Journal of Mathematics Research; Vol. 8, No. 3; June 2016
ISSN 1916-9795 E-ISSN 1916-9809
Published by Canadian Center of Science and Education

A New Heuristic Method for Transportation Network and Land Use Problem

Mouhamadou A.M.T. Baldé[1] & Babacar M. Ndiaye[1]

[1] Laboratory of Mathematics of Decision and Numerical Analysis, LMDAN-FASEG, University of Cheikh Anta Diop, BP 45087 Dakar-Fann, 10700 Dakar, Senegal

Correspondence: Babacar M. Ndiaye. E-mail: babacarm.ndiaye@ucad.edu.sn

Received: April 19, 2016 Accepted: May 3, 2016 Online Published: May 23, 2016

doi:10.5539/jmr.v8n3p94 URL: http://dx.doi.org/10.5539/jmr.v8n3p94

Abstract

Our paper deals with the Transportation Network and Land Use (TNLU) problem. It consists in finding, simultaneously, the best location of urban area activities, as well as of the road network design that may minimize the moving cost in the network, and the network costs. We propose a new mixed integer programming formulation of the problem, and a new heuristic method for the resolution of TNLU. Then, we give a methodology to find locations or relocations of some Dakar region amenities (home, shop, work and 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. Numerical results which show the effectiveness of the method, and simulations based on the scenario of Dakar city are given.

Keywords: transportation network, land use plan, quadratic assignment, heuristic, simulations.

1. Introduction

The Transportation Network and Land Use (TNLU) problem is a combination of transportation network optimization problem and land use planning or quadratic assignment problem. This model appear, for example, if we decided to find locations or relocations of some Dakar region amenities (home, shop, work, leisure places), that may reduce travel time or travel distance. By solving this problem we want to have the best layout of the city in the direction of the location of activities and roads linking these activities. A better configuration in the direction of the location of activities and transportation for a city would reduce congestion and pollution. We consider an activity (i) placed on a zone (k) and an important flow (cars) from other activities, in the direction of the activity (i). Suppose that there is traffic lights at intersections, thus flows will be subject to long waiting times before reaching activity (i), hence congestion. During this waiting, passengers of vehicles are subject to breathing harmful gases from cars. It would therefore make sense to ask whether the location of this activity is the best possible (reallocation of this activity or not). Since the 60s many authors are interested in these two problems separately. Indeed we can cite authors such as Scott (1969), Boyce, Farhi and Weischedel (1973), Hoang Hai Hoc (1973) and Billheimer (1970) for network design problems, Lawler(1963), Schlager (1965) and Gordon-McReynolds(1972), Maniezzo(1997), Xia and Yuan(2006), Huizhen Zhang(2010) and many others for the activities's location problems. Among the researchers who are interested in the combined problem we can cite Marc Los and more recently Lin and Feng (2003). They have considered a more complex model in the sense that their formulation allows the location of several activities in the same area. Marc Los gave an exact solving method and heuristic methods. The Los's model assumes the OD flow streams, once the activities have been fixed via the shortest paths in the sense of a fixed cost independent of the flow on the arcs (see Gueye, 2012).

In this paper, the heuristics that we propose are based on Shortest Path (SP), Location of Activity (LA) and Network Design (ND) problems. It gives results corresponding to the best affectation found, by solving a Quadratic Assignment Problem (QAP) which uses shortest path solutions for distance, and the network design (ND) problem associated to these affectations. We use heuristic methods (QAP-heur and ND-heur) for the resolution of QAP and ND problems.

After locating the n activities into the n locations, we develop a new methodology by which a very large set of relocation possibilities can be simulated, analyzed, and the "best" one can be found. The methodology was coded in a prototype software called Simulation-Driven Optimization (SDO), that originated from two projects DAMA (Balac et al., 2014). In this task, SDO uses the multi-agent simulator MATSim (see Balmer et al., 2009). 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. Following a complete MATSim simulation, in SDO we adopt a global (or collective) view which contrasts with the individual behaviour of the agents in the simulation. 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.

In the following Section 2, we give a mixed integer programming formulation of the problem. Then, in Section 3 we present one heuristic method, TNLU-heur, to solve it. The Section 4 is devoted to the computation of the lower bound for TNLU. We detail in Section 5 the simulation results of our algorithms. The Section 6 deals with SDO that calculates the optimal locations of activities based on the results of the simulations of MATSim. Finally, we conclude and give some perspectives in Section 7.

2. Formulation of the Model

We assume to have n locations and n activities to locate in these locations. Consider the graph $G(N, Ar)$ such as N is the set of node consisting different areas (n) and Ar is the set of links joining the n locations in pairs. Let:

$A = (a_{ik})$: location cost matrix of the activity i on the area k.

$Q = (q_{ij})$: moving flow matrix per unit of times between the activities i and j.

$C = (c_{ij})$: construction cost matrix of link or road (i, j).

$D = (d_{ij})$: direct distance matrix between two locations i and j or distance of link (i, j).

To determine the locations of activities, we consider the decision variables, called assignment, $x_{ij} = 1$ if the activity i is affected on the location j and 0 otherwise. To determine the configuration of the network, we need to introduce binary variables $y_{ij} = 1$ if the link (i, j) is selected and 0 otherwise. Depending on the constructed network, the shortest path within the meaning of the fixed costs of building d_{ij} associated with each arc (i, j) may vary. Then it is necessary to introduce variables of shortest path $l_{kl}(y)$ between k and l and variables z_a^{kl} which take value 1 if the shortest path between k and l goes by the link $a = (i, j)$ and take 0 otherwise. It is also estimated that if the activity i is placed on the node k and activity j on the node l, then the cost of transportation of a vehicle from k to l is equal to the distance $l_{kl}(y)$. On the other hand, the locations of activities induce fixed costs a_{ik} on the back of construction costs of roads c_{ij} for the network construction (for more details see Gueye, 2012).

The sum of the location, transportation and construction costs is expressed mathematically by the following function:

$$\sum_{i,k} a_{ik}x_{ik} + \sum_{i,k,j,l} q_{ij}l_{kl}x_{ik}x_{jl} + \sum_{i,j} c_{ij}y_{ij}$$

This is a cubic integer model. The "Optimal" network and activity locations are obtained by minimizing the above function under a set of constraints described below.

$$\begin{array}{lll}
Min & \sum\limits_{i,k=1}^{n} a_{ik}x_{ik} + \sum\limits_{\substack{i,k,j,l=1 \\ i\neq j, k\neq l}}^{n} q_{ij}l_{kl}x_{ik}x_{jl} + \sum\limits_{\substack{i,j=1 \\ i\neq j}}^{n} c_{ij}y_{ij} & \\
s-t: & & \\
 & (1)\ \sum\limits_{i=1}^{n} x_{ik} = 1\ , & k = 1, \dots, n \\
 & (2)\ \sum\limits_{k=1}^{n} x_{ik} = 1\ , & i = 1, \dots, n \\
 & (3)\ \sum\limits_{a\in\delta^+(k)} z_a^{kl} = \sum\limits_{a\in\delta^-(l)} z_a^{kl} = 1\ ,\ k \neq l, & k, l = 1, \dots, n \\
 & (4)\ \sum\limits_{a\in\delta^+(m)} z_a^{kl} = \sum\limits_{a\in\delta^-(m)} z_a^{kl},\ k \neq l, & k, l = 1, \dots, n \\
 & (5) l_{kl} - \sum\limits_{a\in Ar} d_a z_a^{kl} \geq 0\ ,\ k \neq l, & k, l = 1, \dots, n \\
 & (6)\ z_{a\in\delta^+(i)}^{kl} + z_{a\in\delta^-(i)}^{kl} \leq y_{a\in\delta^+(i)}\ ,\ k \neq l, & k, l = 1, \dots, n \\
 & (7)\ x_{ik},\ y_{ij} \in \{0, 1\}\ ,\quad i \neq j, & i, j, k = 1, \dots, n \\
 & (8)\ z_{ij}^{kl} \in \{0, 1\},\ l_{kl} \geq 0,\ k \neq l,\ i \neq j, & k, l, i, j = 1, \dots, n
\end{array}$$

where m is a intermediate node different to k and l taken respectively as origin and destination of the considered path, $\delta^+(i) = \{(i, j) \in Ar/j \in N\}$, $\delta^-(i) = \{(j, i) \in Ar/j \in N\}$.

The constraints (1) and (2) represent assignment constraints ensuring that only one activity is localized in one area only. For the shortest paths, are used the flow variables z_a^{kl} where are associated the constraints of flow conservation. Constraint (5) can give to variable l_{kl} value of the shortest path from k to l. The constraint (6) ensures that only are taken into account in the calculation of shortest paths, the link actually constructed.

Thus, in our model the variables are mainly x_{ik}, y_{ij} and l_{kl}, involved in the objective function, and z_a^{kl} which occurs only in the constraints.

