data to perform a simulation: the transportation network (network.xml), the amenities location (facilities.xml) and the initial agent plans (plans.xml). At the first MATSim iteration, each agent follows one or several possible initial plans contained in the agent plan file. A plan takes place on **one day**. It is defined, at least, by a sequence of activities (with their geographic locations), and a list of traveling modes (car, bus, walk, bike,...) between all successive activities. For example, an agent can initially be at home, then goes to work by car, then goes shopping by walk and finally reaches a leisure activity by car before coming back home by car. Each agent initially choose a plan. All plans are then simulated by a traffic simulation module, that computes the different routes in the transportation network. Then, agents learn about the travel time or distance experienced during the chosen plans, and try, in the subsequent iteration, to optimize his/her plan (if necessary). He can for example change the transportation mean (car, public transportation, walk and bike), the activity schedules within a certain margin, or the locations of some leisure or shopping available places. The plan optimization is simulated by a genetic algorithm [9], that in fact only concernes 10% of the population. For each agent, some possible new plans are generated, and viewed as the components of a "genetic" population. As in any genetic algorithm, the population components (here the plans) can be crossed (cross-over), muted, and each solution is then evaluated (fitness function). The evaluation consists of giving a score to a plan, called the **utility**. Roughly, the utility is a function defined by the sum of the utilities to perform activities minus the disutility associated to the transportation cost (see Charypar and Nagel [7]). When new plans (eventually similar to the previous ones) are chosen by an agent, a new traffic simulation is performed. Then agents learn again from the new experiences, try to find other better plans, and so on... until a fixed number of iterations is reached. In theory, for an infinite number of iterations, the system converges in a Nash equilibrium state where each agent will choose a definitive plan (see Horni et al. [10]). That is a state where no agent will have some interest to change again its plan for increasing its individual utility. In practice, for a fixed number of iterations, the system has already been tested in more than 7 large cities (Zurich and complete Switzerland, Berlin and Munich, Padang, Gauteng, Toronto, Tel Aviv, Kyoto) and show a certain ability to reproduce real-life observations.

Following a complete MATSim simulation, in SPOT we adopt a **global (or collective)** view which contrasts with the individual behaviour of the agents in the simulation. Given the total amount or a very large ample of flows of Origin-Destination (O-D) trips observed between all amenities, our problem is indeed slightly different: it aims at finding some suitable relocations to increase the global accessibility for a set of selected amenities. Let us remark that the MATSim simulations are operating on only one day, such as the global accessibility we seek to improve. So, to be pertinent, the simulated plans should be as representative as possible of what the agent do most frequently.

The problem of finding a good relocation is viewed as a combinatorial optimization problem and solved using a local search algorithm. The new locations provided by the algorithm are then used to update the facility file, as well as the plan provided. A new MATSim simulation is performed, followed by a new step of location optimization and so on... until a fixed number of iterations. Contrary to the MATSim simulation current process, no theoretical results garantee that the whole iterating process handling individual agent interests (in MATSim) and the collective global optimization of the amenity (re)locations can converge to an balanced state. The figure 1 summarizes the SPOT methodology.

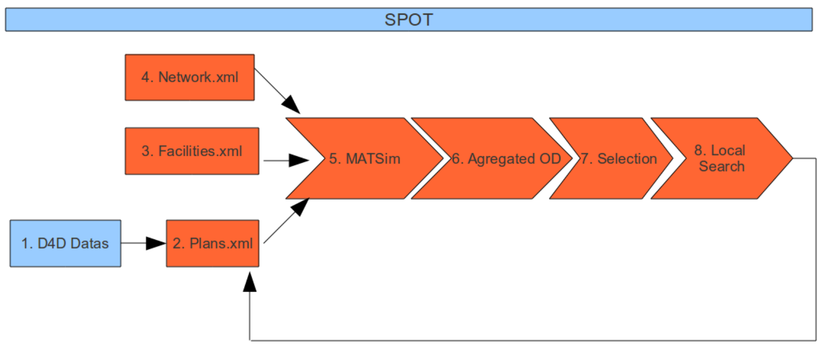

Fig. 1. SPOT

In the sequel, we detail in section 2 how the network and facilities files have been generated. In section 3, we show how the D4D challenge data were exploited to derive a representative initial plans for the agents. Section 4 deals with the computation of the O-D flows, the selection of the amenities to relocate and the local search procedure. Some preliminary numerical results are given in section 5. We then conclude this work and give some perspectives in section 6.

