Multi-day activity models
An extension of the Multi-Agent Transport Simulation (MATSim)

Author(s):
Ordóñez Medina, Sergio A.

Publication Date:
2016-08

Permanent Link:
https://doi.org/10.3929/ethz-b-000120636

Rights / License:
In Copyright - Non-Commercial Use Permitted
Multi-day activity models: an extension of the Multi-Agent Transport Simulation (MATSim).

Sergio Arturo Ordóñez Medina

Working paper
Future Cities Laboratory August 2016
Multi-day activity models: an extension of the Multi-Agent Transport Simulation (MATSim).

Sergio Arturo Ordóñez Medina
Future Cities Laboratory (FCL)
Singapore ETH Centre (SEC)
138602 Singapore
phone: +65-9298 7627
fax: +41-44-633 10 57
ordonez@ivt.baug.ethz.ch

August 2016

Abstract

MATSim is a large-scale multi-agent, activity-based transport simulation model. It can simulate the mobility of each person in an region, managing millions of agents in reasonable computation times. However, it is designed to simulate and reach a user equilibrium for a period of one day. This restricts the study of current transport planning challenges. Recent studies show that the behaviour variety of travellers can not be well analysed with only one day simulation results. Development of advanced time consumption travel models require observations of at least a week for calibration purposes, because a complete cycle of work and leisure must be included. To expand the standard MATSim time horizon, few changes in the current implementation have to be made. However, there are two reasons why the standard MATSim is not ideal for multi-day scenarios. First, the evolutionary algorithm of MATSim takes too long to reach a user equilibrium with longer periods, because of the combinatorial increase in the number possible activity chains to test per agent. The second reason why MATSim is not ideal for multi-day simulations is the difficulty of preparing multi-day plans to start the evolutionary process. In this paper a new approach for multi-day simulations is presented. To solve the mentioned MATSim drawbacks, a differentiation between fixed activities and flexible activities is proposed. An extension of MATSim to initialize agents with incomplete plans of fixed activities and schedule on-the-fly flexible activities is designed implemented and tested. Two case studies were prepared to evaluate the new approach. They show that the process is computationally feasible. The first iteration (where all agents plan flexible activities on-the-fly) takes 3.5 minutes for a 1% sample (37,425 agents) of the Singapore MATSim scenario. This is achieved running 20 threads for parallel flexible activities planning, and using 30GB of RAM memory. In the first case study, the new approach is compared with a standard MATSim evolutionary process. In 100 iterations, the average utility improves 100% more than the average utility increase with the current implementation. This happens because agents can schedule new flexible activities with
new conditions of travel times every iteration. However, the evolutionary process needs 66% more RAM memory, and iterations are 47% slower than the standard MATSim. In the second case study a multi-day scenario with an unexpected event is simulated. Agents react on-the-fly to the event by planning different activities at different places and/or different times, while a few others make planning mistakes traveling to the affected area. Comparative analyses between the affected day and the other days are performed.

Keywords
Agent-based simulation, Activity-based models, Fixed and flexible activities, Multi-day simulation

Preferred citation style
1 Introduction

MATSim ((Horni et al., 2016), (Balmer, 2007)) is a software to simulate transport demand and supply interactions for millions of agents, each one representing an individual person. Each agent has a transport demand represented by a chain of activities it has to perform during one day at different times in different places. The decisions on how to travel between places to perform these activities are made before the mobility simulation as a plan. Models to decide the route, start time, mode and/or destination of the journeys per person are included in MATSim and they can depend on their socio-demographic characteristics. However, as decisions can change when a person interact with others, MATSim executes an evolutionary algorithm to optimize utility of agents. The same day is executed hundreds of times, making changes in some agent planned journeys each time, remembering good decisions or forgetting bad decisions. The process reaches a user equilibrium when no agent can improve their score any further by making any of the allowed changes to its plans.