This model is an extension of Koopmans-Beckmann's problem. In addition, our problem, which generalized quadratic assignment problem[1] is NP-complete (Garey et al., 1979; Sahni et al., 1976). Indeed, if we fix all the variables y_{ij} then the problem becomes of finding activities assignments minimizing the sum of travel costs. This is equivalent to solve a quadratic assignment problem. In sequel, we can reformulate the model by replacing the arc a by (i, j).

$$Min \quad \sum_{i,k=1}^{n} a_{ik}x_{ik} + \sum_{\substack{i,k,j,l=1\\ i\neq j,k\neq l}}^{n} q_{ij}l_{kl}x_{ik}x_{jl} + \sum_{\substack{i,j=1\\ i\neq j}}^{n} c_{ij}y_{ij}$$

$$s-t:$$

$$D = \left\{ \begin{array}{lll}
(1) \sum_{i=1}^{n} x_{ik} = 1 \ , & & k = 1,\dots,n \\
(2) \sum_{k=1}^{n} x_{ik} = 1 \ , & & i = 1,\dots,n \\
(3) \sum_{\substack{j=1\\ j\neq k}}^{n} z_{kj}^{kl} = \sum_{\substack{i=1\\ i\neq l}}^{n} z_{il}^{kl} = 1 \ , & k \neq l, & k,l = 1,\dots,n \\
(4) \sum_{\substack{j=1\\ m\neq j}}^{n} z_{mj}^{kl} = \sum_{\substack{i=1\\ i\neq m}}^{n} z_{im}^{kl} \ , & k\neq l, k\neq m, l\neq m, & m,k,l = 1,\dots,n \\
(5) \sum_{\substack{i,j=1\\ i\neq j}}^{n} d_{ij}z_{ij}^{kl} \leq l_{kl} \ , & k \neq l, & k,l = 1,\dots,n \\
(6) \quad z_{ij}^{kl} + z_{ji}^{kl} \leq y_{ij} \ , & i\neq j, k\neq l \ , & i,j,k,l = 1,\dots,n \\
(7) \quad x_{ik},\ y_{ij} \in \{0,1\} \ , & i \neq j, & i,j,k = 1,\dots,n \\
\quad z_{ij}^{kl} \in \{0,1\}, l_{kl} \geq 0,\ k \neq l,\ i \neq j, & & k,l,i,j = 1,\dots,n
\end{array} \right.$$

3. A Decomposition Method

We propose in this section to solve three sub problems for TNLU, which are:

- Shortest path problems;
- A location of activity problem;
- A network design problem.

It starts with Shortest Path problems (SP) which solutions are the shortest path between the different locations and denoted γ_{kl}. Then the solutions of shortest path problems are introduced into the Location of Activity problem (LA), which returns as solution (x_{ik}^*) the allocation of different activities in different locations. Finally, x_{ik}^* are used in the Network Design problem (ND), which returns as solution the shortest path and roads built, respectively denoted l_{kl}^* and y_{ij}^*. So (x_{ik}^*), (l_{kl}^*) and (y_{ij}^*) are solution of the decomposition method of TNLU. The TNLU-heur (Figure 1) is a decomposition method of TNLU which uses heuristic methods (QAP-heur and ND-heur in section 3.2 and 3.3) for QAP and ND problem.

[1] known to be NP-complete

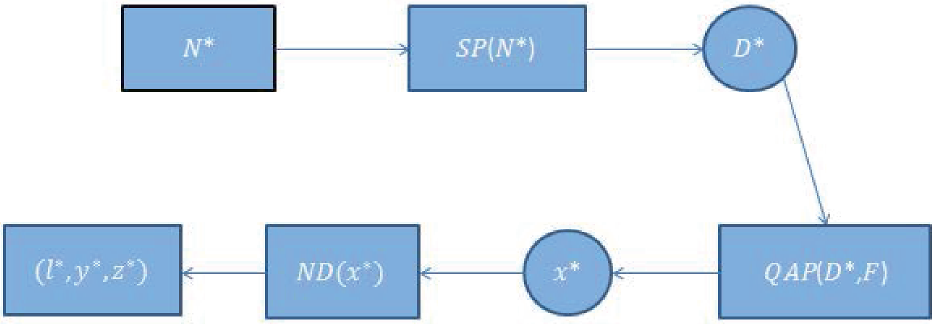

Figure 1. Diagram of TNLU-heur

3.1 Shortest Path Problems (SP)

This section contains the resolution of the SP problems which is formulated as follows:

$$Min \quad \sum_{\substack{i,j=1,i\neq j\\ i\neq l,j\neq k}}^{n} d_{ij} z_{ij}^{kl}$$

$$s-t:$$

$$Z = \left\{ \begin{array}{lll} (1)\ \sum\limits_{\substack{j=1\\ j\neq k}}^{n} z_{kj}^{kl} = \sum\limits_{\substack{i=1\\ i\neq l}}^{n} z_{il}^{kl} = 1,\ \ k\neq l, & & k,l = 1,\ldots,n \\ (2)\ \sum\limits_{\substack{j=1\\ j\neq m,j\neq k}}^{n} z_{mj}^{kl} = \sum\limits_{\substack{i=1,\\ i\neq m,i\neq l}}^{n} z_{im}^{kl} \leq 1\ ,\ k\neq m, l\neq m & & m,k,l = 1,\ldots,n \\ (3)\ \ z_{ij}^{kl} + z_{ji}^{kl} \leq 1\ ,\ \ i\neq j,\ i\neq l,\ \ j\neq k, & & i,j = 1,\ldots,n \\ (4)\ \ z_{ij}^{kl} \in \{0,1\}\ ,\ \ \ \ i\neq j,\ i\neq l,\ \ j\neq k & & i,j,k,l = 1,\ldots,n \end{array} \right.$$

$\forall\ k,l = 1,\ldots,n,\ k\neq l.$

Thus we solve $n^2 - n$ SP problems using IBM CPLEX. Finally, at the end of the iterations (of these SP problems), we obtain $\gamma_{kl} = \min\limits_{z\in Z} \sum\limits_{\substack{i,j=1,i\neq j\\ i\neq l,j\neq k}}^{n} d_{ij} z_{ij}^{kl}$ $\forall k,l = 1,\ldots,n,\ k\neq l.$

3.2 Activity Location Problem (AL)

For the LA problem we use solutions of previous problems(γ_{kl}), and we formulate the problem as following :

$$Min \quad g(x,\gamma) = \sum_{i,k=1}^{n} a_{ik}x_{ik} + \sum_{\substack{i,k,j,l=1\\ i\neq j,k\neq l}}^{n} q_{ij}\gamma_{kl}x_{ik}x_{jl}$$

$$s-t:$$

$$X = \left\{ \begin{array}{l} (1)\ \sum\limits_{i=1}^{n} x_{ik} = 1\ ,\ \ k = 1,\ldots,n \\ (2)\ \sum\limits_{k=1}^{n} x_{ik} = 1\ ,\ \ i = 1,\ldots,n \\ (3)\ x_{ik} \in \{0,1\},\ i,k = 1,\ldots,n \end{array} \right.$$

If we solve the Quadratic Assignment Problem (QAP), we find the solutions x_{ik}^*. Due to the fact that QAP is NP-hard, we propose a heuristic to solve it.

• A heuristic for QAP (QAP-heur)

We are given n initial assignments in a random choice. The value of the objective function is evaluated, then two assignments are chosen according to a specific criterion. Then the value of the objective function is evaluated in the case of swapping these two assignments. If there is a reduction of the cost then this permutation is preserved if not the previous location are kept. The process is repeated until a given number of iterations.

Thus for an initial allocation, we choose according to a criterion the best neighbour of the 2-opt permutation.

The selection criterion is the following (reducing the risk that the algorithm does not stop on a local solution during search):

- $i_0, k_0, j_0, l_0 \in \left\{ i, k, j, l \in [1, \dots, n] : x_{ik} = 1 = x_{jl} \right\}$ which maximizes the function :

$$a_{i_0k_0} + a_{j_0l_0} + \sum_{\substack{i,k=1 \\ i\neq j_0, k\neq l_0}}^{n} q_{ij_0} d_{kl_0} x_{ik} x_{j_0l_0} + \sum_{\substack{j,l=1 \\ j\neq i_0, l\neq k_0}}^{n} q_{i_0j} d_{k_0l} x_{i_0k_0} x_{jl}$$

- Otherwise, if it maximizes the value:

$$a_{i_0k_0} + a_{j_0l_0} + q_{i_0j_0} d_{k_0l_0} + q_{j_0i_0} d_{l_0k_0}$$

- Otherwise, if it maximizes :

$$a_{i_0k_0} + a_{j_0l_0}$$

We put a counter that gives the number of iterations since the last best solution, and if it exceeds a given threshold, then we consider that the algorithm is stuck on a local solution. The threshold is empirically and randomly chosen in the interval $[n^2/2, 2n^2]$.