2 Network and Facilities Files

The network file is generated using Open Street Map (OSM) resources [8], in particular the OSM data for Senegal provided by the Humanitarian OpenStreetMap Team (HOT) [9]. Using the tool Osmosis [10] and the opensource Geographical System QuantumGIS, we separately extract the roads and the highways, and also a list of identified amenities with their geographical locations. Roads and highways poplate the file network.xml, and the amenities are used for the file facility.xml.

Most of time, the type of amenities in the list was not correctly annoted. We processed a semi-automatic assignment using specific requests in the QGIS data base. Thus, when necessary, the activity types was fixed to home, work, shop or leisure. In particular, for the "home" type, the amenities obtained from the OSM provide district names (as Fann, Point E, HLM,...) without (of course) indicating precise individual home location in these districts, as required in the MATSim plan file. For these districts, a spatial sampling constrained by resident area boundaries was then necessary to randomly generate a large set of home locations, respecting the density distribution of the Dakar region population in the different urban districts. Some informations about this distribution had been provided in the CETUD and GMAT document [6]. For instance, we learn in this study that, in 2007, the Dakar region working population was distributed as follows

Ville	Arrondissement	Population	Pourcentage (%)
Dakar	Plateau	215.343	8,71
	Grand-Dakar	253.434	10,25
	Almadies	121.006	4,90
	Parcelles Assainies	237617	9,61
Pikine	Thiaroye	239.053	9,67
	Dagoudane	461.648	18,68
	Niayes	209.859	8,49
Guédiawaye		435.350	17,61
Rufisque		160.860	6,51
Bargny		41.220	1,67
Sébikotane		19.400	0,78
Zone rurale		76.940	3,11
TOTAL		2.471.730	

Table 1. Extracted from the CETUD and GMAT report [6]

[8] https://www.openstreetmap.org
[9] *http://wiki.openstreetmap.org/wiki/WikiProject_Senegal*
[10] http://wiki.openstreetmap.org/wiki/Osmosis

Although the spatial 2014 distribution probably differs from this old one, we used the same proportion of locations, due to the fact it was our unique source of reliable information. Our goal being to process simulations with a maximum of 25000 agents, we distributed 25000 points (supposed homes with an agent) according to the previous percentages in the Dakar districts contained in the file SENEGAL-ARR.csv of the challenge data.

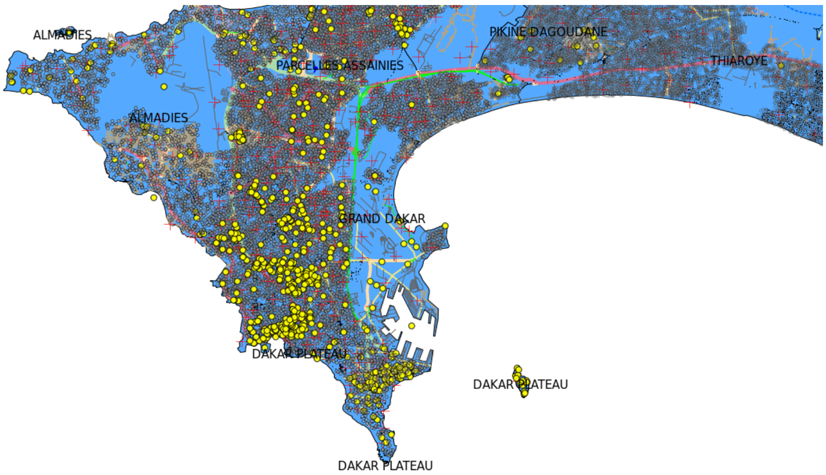

Fig. 2. Homes (grey points), facilities (yellow points), antennas (crosses), and district locations

The figure 2 was designed with the free Geographical Information System QGIS [11]. It uses shape files of the whole Senegal, as well as the different CSV files provided in the challenge. A new layer composed of the transportation network was added. The figure shows (yellow circles) several location types: amenities different from home, mobile phone antennas (red cross) and home locations (grey circles).

Finally and in order to construct the plan file, for each antenna we computed the list of all the amenities within a maximal distance of a given threshold. Those amenities are then easily accessible by an agent detected within a region covered by an antenna.

[11] http://www.qgis.org/en/site/

3 Generating MATSim plan files from mobile phone data

Generating initial plans is an important step generally performed using household surveys and population census. A basic MATsim plan for an agent looks like this:

```
. < person id = "pid120" employed = "no" >
.   < plan selected = "yes" >
.      < act type = "home" facility = "1202" x = " − 17.446145" y = "14.7382253" end_time = "11 : 10 : 00"/ >
.      < leg mode = "car" >
.      < /leg >
.      < act type = "shop" facility = "476" x = " − 17.388872" y = "14.7697" end_time = "15 : 30 : 00"/ >
.      < leg mode = "car" >
.      < /leg >
.      < act type = "home" facility = "1202" x = " − 17.446145" y = "14.7382253" end_time = "23 : 59 : 59"/ >
.   < /plan >
. < /person >
```

This example tells that the agent "$pid120$" lives at "1202", located at the geographic coordinates $x = " - 17.446145"$ and $y = "14.7382253"$. He leaves his house at $11:10:00$, by car, for shopping at the facility "476". He then leaves the shopping place at $15h30:00$ for coming back home where he stays until the end of the day.