Simulations with one day time horizon are restricted when studying current transport planning challenges. By expanding the time horizon of MATSim, comparisons of different scenarios can be performed more accurately. The decisions are not only based on indicators of an average day, but also on weekends and other periods predominantly characterized by leisure activities. For the comprehensive study of travel behaviour, single day simulation results are insufficient. Usual analytic procedures, like clustering the population based on travel patterns, need multi-day information to account for intrapersonal variability (Schlich, 2004). Furthermore, longer time horizons allow to include restrictions like time and money budgets, and to simulate individual mode choice over time, identifying mode clienteles (Kühnimhof and Gringmuth, 2009). Development of advanced time consumption travel models require observations of at least a week for calibration purposes, because a complete cycle of work and leisure must be included (Jara-Díaz et al., 2008). Hence, multi-day simulations improve decision making support of transport models.

To expand the standard MATSim time horizon (one day lasts 30 hours), few changes in the current implementation have to be made. However, there are two reasons why the standard MATSim is not ideal for multi-day scenarios. First, the evolutionary algorithm of MATSim takes too long to reach a user equilibrium with longer periods, because of the combinatorial increase in the number possible activity chains to test per agent. As mentioned by Fei (2010) the fundamental problem of activity scheduling is its combinatorial complexity due to its number of dimensions (activity durations, locations, number of activities, activity types, activity order, etc.). The modeler needs to develop solutions which allow solving the problem in tractable time.
The second reason why MATSim is not ideal for multi-day simulations is the difficulty of preparing multi-day plans to start the evolutionary process. To generate a realistic weekly demand is still an open issue, because commonly used daily plans techniques are not efficient for longer periods of time ((Bayarma et al., 2007), (Arentze and Timmermans, 2009), (Kuhnimhof and Gringmuth, 2009), (Feil, 2010), (Nijland et al., 2012), (Arentze et al., 2013), (Märki et al., 2014)). Furthermore, collected data for periods of time longer than one day is seldom available ((Munizaga et al., 2011), (Stopher et al., 2008)), although several efforts have been made to obtain multi-day records recently ((Doherty et al., 2001), (Doherty and Miller, 2000), (Bhat et al., 2004), (Du and Aultman-Hall, 2007), (Bohte and Maat, 2009)).

In this paper a new approach for multi-day simulations is presented. To solve the mentioned MATSim drawbacks, a differentiation between fixed activities and flexible activities is proposed. An extension of MATSim to initialize agents with incomplete plans of fixed activities and schedule on-the-fly flexible activities is designed implemented and tested. First, results show that by allowing agents to schedule flexible activities, experienced utilities increase much faster than the standard MATSim (the average utility of all the agents increases 100% more in 100 iterations). Furthermore, with the MATSim extension, a multi-day scenario with an unexpected event is simulated. Agents react to the event by planning different activities at different places and/or different times, while a few others make planning mistakes traveling to the affected area. Comparative analyses between the affected day and the other days are performed.

The paper is organized as follows: first, the current state of MATSim for multi-day simulations is reviewed. Then, an overview of the proposed approach is presented, and more details of the implementation are included in the third section. In the results section, two case studies are described and simulation outputs are analyzed. Finally conclusions are presented and future work is proposed.

2 Standard MATSim multi-day simulation

MATSim was originally designed for daily mobility simulations. However, many components are capable of executing longer periods of time. In this section, details of how to run a trivial multi-day simulation using the standard MATSim are presented. A period of seven days is used for the descriptions in this section. The first part focuses on the transport demand, the second on the transport supply, and independent MATSim modules are reviewed in the end.
2.1 Multi-day transport demand in MATSim

Generating a realistic multi-day demand is still an open issue, because commonly used daily plans techniques are not efficient for longer periods of time ((Bayarma et al., 2007), (Arentze and Timmermans, 2009), (Kühnimhof and Gringmuth, 2009), (Feil, 2010), (Nijland et al., 2012), (Arentze et al., 2013), (Märki et al., 2014)). Furthermore, collected data for periods of time longer than one day is seldom available ((Munizaga et al., 2011), (Stopher et al., 2008)), although several efforts have been made to obtain multi-day records recently ((Doherty et al., 2001), (Doherty and Miller, 2000), (Bhat et al., 2004), (Du and Aultman-Hall, 2007), (Bohte and Maat, 2009)).