The algorithm is a 2-opt neighbourhood search method associated with local search as tabou-search and stochastic search.

```
Data: $n$ number of locations and activities, $m$ number of iterations
Random initialization of $x$;
list ⟵ $\varnothing$;
i ⟵0;
while $i < m$ do
    i ⟵i+1;
    Best-Solution ⟵ $x$;
    Choose $i, k, j, l \in [1, \dots, n]$ using the criterion;
    $y$ ⟵ $permute(x)$;
    if $evaluate(x) \leq evaluate(y)$ then
        UPDATE list;
        Choose $i, k, j, l \in [1, \dots, n]$ using the criterion;
        $y$ ⟵ $permute(x)$;
    else
        $x$ ⟵ $permute(x)$;
        UPDATE list;
    end
end
```

Algorithm 1: QAP-heur

3.3 Network Design Problem (ND)

We solve the following problem :

$$Min \quad \sum_{i,k=1}^{n} a_{ik} x_{ik}^{*} + \sum_{\substack{i,k,j,l=1 \\ i\neq j, k\neq l}}^{n} q_{ij} l_{kl} x_{ik}^{*} x_{jl}^{*} + \sum_{\substack{i,j=1 \\ i\neq j}}^{n} c_{ij} y_{ij}$$

$$s-t :$$

$$R = \begin{cases} (1) \sum\limits_{\substack{j=1\\ j\neq k}}^{n} z_{kj}^{kl} = \sum\limits_{\substack{i=1\\ i\neq l}}^{n} z_{il}^{kl} = 1 \ , \ \ k \neq l, & k,l = 1,\ldots,n \\ (2) \sum\limits_{\substack{j=1\\ m\neq j}}^{n} z_{mj}^{kl} = \sum\limits_{\substack{i=1\\ i\neq m}}^{n} z_{im}^{kl} \ , \ \ k \neq l, k \neq m, l \neq m, & m,k,l = 1,\ldots,n \\ (3) \sum\limits_{\substack{i,j=1\\ i\neq j}}^{n} d_{ij} z_{ij}^{kl} \leq l_{kl} \ , \ \ \ k \neq l, & k,l = 1,\ldots,n \\ (4) \ z_{ij}^{kl} + z_{ji}^{kl} \leq y_{ij} \ , \ \ \ i \neq j, \ k \neq l, & i,j,k,l = 1,\ldots,n \\ (5) \ y_{ij} \in \{0,1\} \ , \ \ \ i \neq j, & i,j = 1,\ldots,n \\ \ \ \ z_{ij}^{kl} \in \{0,1\}, l_{kl} \geq 0, \ k \neq l, \ i \neq j, & k,l,i,j = 1,\ldots,n \end{cases}$$

The resolution of ND give us the solutions l_{kl}^{*}, y_{ij}^{*} and z_{ij}^{kl*}. We can simplify the objective function by $\sum\limits_{\substack{i,k,j,l=1\\ i\neq j, k\neq l}}^{n} q_{ij} l_{kl} x_{ik}^{*} x_{jl}^{*} + \sum\limits_{\substack{i,j=1\\ i\neq j}}^{n} c_{ij} y_{ij}$ without changing the problem as the sum $\sum\limits_{i,k=1}^{n} a_{ik} x_{ik}^{*}$ is constant.

• A heuristic for Network Design (ND-heur)

In this sub-section we propose a heuristic for the Network Design problem (ND). This heuristic is known in the literature as deletion method (see Gamvros). The ND-heur consists of:

1. The method start with all edges in accordance with the instance solved. This mean that $\forall i \neq j \in \{1,\ldots,n\}$, if y_{ij}^{*} indicates the constracted roads then :

$$y_{ij}^{*} = 1 \ \text{ if } \ (d_{ij} < \infty) \ \text{ and } \ (y_{ij}^{*} = 0 \ \text{ if } d_{ij} = \infty)$$

2. After that, choose $\bar{y}$ in the neighbourhood of y^{*} such that :

$$\text{For } (k_0, l_0) \in \{1,\ldots,n\}^2, k_0 \neq l_0 \ \ c_{k_0 l_0} = \max_{k,l} c_{kl} y_{kl}$$
$$\text{Set } \bar{y}_{k_0 l_0} = 0 \text{ and } \forall (k,l) \neq (k_0,l_0) \ \bar{y}_{kl} = y_{kl}^{*}$$

3. Then, solve the following problem :

$$\begin{array}{ll} Min & \sum\limits_{\substack{k,l=1\\ k\neq l}}^{n} \Big(\sum\limits_{\substack{i,j=1\\ i\neq j}}^{n} q_{ij} x_{ik}^{*} x_{jl}^{*} \Big) l_{kl} \\ s-t: & \\ & (1) \sum\limits_{\substack{j=1\\ j\neq k}}^{n} z_{kj}^{kl} = \sum\limits_{\substack{i=1\\ i\neq l}}^{n} z_{il}^{kl} = 1 \ , \ \ k \neq l, \qquad k,l = 1,\ldots,n \\ & (2) \sum\limits_{\substack{j=1\\ m\neq j}}^{n} z_{mj}^{kl} = \sum\limits_{\substack{i=1\\ i\neq m}}^{n} z_{im}^{kl} \ , \ \ k \neq l, k \neq m, l \neq m, \qquad m,k,l = 1,\ldots,n \\ & (3) \sum\limits_{\substack{i,j=1\\ i\neq j}}^{n} d_{ij} z_{ij}^{kl} \leq l_{kl} \ , \ \ \ k \neq l, \qquad k,l = 1,\ldots,n \\ & (4) \ z_{ij}^{kl} + z_{ji}^{kl} \leq \bar{y}_{ij} \ , \ \ \ i \neq j, \ k \neq l, \qquad i,j,k,l = 1,\ldots,n \\ & (5) \ z_{ij}^{kl} \in \{0,1\}, l_{kl} \geq 0, \ k \neq l, \ i \neq j, \qquad k,l,i,j = 1,\ldots,n \end{array}$$

   where x^{*} is a fixed activities allocation, to compute the shortest path associated to $\bar{y}$.
   So this problem is the network design problem associated with the activities assignment x^{*}. We solve this problem by using the Flyod-Warshall algorithm (Floyd, 1962).

4. After resolution of the problem, if there is a feasible solution and a cost reduction, update $y^{*} = \bar{y}$.

5. Finally, repeat all the process from phase 2 up to $n^2 - n$ times.

The step 2 specifies that a link which has the most contribution in the cost $\sum\limits_{i,j=1,i\neq j}^{n} c_{ij}y_{ij}$ is a candidate for deletion and is effectively deleted in the step 3 $\Big($if there is a reduction of the global cost : $\sum\limits_{\substack{k,l=1\\k\neq l}}^{n} (\sum\limits_{\substack{i,j=1\\i\neq j}}^{n} q_{ij}x^*_{ik}x^*_{jl})l_{kl} + \sum\limits_{\substack{i,j=1\\i\neq j}}^{n} c_{ij}y_{ij}\Big)$.

4. Lower Bound Based on Distances

In this section, the proposed bound is based on a linearization of the cubic term of TNLU problem formulation (see section 2). We replace $\sum\limits_{k=1}^{n}\sum\limits_{l=1}^{n} l_{kl}x_{ik}x_{jl}$ by a new variable D_{ij} $\forall i, j = 1, ..., n,\ i \neq j$, with additional constraints to keep the problem equivalent. So D_{ij} have a structure of distance, which allow us to add triangular inequality to the constraints (equation (3)). We obtain the following linear problem:

$$\text{(TNLU-LIN)} \qquad Min \quad \sum_{i,k=1}^{n} a_{ik}x_{ik} + \sum_{i,j=1}^{n} q_{ij}D_{ij} + \sum_{\substack{i,j=1\\i\neq j}}^{n} c_{ij}y_{ij}$$

$$s-t:$$

$$D'_{\gamma} = \begin{cases}
(1)\ \ x_{ik} \in X & \\
(2)\ \ D_{ij} \geq \gamma_{kl}(x_{ik} + x_{jl} - 1)\ ,\ \ k \neq l,\ i \neq j, & k, l, i, j = 1, \ldots, n \\
(3)\ \ D_{ij} \leq D_{im} + D_{mj}\ ,\ \ i \neq j, i \neq m \neq j, & k, l, m = 1, \ldots, n \\
(4)\ \sum\limits_{\substack{j=1\\j\neq k}}^{n} z^{kl}_{kj} = \sum\limits_{\substack{i=1\\i\neq l}}^{n} z^{kl}_{il} = 1\ ,\ \ k \neq l, & k, l = 1, \ldots, n \\
(5)\ \sum\limits_{\substack{j=1\\j\neq m}}^{n} z^{kl}_{mj} = \sum\limits_{\substack{i=1\\i\neq m}}^{n} z^{kl}_{im}\ ,\ \ k \neq l, k \neq m, l \neq m, & m, k, l = 1, \ldots, n \\
(6)\ \sum\limits_{\substack{i,j=1\\i\neq j}}^{n} d_{ij}z^{kl}_{ij} \geq \gamma_{kl}\ ,\quad k \neq l, & k, l = 1, \ldots, n \\
(7)\ \ z^{kl}_{ij} + z^{kl}_{ji} \leq y_{ij}\ ,\quad i \neq j, k \neq l, & k, l, i, j = 1, \ldots, n \\
(8)\ \ \sum\limits_{j=1}^{n} D_{ij} \geq \sum\limits_{k,l=1}^{n} \gamma_{kl}x_{ik}\ ,\quad i \neq j, & i, j = 1, \ldots, n \\
(9)\ \ \sum\limits_{i=1}^{n} D_{ij} \geq \sum\limits_{k,l=1}^{n} \gamma_{kl}x_{jl}\ ,\quad i \neq j, & i, j = 1, \ldots, n \\
(10)\ \ D_{ij} \geq 0\ ,\quad i \neq j, & i, j = 1, \ldots, n \\
(11)\ \ z^{kl}_{ij},\ y_{ij} \in \{0, 1\}\ ,\quad i \neq j,\ i, j = 1, \ldots, n, \quad k \neq l,\ k, l = 1, \ldots, n &
\end{cases}$$

with X the domain of affectation variables presented in subsection 3.2, and $\gamma_{kl} = \min\limits_{z\in Z} \sum\limits_{\substack{i,j=1,i\neq j\\i\neq l,j\neq k}}^{n} d_{ij}z^{kl}_{ij}$.