The methodology presented in this paper is an attempt to substitute to the surveys and censuses, thanks to the exploitation of mobile phone data available in real time, while surveys require longer updating periods and are expensive in financial and human resources. For such a purpose, we are particularly interested in the challenge data set named SET2. Let us recall that the data are organized in 25 files, each file containing the list of visited antennas, over a period of 2 weeks, for 320,000 individuals, randomly selected. For each file, the sample of 320,000 individuals is renewed to ensure anonymity. Reading the files, it appears that the user detections had been made with a frequency of **10 mn**. Let us notice that from time to time, several antennas can be co-located nearby, so that a call can be supported over a short period, by several antennas.

Each file contains:

- `user_id`: the identifier of the person;
- `timestamp` (format YYYY-MM-DD HH:M0:00): the date and time during which the connection was made;
- `site_id`: the identifier of the antenna. A second file (SITE_ARR_LATLON.csv) allows to find the associated geographic coordinates.

A short example extracted from the file SET2_P01 is given below :

1,2013-01-07 13:10:00,461
1,2013-01-07 17:20:00,454
1,2013-01-07 17:30:00,454
1,2013-01-07 18:40:00,327
1,2013-01-07 20:30:00,323
1,2013-01-08 18:40:00,323

1,2013-01-08 19:30:00,323
1,2013-01-08 21:00:00,323
1,2013-01-09 11:00:00,323
1,2013-01-09 14:50:00,323
1,2013-01-09 15:10:00,318
1,2013-01-09 15:10:00,318
1,2013-01-09 15:20:00,318
1,2013-01-09 20:50:00,323
1,2013-01-09 21:40:00,323
1,2013-01-09 21:40:00,318
1,2013-01-09 21:50:00,318

From these data, we aim at generating some representative plans of daily trips of the Dakar (and suburbs) inhabitants during a **working day**. The methodology we propose is divided in several steps detailed below. Notice that in some steps (in particular the step 2), we introduce some concepts closed to a previously contribution in this topic (see Berlingerio et al. paper in [3]).

3.1 Step 1: Clustering of antennas

The step 1 deals with the problem of antennas co-localization, each being capable to detect, at almost the same time, a (or many) user(s). Thus it gives the illusion of a agregated movement. We tackle this issue, by grouping antennas in clusters using a standard hierarchical ascendant clustering algorithm (see [13] for a survey on clustering methods) applied to the file SITE_ARR_LATLON.csv. At the end of this algorithm, in each cluster, the maximal distance between any couple of antennas does not exceed a given threshold. Thus an agent successively detected by two antennas in the same cluster will be then considered as motionless.

An illustration of this process is provided with the agent 2 in the file $SET2\_P01$ who was detected, almost at the same time, by three different antennas, as described by the following lines:

2,2013-01-10 19:30:00,408
2,2013-01-10 19:30:00,416
2,2013-01-10 19:30:00,419

Computing the geographical distances between each of these locations gives a maximum distance of approximately 4.1 km.“ Taking as threshold the value 5 km will have the effect to put the three antennas in the same cluster. Thus, in this case, we consider that the agent was “stopped” somewhere in the area covered by these antennas.

3.2 Step 2: Generating individual trajectories

Definition 1. *For any agent j, we call* **"stop"***, noticed* $p^j = (c^j, s^j, e^j, a^j, l^j, m^j)$ *a time interval where j stay on a region covered by the antennas of a cluster c^j.*

A stop is characterized by a starting date (s^j), an ending date (e^j), the type of activity performed (a^j: home, work, shop, leisure,...), the geographic location of the performed activity, and the transportation mode used to leave the stop (m^j).

Definition 2. *For one day, we define a "trajectory" T^j, for an agent j, as an ordered sequence of stops : i.e* $T^j =< p^j_1, p^j_2, ..., p^j_n >$.

