

# SimTaxi

Guide de l'utilisateur

3 mars 2003

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Pré-requis</b>	<b>2</b>
<b>3</b>	<b>Installation</b>	<b>2</b>
<b>4</b>	<b>Configuration</b>	<b>3</b>
<b>5</b>	<b>Utilisation</b>	<b>4</b>
5.1	les dumps . . . . .	4
<b>6</b>	<b>Interprétation de l'affichage</b>	<b>4</b>

## 1 Introduction

Dans cette 1ère partie de développement nous nous sommes principalement attardés sur la simulation ainsi que l’affichage de celle-ci. De ce fait, il n’y a pas vraiment d’interface utilisateur.

## 2 Pré-requis

L’exécution de SimTaxi requiert les éléments suivant :

- Python  $\geq$  2.2.1
- wxPython  $\geq$  2.3.3
- PyOpenGL  $\geq$  2

Dans le cas où l’une ces librairies ne serait pas présente sur votre système, startSimTaxi vous l’indiquera ainsi que le lien pour l’obtenir.

Tous les tests ainsi que les validations du logiciel ont été fait sur une Mandrake(9) Linux(2.4.19). Certains développeurs étant sous win32, nous avons pu constater que le logiciel y fonctionnait aussi.

Nous n’avons pas fait de tests sur des versions antérieures des librairies.

## 3 Installation

SimTaxi ne requiert pas d’installation particulière. Il suffit de posséder le dossier des sources que l’ont peut obtenir à l’aide de la commande<sup>1</sup> :

```
cvs -z3 -d:pserver:anonymous@cvs.sf.net:/cvsroot/simtaxi co dev
```

ou depuis le site *[http : //SimTaxi.sf.net](http://SimTaxi.sf.net)*

---

<sup>1</sup>requière un client CVS

## 4 Configuration

Actuellement, SimTaxi se configure à l'aide d'un fichier texte (*config.txt*). On y trouve les options suivantes (les options commentées ne sont pas encore prises en compte) :

```
1 # Fichier de préférences de SimTaxi (http://SimTaxi.sf.net)
2
3 #####
4 # Interface utilisateur
5 gui : True # utilisation ou non de l'affichage
6 # ATTENTION dans le cas de la non utilisation du gui, la simulation
7 # est difficile à stopper
8
9 # uniquement si gui
10 pseudoContinu : False # crée un intervalles de temps artificiel entre les év.
11
12 #uniquement si pseudoContinu
13 dureeSec : 0.05 # la durée d'un sec dans la simulation
14
15 #####
16 # Initialisation
17 nbTaxi : 180
18 vitesseTaxiKMH : 50
19
20 ndStation : 30
21 tailleStation : [20, 10, 10, 5] # 1x20 2x10 et les autres 5
22
23 #####
24 # Demandes
25 nbCoursesJour : 2700 # nb de clients pour 1 "journée"
26 hPremiereCourse : 0 # date du 1er client
27 hDerniereCourse : 3600*24 # durée en seconde d'une "journée" (d'une simulation)
28 moyenneCourseKM : 0 # 8, 0 = ne pas en tenir compte
29 #germeDemandes : 1
30
31 #####
32 # Graphe
33
34 # sep permet de séparer les dossiers
35 # curdir est le dossier courant
36 fichierGraphe : curdir+sep+'graphe'+sep+'graphe.gr'
37 #nbCarrefour : 80
38 #nbRue : 140
39 #diametreKM : 12 # 20 min en temps
40 #germeGraphe : 1
```

## 5 Utilisation

Le lancement de SimTaxi se fait à l’aide de *startSimTaxi.pyw* :

```
./startSimTaxi.pyw
```

L’extension *pyw* indique sur certains systèmes, que nous avons à faire un à GUI.

Un démarrage correct doit afficher les lignes suivantes :

```
# Id: SimTaxi.py,v 1.25 2003/03/02 19:53:36 erreur Exp #

Chargement des chemins déjà calculés...
800 arbres existants.
Initialisation stations...
Initialisation taxis...
Recup de la demande...
2700 clients
==== Début de la simulation
0.0 %
0.0150462962963 %
0.0162037037037 %
```

SimTaxi commence par charger<sup>2</sup> tous les arbres qui sont déjà calculés (fichier *dumpChemins*). Ensuite la mise en place des stations et des taxis est faite. Le tout est suivi du chargement de la demande (fichier texte *dumpClients*). Puis la simulation commence avec le % du déroulement total de celle-ci.

### 5.1 les dumps

Pour améliorer la performance de la simulation, les chemins ainsi que les clients sont sauvés dans des fichiers pour ne plus être calculés lors de l’exécution. Et ce, pour plusieurs raisons :

**dump des chemins** Si nous ne stockions pas les arbres retournés par l’algorithme de Dijkstra, ils seraient calculés un nombre important de fois. Leur stockage sous forme de fichier est raisonnable (10MB pour 800 arbres).

**dump des clients** Le respect de la moyenne de longueur des chemins lors de la génération de la demande étant un point lourd ce qui nous pose certains problèmes. Nous arrivons à la respecter mais de façon trop lourde pour pouvoir le faire en cours de simulation. Nous l’avons donc séparée de la simulation. La demande étant très légère à stocker (fichier texte contenant les dates, lieu de départ et lieu d’arrivée des clients).

## 6 Interprétation de l’affichage

Les carrés rouges sont les carrefours, reliés par les morceaux de routes. Les clients sont des carrés bleus clignotants. Les taxis sont les triangles jaunes accompagnés de leur numéro et d’un symbole. Quand ce symbole est une flèche bleue ils vont chercher un client, quand il est un rond rouge ils transportent un client et quand il est une flèche orange ils rentrent en station. Quand un taxi pose un client, celui-ci est toujours bleu mais disparaît progressivement. Les stations sont les polochons bruns sur le bord des routes. Elle sont accompagnées de leur taux de remplissage.

<sup>2</sup>Pour des raisons inconnues le chargement est très long ( 2min pour un 1,6GHz). Et tant que la simulation n’a pas tourné assez longtemps pour pouvoir calculer tous les arbres, l’arrêt de celle-ci est tout aussi long (enregistrement des nouveaux arbres).