The relaxation of TNLU-LIN is a lower bound of TNLU, and we solve the problem using IBM CPLEX. Depending on the structure of the instances, to have a tighter bound, we can replace the inequalities of the constraints (6), (8) and (9) of (D'_{γ}) by equalities.

5. Numerical Simulations on Test Problems

The behavior of TNLU-heur is evaluated using test problems on academics instances and Dakar city. For academics instances we use data from Los (1978) and QAPLIB (Burkard et al., 1996) benchmark. The QAPLIB problems are instances of Nugent, Vollman, and Ruml (denoted by Nug), Elshafei (Els), Wolkowicz (Had), Krarup and Pruznan (Kra). More precisely we construct the instances from flows and distances of "Nug", "Els", "Had" and "Kra", where there is no localization and construction costs. In addition, the costs of localization are null and construction costs are obtained by multiplying the distances by a mean cost 10 ie $c_{ij} = 10.0 \times d_{ij}\ \forall\ i, j = 1, \ldots, n$ for instances “Nug” and “ Had”, by 100 for instances “Els”, and by 1000 for instances “Kra”. The mean construction costs are fixed according to the order of size of the values of distances.

Recall the considered data:

n : number of locations and activities.

$A = (a_{ik})$: location cost matrix of the activity i on the area k.

$Q = (q_{ij})$: moving flow matrix per unit of times between the activities i and j.

$C = (c_{ij})$: construction cost matrix of link or road (i, j).

$D = (d_{ij})$: direct distance matrix between two locations i and j or distance of link (i, j).

The lower bound has been solved with IBM Cplex Optimizer 12.5 solver interfaced with C++. The tests have been performed on a HP ProBook 450 Intel (R) Core (TM) i5 3230M CPU @2.40GHz.

For simulations on academic instances and Dakar city, we have executed ten times and have kept the best solution. This is due to the stochastic nature of the algorithm.

5.1 Tests on Academic Instances

In all tables, Los x means data of Los for n = x, Nug x means data of Nug for n = x and so on. In order to evaluate the result obtained by our heuristic we calculate the gap as follows:

$$Gap = \frac{opt - LB}{opt}$$

where opt is the best value found (upper bound) and LB is the lower bound based on distances. The following table 1 summarizes results for different step sizes. We see that the heuristic TNLU-heur provides good upper bounds for the TNLU problem. Indeed, for some instances of the academic test the gap is quite small. For instances Had and Nug, the CPU time grows at a linear rate with n. However, for instances Los the CPU time remains nearly constant, firstly for n=8, 10,12; and secondly for n=14,15.

5.2 Tests on Dakar City

For real life simulation of Dakar we could get from the DGTC (Direction des Travaux Géographiques et Cartographiques), a zonal division of the region into zones corresponding to the 2000 Census Report information with projection to 2015. And from this Census Report we could have information on the flow of movement (q_{ij}) between areas. For data of cost construction of roads (c_{ij}), we obtained information about mean costs updated in January 2008, and our choice fell on the road type monolayer whose width is $7m$ and the cost per km (all taxes included) is 33 246 500 FCFA ($\simeq 21.7764575e+09$ euro). So if we denote by C_M the mean cost of road construction, then the cost of construction of the road is calculated by $c_{ij} = C_M * d_{ij}$.

To calculate the distance we used OSRM project (Open Source Routing Machine) which is a C++ implementation of a routing engine high performance for shortest paths in road networks. It combines sophisticated routing algorithms with the data of the road network and free from OpenStreetMap (OSM). OSRM is able to calculate and output a shortest path between any origin and destination in milliseconds.

For the cost of localization activities (a_{ik}), we did not find an institution that can provide us with this information. Therefore we decided to build ourselves these costs realistically. We start with a choice of activities to locate in zones. The tests were done for a maximum number of zones equal to $n = 20$. Table 2 illustrates the concerned activities and zones.

In addition, to give the cost of locating an activity on a zone, we evaluated the mean cost of land for this activity in this zone and the cost of construction of the building for the same activity. The sum of these two costs gives the cost of location. So we look for the mean cost of $150m^2$ and for a considered activity we estimate approximately a mean size of the area, then we can obtain a mean cost of land for the activity in this zone.

For example, the Plateau Sud price of $150m^2$ can be estimated to $11.0e + 07$ FCFA ($\simeq 7.205e + 10$ euro). Area can be estimated for a Ministry to $300m^2 = 2 \times 150m^2$, thus the cost of land for this ministry in this zone is $2 \times 11.0e + 07 = 2.2e + 08$ FCFA($\simeq 1.441e + 11$ euro). In addition, we can estimate the cost of construction of the building to $2.0e + 08$ FCFA($\simeq 1.310e + 11$ euro). So the cost of locating the Ministry in Plateau Sud is $2.2e + 08$ FCFA + $2.e + 08$ FCFA = $4.2e + 08$ FCFA ($\simeq 2.751e + 11$ euro).

The other data used for simulations on Dakar city are reported in the following tables 3, 4, 5, 6 . Each table contains a

Table 1. Results of TNLU-heur

Instances	Best value found	Time CPU in second	Number of iterations QAP/ND	Lower bound	Gap
Els19	1.75309e+007	23	10000/342	8.18476+006	53.31%
Had12	1932	14	10000/132	1869.5	3.23%
Had16	4120	19	10000/240	3986.3	3.25%
Had20	7422	28	10000/380	7246	2.37%
Kra32	2.93893e+006	104	10000/992	1.55e+006	47.26%
Los8	6.83477e+006	15	10000/56	5.60862e+006	17.94%
Los10	1.24041e+007	15	10000/90	1.10511e+007	10.91%
Los12	2.40381e+007	15	10000/132	2.13929e+007	11.00%
Los14	4.03875e+007	25	10000/182	3.55418e+007	12.00%
Los15	4.92603e+007	25	10000/210	4.26734e+007	13.37%
Los16	5.89492e+007	32	10000/240	5.11031e+007	13.31%
Nug12	904	17	10000/132	824.222	8.83%
Nug14	1398	31	10000/182	1304.5	6.69%
Nug15	1570	20	10000/210	1453.78	7.40%
Nug16	2094	20	10000/240	1883.99	10.03%
Nug17	2232	20	10000/272	2053.28	8.01%
Nug18	2486	23	10000/306	2294.9	7.69%
Nug20	3196	26	10000/380	2918.45	8.68%
Nug22	10297	31	10000/462	8634.81	16.14%
Nug25	14353	43	10000/600	12041.4	16.11%
Nug30	7192	73	10000/870	6656.27	7.45%

Description: For each instance, the Time CPU is the minimal execution time in the ten experiments.

Table 2. Activities and areas

n	Activities	Zones
1	Ministry	Plateau sud / PL.SUD
2	Supermarket	Médina / MED
3	University	Grand dakar-usine / GD.US
4	Medical technology clusters	Castor / CAST
5	Medical Poles	Fann Point E - Amitié / F.P.E.A
6	Embassy	Ngor-Yoff / NG-YO
7	Police	Parcelles assainies / P.A
8	Fire Brigade	Patte d'oie-mariste / PO-MAR
9	Town Hall	Thiaroye-yembeul / TH.YEUM
10	Public Garden	Guediewaye-ouest / GUED-OUEST
11	Stadium	Guediewaye-centre / GUED-CENT
12	Residence	Guediewaye-est / GUED-EST
13	Telecom	Thiaroye sur Mer / TH-MER
14	Real Estate Agency	Malika-Keur Massar / MAL.KM
15	Post Office	Mbao / MB
16	Country-Police	Rufisque Quartiers traditionnels / RUF-TRADI
17	Secondary education	Diamniadio / DIAM
18	Olympic swimming pool	Hann mariste / H.M
19	Primary school	Sangalkam / SANG
20	Bar restaurant	Zone industrielle / Z.I

17×17 matrix. The values of localization and construction costs are rounded to two decimal.