The step 2 consists initially of finding, for any SET2 file, the trajectories of the agents in each day, of the 15 possible ones, but without indicating the information at this step a^j, l^j, m^j. For a given SET2 file, finding the cluster (c^j), the starting and ending dates (s^j, e^j) is done by reading the file line by line. As long as two successives lines involve some antennas located in the same cluster, we consider that the agent "stops" in the corresponding area. For instance, coming back to our first example, the algorithm will find that the agent $j = 1$ "stopped" in the region defined by the cluster of the antenna 323 between 2013-01-08 18:40:00 and 2013-01-09 11:00:00.

Potentially, applying this procedure to the whole 25 files, may result in $320000 \times 25 \times 15$ trajectories. In sequel, we decided to build more reasonable set of trajectories that will considerably reduce the quantity of simulated plans used to derive a MATSim aggregated plan file.

3.3 Step 3: Finding stop activity

The step 3 tries to assign an activity type (a^j) to each stop. The available activity types are the following: home, work, shop, leisure. This assignment is done in a precise order: home first, then work, then shop and finally leisure.

For home and work activities, we adopt a process closed to the one used in Berlingerio et al. paper [3]. For each agent in a SET2 file, and each stop of the agent, we compute the number of hours passed in this stop during the night. If this number exceeds a given threshold, then we consider that the agent passed one night in the cluster associated to the stop. We then compute the total number of nights of each agent in each visited cluster, and retain the cluster with the highest number of nights. If this number exceeds a certain threshold, and if some home facilities exist "around" this cluster, then it is identified as its home location. Let us recall that when generating facilities files (see section 3), each antenna has been associated to a list of amenities at a maximal distance of a given threshold. By "around", we include all the amenities belonging to, at least, one amenity list of the cluster. A "home" amenity is then randomly chosen in these lists and it geographical location assigned to the attributes l^j. After that, all "sufficiently" (i.e exceeding a given threshold) long stop of the agent detected in this cluster will be considered as a "home" activity.

We proceed similarly for identifying the work activities, taking into account the cluster with the highest total number of working hours. The working hours for a given stop must be in a given fixed time period (between 6:00:00 and 18:00) where work is supposed to start and end. It must also exceed a certain given minimal threshold supposed to be a minimal amount a working times. Working activities must also exist around the cluster where the stop is located. We additionaly check that the best "working" cluster has not been yet fixed as a "home" location, assuming that (in general) working place and home are not co-located. If this case happens, we take the second cluster with the highest number of working hours. The remaining stops which are not identified as "home" or "work" are then fixed, when possible, as shop or leisure as follows.

We start by shopping activities. All stops in a cluster for which shopping amenities exist are assigned to shopping activities if the duration of the stop is large enough (in respect with a given threshold), and if the start and end of the stop is included in a given interval representing shop opening and closing times in Dakar. After this step, the last stops that are not identified as shops are considered to be "leisure", using the same criteria as before with different duration thresholds, and different start and end intervals. At the end of the process, all the stops without any assigned activity are erased. If by removing these stops, a trajectory of an agent becomes empty, we also erase the corresponding list. All these deleted data represent a significant reduction of combinatory in our numerical experiments.

Having obtained these purged data sets, since we seek to do a one day simulation, we choose for each agent a single trajectory among the existing ones, using different possible rules: randomly, from the longest list, from the longest list starting by a home activity (if it exists), from the longest list containing home and work activities. Let us notice that proceeding this way may lead to choose two lists dealing with two different days (for two agents). But what we want is something representative of the plans that the transport infrastructure should support. By choosing the longest list, for instance, we are interested in a kind of "worst case".

For each file SET2, we potentially obtain 320000 trajectories, each corresponding to one agent plan. This number, although being far away from the maximal number of trajectories of one file (320000×15), remains too high for our purpose. We drastically reduce it using a clustering steps detailed in the subsection 3.5. However, prior to this step, we assign a mode for each stop.

3.4 Step 4: Mode assignment

The goal of the mode assignment step is to fix the mode (m^l) that the agent was supposed to use for leaving a stop to reaching the next one. We consider three possible modes: car, public transport (pt) and walking. For each stop to which we want to assign a mode, we compute the agent **minimal speed** from a stop to the next one. This can be done by dividing the maximal distance between the origin cluster and the destination one, by the time difference between the instant where the agent leaves the stop, and the instant where he enters in the next stop. " This process gives one speed by stop, except for the last one of the moving chain. If the speed is greater than a given threshold, we consider that the mode type is "motorized" without precising at this step if it is "car" or "public transport". A speed below the same threshold is consider as "walk" only if the distance between the two stop clusters are "reasonable". That is to say below another threshold.

To determine the precise "motorized" mode, we associate to each agent an **average speed**, (i.e the sum of all speeds of its trajectory divided by the number of stops), and we use statistic information. We know, using a survey performed by CETUD [5], that in 2000 the number of car owners for 1000 Dakar habitants was 20. By which we can evaluate that in 2014 this number has been approximately increases to 30 cars for 1000 habitants, thus giving a percentage of 3%. The agents are then sorted in the decreasing order of their average speeds. We assign the mode "car" to the 3% faster motorized agents, and "pt" to the remaining motorized ones. All stops with no assigned mode are erased. If it happens that the trajectory list of an agent becomes empty afterwards, the agent himself is erased which leads to another reduction in the data.

3.5 Step 5: Trajectories clustering

The aim of this step is to select a significantly reduced sample of plans which are, as much as possible, representative of the whole trajectories in the SET2 files.

Thus, we try to group trajectories in clusters, each cluster being composed of plans "closed" to each others, according to a given distance measuring the similarity between two plans. Then, in each cluster, only one trajectory, representing all the others, will be chosen for the simulation. For instance, if two agents live in the same area and have the same sequence of activities in a similar cluster, we expect the two trajectories to be grouped in the same trajectory cluster and we only consider a single trajectory in the simulation. At the opposite, two different sequences of activities should be placed in different clusters and analyzed separately.

Following the observation made on files SET2, i.e. the agent detections are made every 10 mn, we associate to the trajectory of an agent j, a vector $t^j = (v^j, w^j)$ of dimension 288, where v^j and w^j are vectors of dimension 144 ($i.e$ 24 h / 10 mn).

Each component of v^j and w^j represents a detection instant, in a day period. For each i, v_i^j is the cluster where the agent is located in the instant $i = 1, 2, ..., 144$, eventually "*unknown*" if no detection have been made. And w_i is the type of activity made by the agent at the instant i, eventually "*unknown*". v and w can be computed from the trajectory lists.

For two vectors t^j and t^k of two agents j and k, we define the distance between them as follows :

$$d(t^j, t^k) = \sum_{i=1}^{144} \chi(v_i^j, v_i^j) + \sum_{i=1}^{144} \chi(w_i^j, w_i^j)$$

where, in general, $\chi(a, b) = 1$ if $a = b$ and 0 otherwise. In other words, this distance gives the sum of the cluster differences, and activity differences. It can be proved to be a metric in the mathematical sense. Using this metric, the same hierarchical clustering algorithm used for antennas clustering are performed for trajectories within a given threshold for the maximal distance between two plans.

After this step, we judiciously have to choose one trajectory in each cluster that will represent each class. We choose in each cluster the trajectory minimizing the total distance to the other trajectories of the same cluster. That is the so-called 1-median problem optimal solution (see Daskin [8]) computed in each cluster. Notice that in the hierarchical clustering algorithm, fixing a high threshold will have the effect to obtain large clusters, thus in turn to reduce significantly the number of plans to simulate, since only one plan is chosen by cluster. But when the threshold increases, the plans chosen become less representative of the whole set including those erased. In this case, numerical observations provide some relevant indications on the suitable value.

At the end of the step 3, we have a list of agent trajectories (one for each agent) supposed to be "representative" of the population. This list is transformed in a MATSim xml plan file and used for simulation with the previously generated network and facility files. At the end of the simulation, the optimization of amenities (re)locations starts (steps 6, 7, 8 of the figure 1). Below, we detail how this process works.

4 O-D flows, Amenities Selection and Local Search Algorithm

At the end a the MATSim simulation, each agent performed a plan in the transportation network, thus generating some flows between the amenities. We aggregate these flows to have a global view of the traffic. More precisely, between all couples of amenities, we compute the total number of trips during the simulation time. This gives an Origin-Destination flows matrix (F) between the amenities. We also compute the distances (in kilometers and in time) between each couple of amenities giving us two matrices (D and S). For each amenity, we store also the sum of the incoming and outgoing flows, giving us a view of the **traffic intensity** in each amenity.

Defining the global accessibility (see the introduction section 1) as the sum of the travel time, or travel distance, between couple of amenities and for all agents, the data generated above allow us to evaluate this global accessibility as follows:

$$\sum_{i=1}^{n}\sum_{j=1}^{n} F_{ij} D_{ij}$$

for the travel distance, or

$$\sum_{i=1}^{n}\sum_{j=1}^{n} F_{ij} S_{ij}$$

for the travel time, where n is the number of amenities.

The next step consists in selecting a set of amenities to relocate them in the best way. Two mechanisms are possible. Either we give (by hand) a list of amenities to study, or the code automatically computes one as follows.