Table 3. Distance matrix in kilometer based on the road network

0	4.05	6.82	9.13	7.33	19.90	12.90	10.40	20.20	18.90	17.00	15.40	16.20	27.00	27.30	29.80	40.20
3.71	0	8.44	7.55	4.57	11.20	11.90	8.68	18.20	17.80	15.80	14.10	15.00	18.16	25.70	27.90	38.70
6.14	8.38	0	2.87	3.78	11.10	7.09	5.65	15.80	14.90	11.60	10.30	12.00	22.80	23.00	25.30	35.90
7.97	7.37	2.79	0	4.93	11.80	5.11	3.79	13.60	12.20	9.82	8.40	9.91	19.80	21.10	23.60	34.00
7.18	4.59	3.81	4.86	0	7.37	8.91	8.04	18.10	15.40	12.90	11.50	14.40	24.60	25.90	27.80	38.40
12.74	12.30	14.60	11.40	7.50	0	10.20	13.90	20.80	18.30	15.90	14.40	16.80	27.10	28.00	30.50	40.90
11.29	11.40	8.40	5.23	8.63	9.42	0	5.47	10.90	9.29	6.85	5.40	8.88	17.00	10.82	22.50	32.90
9.43	9.85	8.31	6.86	10.80	13.30	5.76	0	10.50	9.88	7.51	6.05	6.89	16.60	18.00	20.40	30.80
15.82	17.20	15.40	13.70	17.80	20.60	10.90	10.70	0	3.86	4.84	5.58	4.81	6.59	14.70	16.90	27.50
15.54	16.60	14.40	13.00	15.40	18.30	9.30	9.97	3.86	0	2.82	4.07	8.72	10.80	15.50	17.70	28.40
16.00	15.90	12.70	9.92	12.90	15.80	6.80	8.36	4.84	2.82	0	1.59	7.08	10.80	15.50	17.80	28.30
15.20	14.10	10.70	8.50	11.50	14.30	5.35	6.16	5.58	4.01	1.59	0	8.21	11.80	16.60	18.80	29.40
13.58	15.90	12.40	10.90	15.10	17.70	10.20	8.00	4.35	6.99	5.25	6.23	0	10.60	11.90	14.40	24.80
27.10	18.16	22.40	19.90	24.30	26.90	17.00	16.80	6.59	10.80	10.80	11.80	11.00	0	10.60	12.80	23.40
23.40	21.70	18.70	17.30	21.70	23.90	16.30	14.30	10.60	11.50	11.50	12.50	8.44	6.61	0	8.47	19.60
30	27.90	25.00	23.70	27.60	30.40	22.80	20.70	16.90	17.70	17.70	18.80	14.90	12.80	12.00	0	24.80
40.40	38.40	35.50	34.10	38.20	40.80	33.10	31.10	27.50	28.30	28.30	29.40	25.30	23.40	17.30	25.30	0

Table 4. Matrix of moving flow(Origin-Destination matrix)

0	13310	394	2266	7218	1579	9148	1583	3235	1139	5023	981	1410	400	1917	74	24
12428	0	759	1055	5197	1881	3565	425	762	123	1418	926	704	239	467	0	15
328	850	0	1433	3986	0	446	0	149	245	0	0	0	0	467	0	24
2737	1234	904	0	2625	1404	1937	465	0	167	186	0	744	330	0	0	0
7882	4712	4014	2836	0	1222	2626	545	677	24	837	346	110	0	749	0	0
2948	1467	105	1218	1210	0	2543	453	0	0	0	327	903	84	0	0	24
9635	3479	446	1397	3302	2621	0	2937	761	734	1928	291	185	18	0	0	0
1773	356	0	465	730	235	2820	0	0	249	59	0	153	0	12	0	0
3633	1346	149	11	492	283	832	174	0	559	7610	2564	3939	2550	2634	0	39
1405	123	245	167	24	0	902	119	694	0	4138	654	101	0	0	0	0
5245	1841	59	447	1416	0	1786	153	7536	4920	0	6292	268	91	167	128	68
1616	926	0	0	291	327	291	0	2582	945	7380	0	0	149	0	0	0
1973	397	0	310	165	903	185	306	3583	101	463	0	0	0	534	0	0
685	239	0	330	0	0	0	0	2316	0	91	0	0	0	0	0	49
1956	467	467	0	282	0	0	0	2499	0	302	0	534	12	0	467	39
69	0	0	0	0	0	0	0	0	0	138	0	0	0	560	0	0
39	15	24	0	0	24	0	0	39	0	53	49	0	49	278	0	0

Table 5. Matrix of localization cost ($e + 08$ FCFA($\simeq 655\ e + 08$ euro))

4.2	3.1	2.5	2.8	3.3	2.6	2.4	2.6	2.1	2.3	2.4	2.5	2.3	2.1	2.2	2.1	2.1
74	71.8	70.6	71.1	72.2	70.8	70.4	70.7	69.7	70.2	70.4	70.5	70.2	69.8	69.9	69.7	69.7
1190	975	850	906	1016.5	871.6	838	860	767.1	816	830	848	810	778	784	764	762
0.4	0.3	0.2	0.2	0.3	0.2	0.1	0.2	0.1	0.1	0.1	0.2	0.1	0.1	0.1	0.1	0.1
970	862.5	800	828	883.2	810.8	794	805	758.5	783	790	799	780	764	767	757	756
7.6	5.5	4.2	4.8	5.9	4.4	4.1	4.3	3.4	3.9	4	4.2	3.8	3.5	3.5	3.3	3.3
41.6	21.5	9.8	15.1	25.4	11.9	8.7	10.8	2.1	6.7	8	9.6	6.1	3.1	3.7	1.8	1.6
20.7	13.5	9.3	11.2	14.9	10.1	8.9	9.7	6.6	8.2	8.7	9.3	8	6.9	7.1	6.5	6.4
46	24.5	12	17.6	28.6	14.2	10.8	13	3.7	8.6	10	11.8	8	4.8	5.4	3.4	3.2
37.7	19.8	9.3	14	23.2	11.1	8.3	10.2	2.4	6.5	7.7	9.2	6	3.3	3.8	2.2	2
307.7	284.4	270.8	276.9	288.9	273.2	269.5	271.9	261.8	267.2	268.7	270.6	266.5	263	263.7	261.5	261.3
246.7	175	133.3	152	188.8	140.5	129.3	136.7	105.7	122	126.7	132.7	120	109.3	111.3	104.7	104
6	4.4	3.5	3.9	4.7	3.6	3.4	3.5	2.8	3.2	3.3	3.4	3.2	2.9	3	2.8	2.8
0.7	0.4	0.3	0.3	0.5	0.3	0.2	0.3	0.1	0.2	0.2	0.2	0.2	0.2	0.2	0.1	0.1
8.1	4.9	3	3.8	5.5	3.3	2.8	3.2	1.8	2.5	2.7	3	2.4	1.9	2	1.7	1.7
52.3	27.3	12.7	19.2	32.1	15.2	11.3	13.8	3	8.7	10.3	12.4	8	4.3	5	2.6	2.4
290	236.3	205	219	246.6	210.4	202	207.5	184.3	196.5	200	204.5	195	187	188.5	183.5	183

Table 6. Matrix of construction cost ($e + 08$ FCFA($\simeq 655\ e + 08$ euro))

0	1.35	2.27	3.04	2.44	6.62	4.29	3.46	6.72	6.28	5.65	5.12	5.39	8.98	9.08	9.91	13.37
1.23	0	2.81	2.51	1.52	3.72	3.96	2.89	6.05	5.92	5.25	4.69	4.99	6.04	8.54	9.28	12.87
2.04	2.79	0	0.95	1.26	3.69	2.36	1.88	5.25	4.95	3.86	3.42	3.99	7.58	7.65	8.41	11.94
2.65	2.45	0.93	0	1.64	3.92	1.70	1.26	4.52	4.06	3.26	2.79	3.29	6.58	7.02	7.85	11.30
2.39	1.53	1.27	1.62	0	2.45	2.96	2.67	6.02	5.12	4.29	3.82	4.79	8.18	8.61	9.24	12.77
4.24	4.09	4.85	3.79	2.49	0	3.39	4.62	6.92	6.08	5.29	4.79	5.59	9.01	9.31	10.14	13.60
3.75	3.79	2.79	1.74	2.87	3.13	0	1.82	3.62	3.09	2.28	1.80	2.95	5.65	3.60	7.48	10.94
3.14	3.27	2.76	2.28	3.59	4.42	1.91	0	3.49	3.28	2.50	2.01	2.29	5.52	5.98	6.78	10.24
5.26	5.72	5.12	4.55	5.92	6.85	3.62	3.56	0	1.28	1.61	1.86	1.60	2.19	4.89	5.62	9.14
5.17	5.52	4.79	4.32	5.12	6.08	3.09	3.31	1.28	0	0.94	1.35	2.90	3.59	5.15	5.88	9.44
5.32	5.29	4.22	3.30	4.29	5.25	2.26	2.78	1.61	0.94	0	0.53	2.35	3.59	5.15	5.92	9.41
5.05	4.69	3.56	2.83	3.82	4.75	1.78	2.05	1.86	1.33	0.53	0	2.73	3.92	5.52	6.25	9.77
4.51	5.29	4.12	3.62	5.02	5.88	3.39	2.66	1.45	2.32	1.75	2.07	0	3.52	3.96	4.79	8.25
9.01	6.04	7.45	6.62	8.08	8.94	5.65	5.59	2.19	3.59	3.59	3.92	3.66	0	3.52	4.26	7.78
7.78	7.21	6.22	5.75	7.21	7.95	5.42	4.75	3.52	3.82	3.82	4.16	2.81	2.20	0	2.82	6.52
9.97	9.28	8.31	7.88	9.18	10.11	7.58	6.88	5.62	5.88	5.88	6.25	4.95	4.26	3.99	0	8.25
13.43	12.77	11.80	11.34	12.70	13.56	11.00	10.34	9.14	9.41	9.41	9.77	8.41	7.78	5.75	8.41	0

The map in Figure 2 give zones we have considered for simulation for n=17. The results of simulations are given in the Table 7, where DKx means data on Dakar city intances with size n=x. With this data (information) the simulations has been performed. Figure 3 shows the logical connections between the different areas, given by the optimal solution for $n = 17$. We can see that Diamniadio must be connected to Rufisque-Traditionnel and itself joined to Medina and Plateau-Sud, through direct links or roads. That is, it would be interesting to join the suburbs to downtown of Dakar, through direct connections to make the road network more fluid. In order not to overload the road network, we can develop a network of maritime shuttle linking these suburb areas and the downtown. And this is in perfect harmony with the geometrical shape of the entire south-eastern part of the region of Dakar. We see that in Table 7 the good result for simulations, where the gap is quite small for all sizes of tested problem.

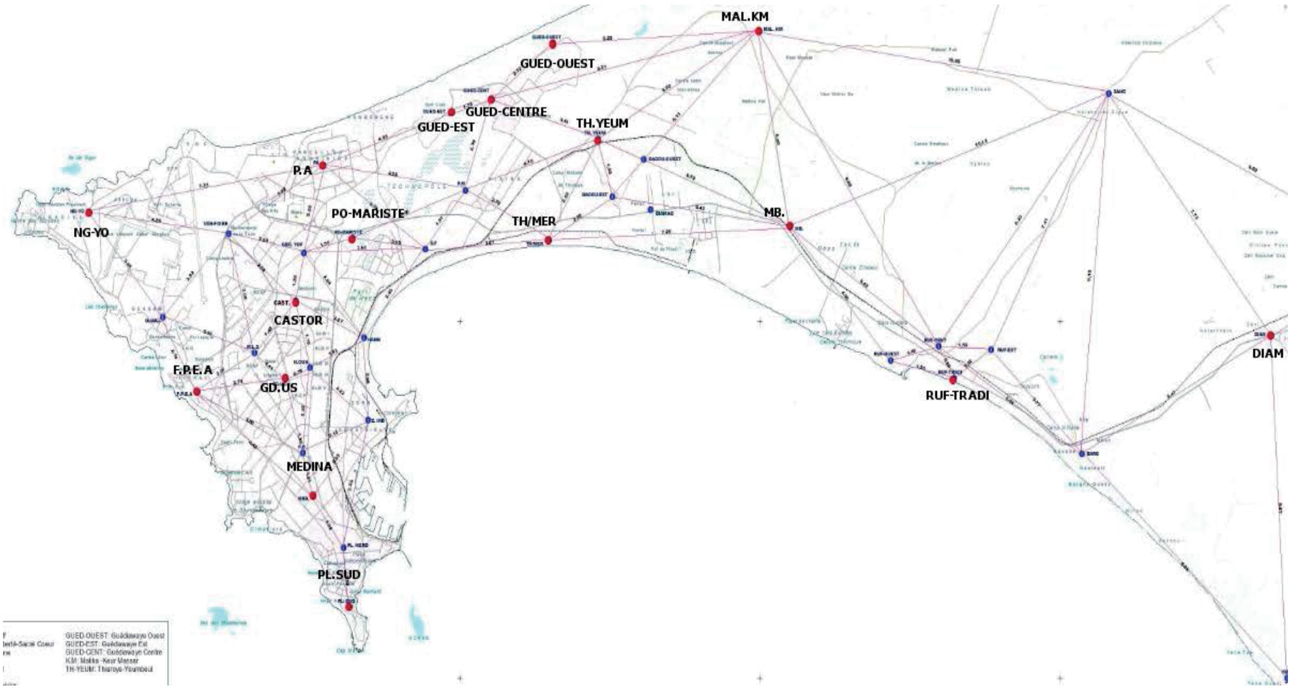

Figure 2. The zones for n=17.

Table 7. Results of TNLU-heur

Data size	Best value found	Time CPU(s)	Number of Iterations QAP/ND	Lower bound	Gap
DK6	1.75613e+011	17	10000/30	1.75339e+011	0.16%
DK10	1.69992e+011	18	10000/90	1.69259e+011	0.43%
DK12	2.09745e+011	19	10000/132	2.08995e+011	0.36%
DK17	2.26454e+011	25	10000/272	2.24796e+011	0.73%
DK20	2.43257e+011	30	10000/380	2.42865e+011	0.16%

In addition to the analysis of the SP, AL and ND problems, a new analysis of the set of relocation possibilities is proposed in the next section.

6. Multi-agent Simulations

In this section, the behavior of SDO, only limited to the test of **localisation**, is evaluated using test problems on Dakar city. We show how the simulator has been used to give the SDO software that can be seen as an over-layer of MATSim. We use MATSim as a multi-agent simulator system, and need for that to generate agent plans. The necessary adaptation elements of the simulations for Dakar city are presented in sub-section 6.3.

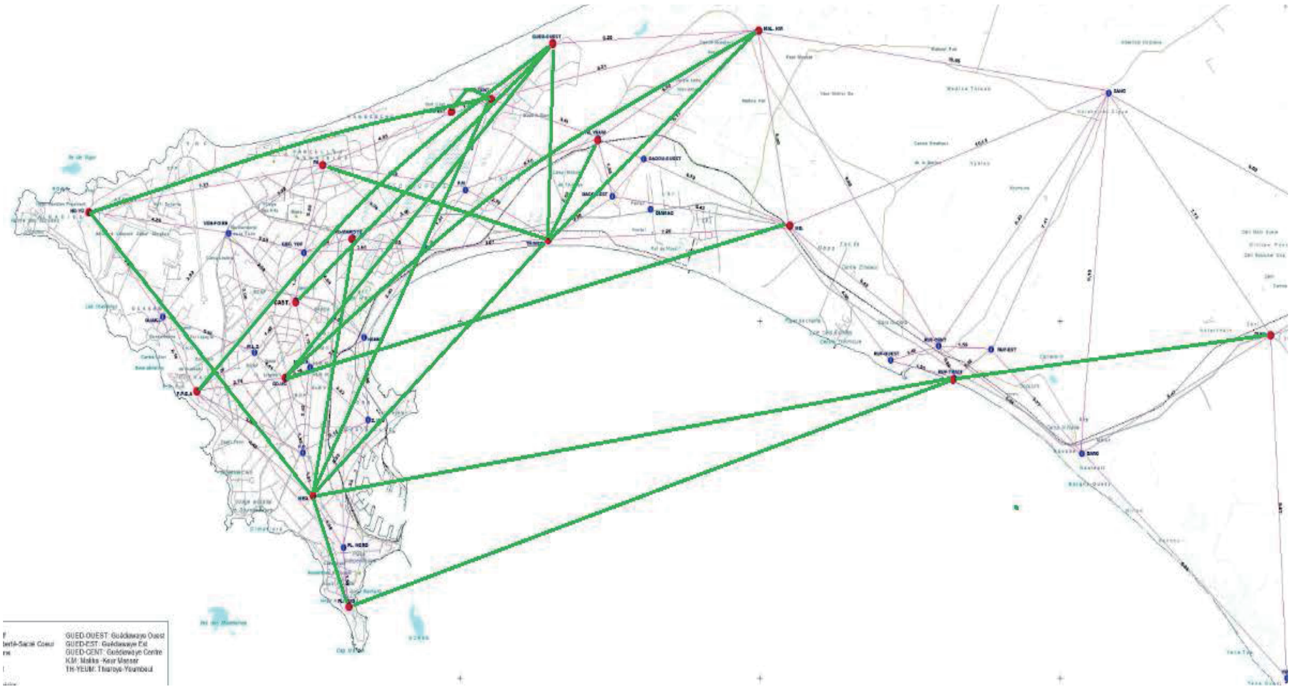
GUED-OUEST: Guédiawaye Ouest
GUED-EST: Guédiawaye Est
GUED-CENT: Guédiawaye Centre
K.M: Malika - Keur Massar
TH-YEUM: Thiaroye-Yeumbeul

Figure 3. Logical links between nodes

6.1 MATSim

A " multi-agent system (MAS)" is a system composed of a set of agents located in a certain environment and interacting according to certain relations. An agent is an entity characterized by the fact that it is, at least partially, autonomously. The implementation of this system on a computer by computer programs, translating these interactions is the multi-agent simulation. In our paper, the agents are individuals (flows of activities) of a city that interact during the displacement. MATSim is a simulation tool multi-agent especially in transport system simulation. The tool is designed since a decade thanks to the collaboration of volunteers and researchers. This is one of the biggest free systems in that category. With adequate computer resources, MATSim is able to simulate the behavior of millions of individuals on very large transportation networks. It is compatible with the data sets of various geographic information systems as OpenStreetMap. And being free, it is designed to be extensible and customizable.