The automatic amenities selection starts by sorting the amenities in the decreasing order of their traffic intensities. Then $x\%$ (for a given x) of the amenities with the highest traffic intensity, and of a certain given types of activity, are chosen to be candidates for relocations or spatial switching. The goal here is to search how the locations of the selected amenities should be exchanged in order to reduce the global accessibility cost. The exchange of the position of two amenities can be mathematically formalized by a permutation π defined in the set of amenities. More precisely, $\pi(i) = j$ means that the position of the amenity i is exchanged with the position of the amenity j. We thus search for a permutation π involving only the selected amenities and minimizing one of the following value:

$$(V_1) \ : \ \sum_{i=1}^{n}\sum_{j=1}^{n} F_{ij} D_{\pi(i)\pi(j)}$$

or

$$(V_2) \ : \ \sum_{i=1}^{n}\sum_{j=1}^{n} F_{ij} S_{\pi(i)\pi(j)}.$$

We add a constraint in the optimal permutation, that consists in accepting only the exchange of amenities of different types, i.e. exchanging two amenities of the same type do not impact at all the global accessibility.

Let us notice that in the current state of the code, moving an amenity in an non-occupied place, (as done for "Baux maraichers": see the introduction) is not possible but will be included in the next version. Such an option can be viewed as an extension of this work. Indeed, non-occupied place can be represented by a set of possible available locations in which “fictive” activities can take place, whith 0 incoming and outgoing flows.

The problem consists in minimizing (V_1) (or (V_2)) is a well-known problem in the combinatorial optimization litterature. It is called the Quadratic Assignement Problem [11]. We solve it using a standard local search procedure, also know as 2-opt neighborhood search (see [4] or [14] for a survey).

5 Preliminary Numerical Results

The aim of this section is to see how the methods we proposed behave in a complete round of SPOT simulations. We would like to know if the locations optimization may contribute to improve the overall utility or to decrease the global travel time.

It is important to state that these tests are preliminary. MATsim simulations have included a large set of scenario configurations, impacting the way that the utility function is computed, the way that each agent choose new locations for activity in the replanning process, the replanning strategy, the computation of the score of the plan which determines the utility values. For each of them, we have done some arbitrary choices that should be fit more correctly, considering real-life observations. We have also made many choices in the different steps of the plan construction. Some of them being open to criticism, considering the way the agents are detected by mobile phone antennas. Indeed, agents being detected when a call occur, the concept of “stop” does not rigorously correspond to a real-life stop, since we don’t know what the agent really does between two calls. Moreover, some activities (work for instance) may occur during travelling times. Using another detection technology, more accurate than a time granularity of 10 mn, may make more realistic the “stop” concept. Because of all of this drawbacks, the results reported here should be seen as an illustration of the “potential” of our method to contribute to urban planning process and planning. Further research will be necessary to make it more “operational”.

We launched two sets of simulations with an increasing number of agents in each, to assess the scalability. We ran the simulations using a DELL R510 server equipped with 125GB of memory and an Intel® Xeon® 64-bit processor with two cores of 2.67GHz each.

The first set has been performed by generating agent plans from the trajectory file $SET2\_P01.CSV$, dealing with the first two weeks of January 2013. Using the methodology explained above, we generate 4693 agent plans and simulate the plans with MATSim with a scale factor of 100. “Scale factor” is a MATSim parameter by which each agent will represent, in our case, 100 others.

We fixed to 100 the number of MATSim iterations, and to 5 the number of iterations of the whole SPOT loops. At the end of the 5 iterations, some statistics have been performed to analyze the variation of utility value and travel time following each relocation. The pictures 3 and 4 show the mean utility values, and mean travel times for all agents, after each relocation iteration.

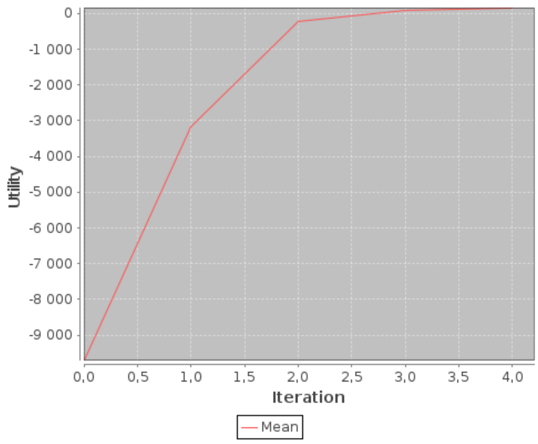

Fig. 3. Utilities Mean through simulation iterations $SET2\_P01.CSV$

The statistics are computed using a toolbox coded with the SPOT software. The utility means are obtained by computing the sum of the selected plan scores of all the agents at the end of MATSim iterations, divided by the number of agents. One can see that the utilities at the first iteration of the SPOT method are very low (even negative), showing that many agents perform long trips to realize their activities. Let us notice that the utility value is roughly the difference between the utility to perform the activities minus the disutility of the trip to reach these activities. Hence, longer the trip to perform few activities, lower the score. But whereas we propose some relocations, the average utility increases until it becomes positive. In the same time, the mean travel times decrease until a certain point where it increases. Notice that the utility function is more complex than the rough explaination above. We should intuitively expect that while utility increases, travel time decreases. However, some agents may realize more important activities, explaining this counter-intuive variation in the last iterations.