Very schematically, the MATSim simulations are conducted in three steps (see Figure 4). Each agent (or individual) has at the beginning of the simulation a plan of activities that he wants to perform on the day. This can be for example: leave home at 8:00 am to bring his children to school and then go to work, take back their children at 4:30 pm to lead them in a sporting activity, retrieve them to 6:00 pm to do some shopping and then come back to home. Initially, the plan thus

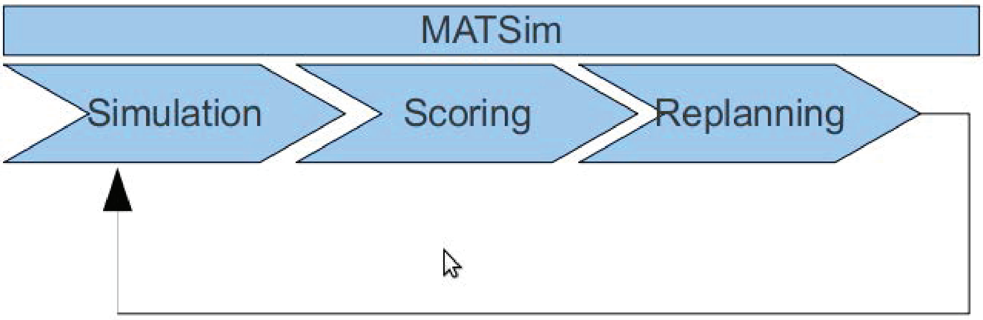

Figure 4. Diagram of MATSim simulations

specifies the sequence of tasks to be done by individuals but does not mention where exactly and how these activities will be made. The execution of the plan will specifically consists in making choices on these specific places and displacement routes to perform these activities.

After the first iteration of the simulation, a phase called ”scoring” is performed. It will be to evaluate, to put a note (or a score) on the selected locations and routes. This score is a function called **utility** in the mathematical sense (function) and economic (utility) of the term that the alleged agents thinking and all acting in the same way (which is restrictive,...) will seek to maximize.

In this objective of maximizing that a replanning phase follows that of ”scoring”. Replanning allows agents to change eventually their plans by deciding other execution venues of their activities, other routes, or of elimination of activities, helping to increase the utility of their displacements.

The system iterate through these three steps until a maximum number of iterations, or until observation of minimal changes of utility scores. Note, however, that from a mathematical point of view, there is no evidence, in this system, of convergence to a stable state. MATSim simulation requires data on the transport network, the plans of the agents and the location of all the activities they will perform.

6.2 Simulation-Driven Optimization (SDO)

The SDO is an overlay of MATSim as part of a similar project called DAMA (Balac et al., 2014). The tool calculates the optimal locations of activities based on the results of the simulations of MATSim. The general operation is described in the Figure 5 below. Generally, after a MATSim simulation, SDO takes into account the displacements generated by the simulation and propose a relocation of some activities (possibly all) to increase the utility scores. The problem that solves SDO is whether where should be located the activities so that the sum of utility scores of the agents is the highest possible. Due to the complexity of the function used by MATSim, we substitute in SDO, to that function the sum of displacement time or traveled distances. Intuitively, more displacement will be shorter in time and distance, and more one can hoped that the utility for the agent will be strong.

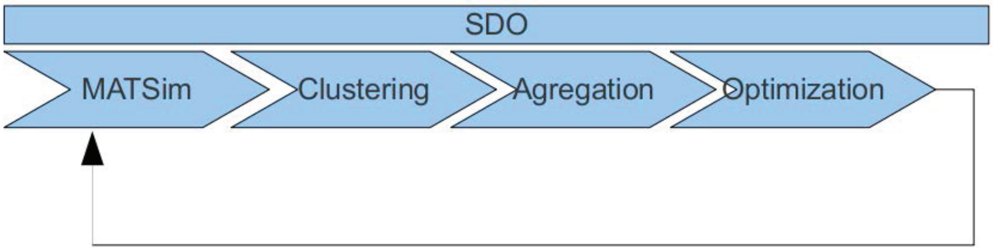

Figure 5. Simulation-Driven Optimization (SDO)

Thus we have the same problematic of combinatorial optimization designated by the acronym LU. For the time, SDO only limited to the aspect of localization. We do not treat the problem TNLU discussed in Section 2 but only a part (AL) thereof. The full integration of methods developed for TNLU will be the subject of further developments.

The large quantity of possible activities in a city makes necessary the execution of a previous phase called ”clustering” where activities will be grouped into groups (clusters) of activities. Each group will be considered as representative of a single entity. This has the effect of reducing the number of activities to be taken into account. The total amount of displacement between these groups, as well as the displacement times are calculated in the phase known as ”aggregation”. These data represent matrices of flow and distance of the problem AL. Due to them, the heuristics developed for solving the problem AL are executed, after which an ”optimal” relocation is proposed. This relocation changes the geographical coordinates of the proposed activities. Then, a new MATsim simulation is restarted, with these new localizations, and so on until a fixed number of iterations. Analogously to the MATSim simulation, there is no mathematical proof that the loop which concatenates simulation and optimization of localizations converges to a stable state of the displacements.

6.3 SDO computations on Dakar City

The use of SDO tool to the cases of Dakar requires to acquire the necessary files for MATSim simulations. Three types of files are generally used: the network file, the file containing the original plans of the agents, and one containing the list of activities and localizations.

Files describing transport networks were first obtained by the OpenStreetMap website (2014). The OpenStreetMap (OSM) is an international project founded in 2004 with the aim of creating a free map. Volunteer contributors send GPS data to the computer system which is responsible to integrate them in the geographic database. The geographical information about Dakar city are refined but remains incomplete. So we used partial information. These OSM files were transformed into the format used by MATSim as shown by the following network diagrams (Figure 6) obtained with the company's software Senozon (2014). The second type of file contains agents' plans. In order to make sure first that the proposed methods were operating in computational point of view, we considered for Dakar, a sample of 102 agents (from n=17 zones) and their selected randomly activities. The third type is the file indicating the selected activities and their localizations, extracted from files of OSM networks.

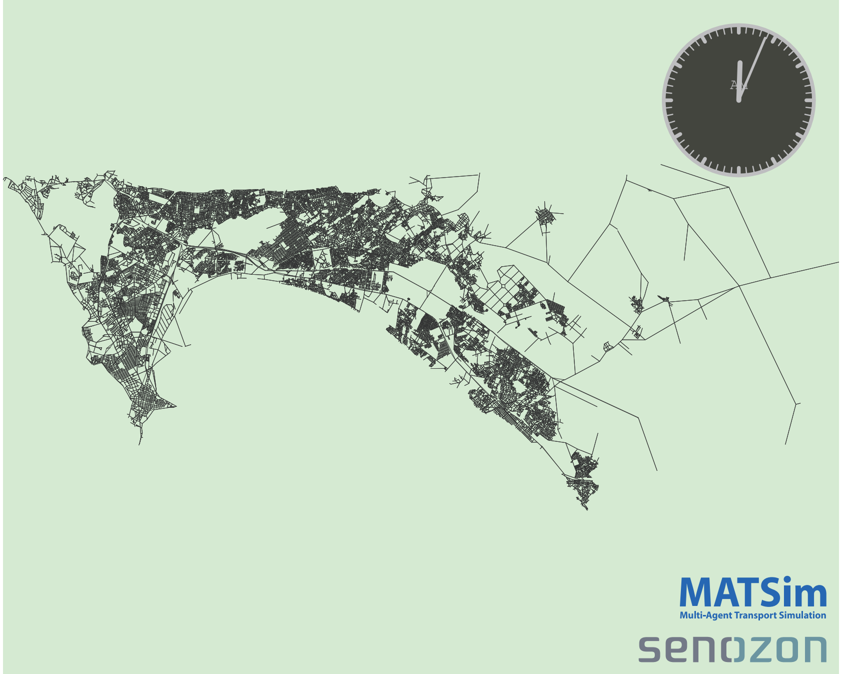

Figure 6. The Dakar network

With this information, the simulation and optimization loop explained above has been performed. Each iteration of the loop generates the outputs of the MATSim simulation which can be analyzed at the end of process. One can for example for each iteration perform the average of displacement times (or distances traveled, or scores) of all agents and see its evolution during the iterations. As in each loop an optimization phase is realized, the shape of the curve on a plane (horizontal axis: the iteration number, vertical axis: the average) allows to analyze the relevance of the approach. The decreasing displacement times indicate that the proposed relocations allow to facilitate access to activities. We give in Figures 7, 8 and 9 the evolution of averages obtained for the case of Dakar. The TravelTime is in seconde (s) and the TravelDistance in kilometer (km).

The graph on the utility (utility) of the Figure 7 shows that during the iterations, performed relocations made it possible to increase the average utility of users because the curve is increasing. In other words, people do displacements more and more considered as "useful" in the sense of the metric of the utility function of MATSim. Thus, the relocations increase the average "satisfaction" of users.

The curve of displacement times (TravelTime) in Figure 8 is generally decreasing, showing that times decrease with the proposed relocations.

On the other side, It is surprising that the distances (TravelDistance) traveled (Figure 9) increase slightly and decrease simultaneously. Everything goes as if the agents were doing, sometimes, a little more of distance, but in less time on average. This may be explained by a phenomenon called "congestion" that we have certainly confirmed by a more detailed analysis of the functioning of MATSim.

The congestion hypothesis consists in considering that the transport time between two locations not only depends on the distance between them, but also the number of people following the same displacement itinerary. If we place under this

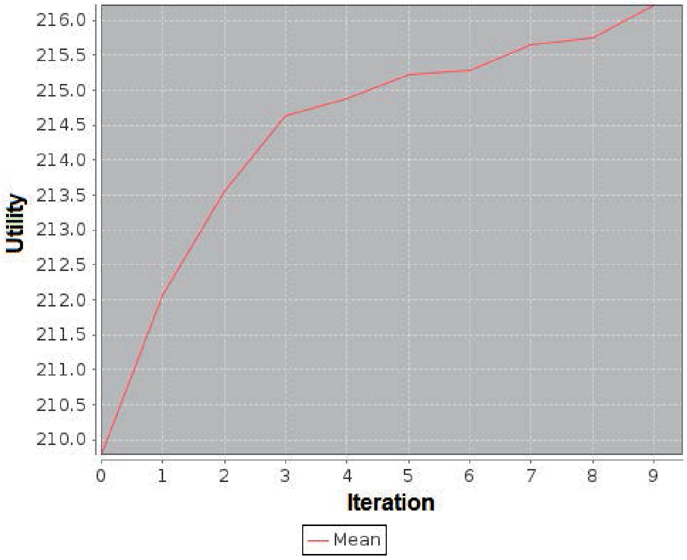

Figure 7. Average of the utilities

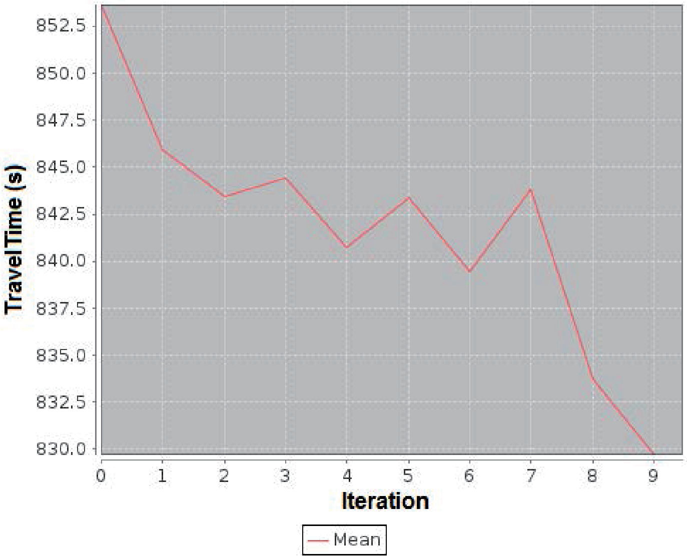

Figure 8. Average of displacement times

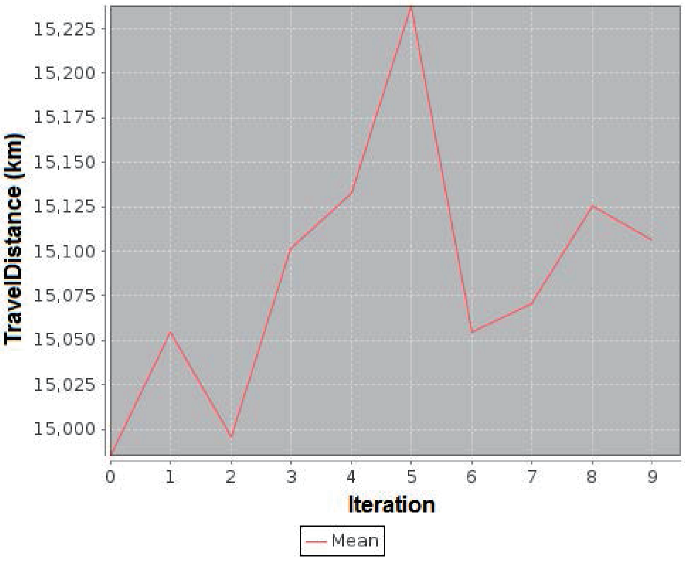

Figure 9. Average of the distances covered

hypothesis, a longer route, less frequented than another may still generate lower displacement times.
The variations (in reverse directions) of curves of distance and time appear to show that it is this phenomenon that has occurred for the case study in Dakar.

7. Conclusion

In this paper, we have proposed and analysed for the first time a new class of heuristic (TNLU-heur) for the TNLU problem. The new mixed integer formulation is a reformulated version of the Los's one in a more general sense. Unlike the model of Los we do not impose symmetry on the roads. Our new algorithm for TNLU was presented and applied to problems from the literature designed to highlight some intrinsic difficulties of TNLU, and from Dakar city. In addition to the convergence analysis, a new methodology of relocation possibilities is proposed. The SDO perfomance is evaluated by studying the quality of solution set in terms of optimal locations of activities. We investigated the average of displacement times, distances and scores; and established an interesting result characterizing the Utility, Travel Time and Travel Distance. In future work, we plan to study the combination between TNLU-heur and genetic or greedy algorithm. Another aspect that we wish to study will comprise the full integration of SP, AL and ND problems for further development.

Acknowledgement

The authors thank Serigne Gueye for his invaluable suggestions. They also thank the anonymous referees for useful comments and suggestions.

References

Balac, M., Ciari, F., Genre-Grandpierre, C., Voituret, F., Gueye, S., & Michelon P. (2014). *Decoupling accessibility and automobile mobility in urban areas*. In Transport Research Arena, Paris.

Balmer, M., Rieser, M., Meister, K., Lefebvre, N., Charypar, D., & Nagel, K. (2009). *Matsimt: Architecture and simulation times*. Multi-Agent Systems for Trac and Transportation Engineering, pages 57-78. http://dx.doi.org/10.4018/978-1-60566-226-8.ch003.

Billheimer, J.W. (1970). *Optimal route configurations with fixed link construction costs*, Stanford Research Institut. SRI-Project 454531-309.

Boyce, D.E., Farhi, A. & Weischedel, R. (1973). *Optimal network problem: A branch-and-bound algorithm*. Environment and Planning, 5,519.-533. http://dx.doi.org/10.1068/a050519.

Burkard R. E., Karisch S. E., & Rendl F. (1996, September). *QAPLIB- A Quadratic Assignment Problem Library*. European Journal of Operational Research.

Floyd, R. W. (1962, June). *Algorithm 97: Shortest Path*. Communications of the ACM 5 (6): 345.

http://dx.doi.org/10.1145/367766.368168.

Gamvros, I., Golden, B., Raghavan, S., & Stanojevié, D. *Heuristique search fo network design*. The Robert H. Smith School of Business. University of Maryland. College Park, MD 20742-1815.

Garey, M-R., & Johnson, D-S. (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co. New York, NY, USA.

Gordon, P., & McRcynolds, W.K. (1972). *Optimal urban forms : Towards a complete modelling of urban interdependencies*. in : Research Papers in Economics no.4 (University of Southern California, Los Angeles, CA).

Gueye, S. (2012 January 8). *Modèle affectation quadratique / Dimensionnement réseaux projet ortrans*.

Hoang Hai Hoc. (1973 Jan). *A computational approach to the selection of an optimal network*, Management Science 19. no.5. http://dx.doi.org/10.1287/mnsc.19.5.488.

IBM ILOG CPLEX Optimization Studio V12.5, Inc. Using the CPLEX^R Callable Library and CPLEX Barrier and Mixed Integer Solver Options (http://www-01.ibm.com/software/integration/optimization/cplex-optimization-studio).

Lawler, E.L. (1963). *The quadratic assignment problem*. Management Science, 9, 586-599. http://dx.doi.org/10.1287/mnsc.9.4.586.

Lin, J-J., & Feng, C-M. (2003). *A bi-level programming model for the land use-network design problem*, The Annals of Regional Science, 37:93-105. http://dx.doi.org/10.1007/s001680200112.

Los, M. (1978). *Simultaneous optimization of land use and transportation. A Synthesis of the Quadratic Assignment Problem and the Optimal Network Problem*, Regional Science and Urban Economics, 8, 21-42. http://dx.doi.org/10.1016/0166-0462(78)90010-8.

Maniezzo, V. (1997). *Exact and approximate nondeterministic tree-search procedures for the quadratic assignment problem*. Research Report CSR 97-1, Scienze dell'Informazione, Cesena site, University of Bologna.

MATSim. (2014). Matsim. URL www.matsim.org.

OpenStreetMap Development Team. (2014). Open Street Map. URL www.openstreetmap.org. The Free Wiki World Map.

Open Source Routing Machine. URL http://project-osrm.org

Sahni, S., & Gonzalez, T. (1976). *P-complete approximation problems*, Journal of the Association of Computing Machinery, 23, 555C565. http://dx.doi.org/10.1145/321958.321975.

Schlager, K.J. (1965). *A land use plan design model*, Journal of the American Institute of Planners. http://dx.doi.org/10.1080/01944366508978151.

Scott, A.J. (1969). *The optimal network problem: Some computational procedures*, Transportation Research, 3,201-210. http://dx.doi.org/10.1016/0041-1647(69)90152-X.

Senozon. (2014). Senozon. URL http://senozon.com/

Xia, Y. & Yuan, Y.X. (2006). *A new linearization method for quadratic assignment problem*. Optimization Methods and Software, 21(5):803C816. http://dx.doi.org/10.1080/10556780500273077.

Zhang, H., Beltran-Royo, C., & Ma, L. (2010). *Solving the quadratic assignment problem by means of general purpose mixed integer linear programming solvers*.

Copyrights