Instead of tracing the mean travel times, it is also possible to plot the maximum travel times. In this case, the maximum travel times experienced by all agents are extracted after the MATSim iterations. We obtain the picture shown

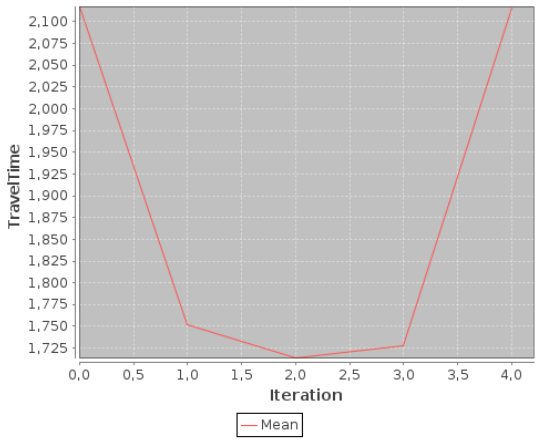

Fig. 4. Travel Times Mean $SET2\_P01.CSV$

in fig 5. We also observe a non-monotone variation, however we can see that maximum travel times experienced by the travelers tend to decrease with the relocations proposed.

To visualize the agent vehicle moves, the events of the last MATSim simulation, following the last relocation, have been displayed using the Senozon software [12]. This movie is available in the dropbox link given in footnote [13]. It can be observed, in this movie, that the plans generated initially with the SPOT methodology, and simulated in MATSim, are able to capture a simple fact observable in the Dakar region. The habitant trips from the popular east districts to the west, centre, and south areas where the majority of working, commercial and administrative activities are concentrated. And the trips in the opposite direction where probably the agents go back to home.

The second set of numerical tests concerned the trajectory file $SET2\_P05.CSV$ dealing with the first two week of March 2013. We generate in this case 6356 agent plans with the same scale factor of 100. Due to limited computational times, we ran 4 iterations of the complete SPOT method (instead of 5 as before).

The same variation as in the previous experiments can be observed in figure 6, showing that (at least for these two cases) the relocation contributes in increasing the utility mean. Moreover, at the opposed of the previous graph, the variation of the travel times mean (figure 7) here is monotonic, decreasing

[12] http://http://www.senozon.com/
[13] https://www.dropbox.com/s/syxgkn9w7w69q12/SPOT-SENOZON.mov?dl=0

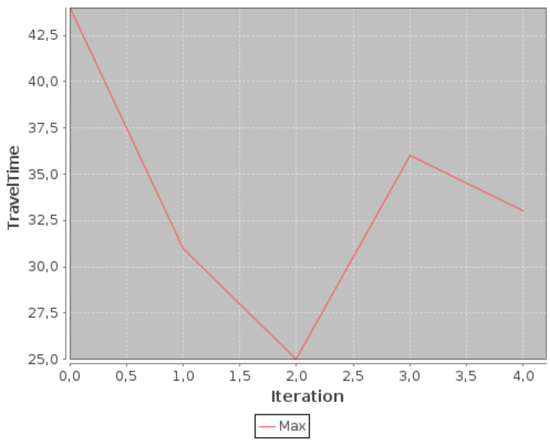

Fig. 5. Travel Times Max $SET2\_P01.CSV$

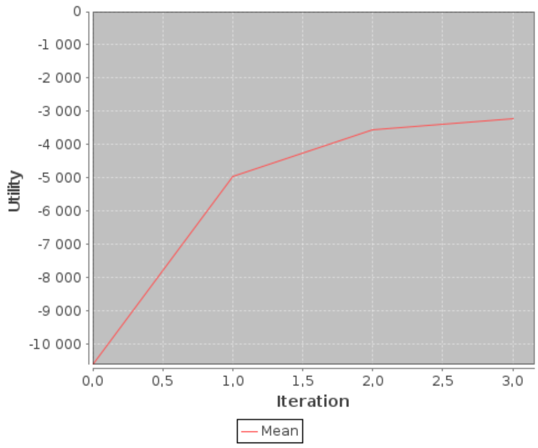

Fig. 6. Utilities Mean for $SET2\_P05.CSV$

iteration by iteration.

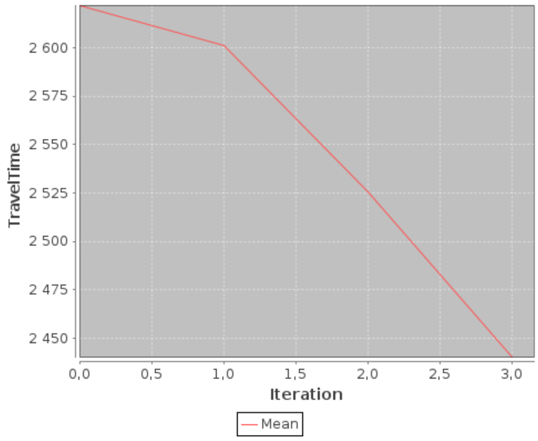

Fig. 7. Travel Times Mean $SET2\_P05.CSV$

Thus these experiments give promising indications on the ability of the method to proposed representative plans, and on the capacity of the relocation algorithm to improve, globally, people moves. However, as point out in the beginning of this section, further investigations are evidently needed to validate the approach.

6 Conclusion and Perspectives

We presented in this paper a set of techniques used for generating agent plans from mobile phone data, and for automatically proposing suitable relocations of some amenities within a large set of spatial opportunities, by simulating urban trips. This work is based on a research developed in agent based systems and operation research by an inter-disciplinary team composed of computer scientists and geographers. It opens on hard scientific and operational issues including representative spatial resampling, suitable activity assignment in time geography, agent utility modelling and optimisation. It also opens, after validating the methodology with further improvements, some perspectives on technological developments.

Acknowledgement

This work has been possible with the suggestions, advices, code contributions, implications in other projects of : Rosa Figueiredo, Mohammed A. Ait Ouahmed, Mouhamadou A.M.T. Baldé, Diaraf Seck. We gratefully thanks all. We also thanks Sonatel and Orange Group for providing us with the data.

References

1. M. Balàc, F. Ciari, C. Genre-Grandpierre, F. Voituret, S. Gueye, and P. Michelon. Decoupling accessibility and automobile mobility in urban areas. In *Transport Research Arena, Paris*, 2014.
2. M. Balmer, M. Rieser, K. Meister, N. Lefebvre D. Charypar, and K. Nagel. Matsim-t: Architecture and simulation times. *Multi-Agent Systems for Traffic and Transportation Engineering*, pages 57–78, 2009.
3. V.D. Blondel, N. De Cordes, A. Decuyper, P. Deville, J. Raguenez, and Z. Smoreda. Mobile phone data for development : Analysis of the mobile phone datas for the development of ivory coast. Technical report, 2013.
4. R-E. Burkard, E. Cela, P-M. Pardalos, and L. Pitsoulis. *The quadratic assignment problem*, pages 241–238. Kluwer Academic Publishers, Dordrecht, 1998.
5. CETUD. Enquête sur la mobilité, le transport et les services urbains à dakar (emtsu) - 2000. Technical report, 2000.
6. MMTTA CETUD, GMAT Ltée. Projet de plan de déplacements urbains pour l'agglomération de dakar, (ppdud) horizon 2025. Technical report, 2007.
7. D. Charypar and K. Nagel. Generating complete all-day activity plans with genetic algorithms. *Transportation*, 32(4):369–397, 2005.
8. M. S. Daskin. *Network and Discrete Location: Models, Algorithms and Applications*. John Wiley and Sons, Inc., New York, 1995.
9. J.H. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor, Michigan; re-issued by MIT Press, 1975.
10. H. Horni, D.M. Scott, M. Balmer, and K.W. Axhausen. Location choice for shopping and leisure activities implemented in the activity-based multi-agent transport simulation matsim. In *8th Swiss Transport Research Conference*, 2008.
11. T-C. Koopmans and M-J. Beckmann. Assignment problems and the location of economic activities. *Econometrica*, 25:53–76, 1957.
12. B.M. Ndiaye, M.A.M.T. Baldé, and S. Gueye. Rapport technique - activitÃ© 1 : Optimisation des réseaux de transport : Analyse des flots et localisation dâactivité (ortrans). Technical report, 2012.
13. M. Steinbach P-N. Tan and V. Kumar. *Introduction to Data Mining*. 2005.
14. P-M. Pardalos, F. Rendl, and H. Wolkowicz. The quadratic assignment problem: A survey and recent developments. In *In Proceedings of the DIMACS Workshop on Quadratic Assignment Problems, volume 16 of DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 1–42. American Mathematical Society, 1994.