In MATSim, the demand is defined by detailed activity plans for every person in a synthetic population. For a weekly simulation every plan must be defined for 168 hours or more. MATSim is able to read any time in the hh:mm:ss format and convert it to seconds (e.g. the last second of the last day of the week corresponds to the time 167:59:59 or 604,799 seconds).

From the computation perspective, to handle a multi-day demand evidently requires more RAM memory than a daily demand. As mentioned in Erath et al. (2012), a 25% sample of the Singaporean synthetic population needs 12.2 GB of RAM memory to allocate one plan for more than 500,000 agents. As the size of daily plans of different days in one week is relatively uniform, more than 85 GB are needed to allocate weekly plans. Besides, in the standard MATSim process each agent retains several executed plans in memory and the default number of plans is five. Therefore, a functional 25% sample for a weekly simulation needs approximately 430 GB.

More computation time is also needed to simulate a weekly transport demand in MATSim. Evidently, the mobility simulation takes 7 times longer than a daily demand. For instance, the 25% Singapore scenario takes about 25 mins as described by Erath et al. (2012), therefore, a weekly mobility simulation takes almost 3 hours. Re-planning processes also take 7 times longer because weekly plans require 7 times more mutations. As a weekly simulation generates 7 times more events, computation time of Scoring and Analysis MATSim modules increase linearly with the number of simulated days. Some details of these modules are discussed below.
2.2 Multi-day transport supply in MATSim

2.2.1 Road Network

As the MATSim road network model is not dynamic, the road network used for daily simulations works just fine for multi-day simulations. MATSim network interfaces provide methods for dynamic properties (e.g. time-dependent free speed of a link), but the standard implementations are static. Additionally, a dynamic implementation of a road network can be designed to store dynamic properties for longer periods of time.

The travel times data structure, in which dynamic travel times are recorded every iteration for each link in the network, is hard-coded with a maximum time of 30 hours. By creating a new `TravelTimeCalculatorProvider`, the total time of the simulation can be included as the maximum time to store travel times.

2.2.2 Public Transport

The public transport model in MATSim is designed to run for longer periods of time. The transit schedule model defines times when public transport vehicles reach the stations as offsets from the departure time. Thus, to specify a weekly public transport schedule, it is necessary to specify two aspects: (i) Times when public transport vehicles depart within the interval 0:00:00 - 168:00:00, and (ii) the specific vehicles which are departing during one week. In reality, the same vehicle performs several departures during the week, or even within a day. If this information is available, MATSim simulates delayed departures, i.e. when a vehicle arrives late after finishing a public transport journey, its new departure will start later than the scheduled time.

2.3 MATSim modules in a multi-day simulation

2.3.1 QSim

The default MATSim mobility simulator is designed to run simulations of cars and public transport of any time length. In the `QSim config group`, it is necessary to define the end time of the simulation as 168:00:00.
2.3.2 Scoring

For scoring, the Scenario config group includes a new parameter called \textit{Simulation period in days} which allows to score plans with longer time lengths. This parameter is useful for simulation periods which are a multiple of one day, but it has two problems: (i) it is not related to the mobility simulation and (ii) if the simulation period is not a human cycle, the scoring is performed in a wrong manner. Because of the first problem, scores won’t be correct if the mobility simulation \textit{End time} parameter and the \textit{Simulation period in days} parameter are different. The second problem happens because the default activity scoring function in MATSim called \textit{CharyparNagelActivityScoring}, checks if the type of the first activity is the same than the type of the last activity and joins them as only one activity for scoring. If the simulation period is not a human cycle (e.g. the simulation starts on Tuesday and finishes on Saturday, or the simulation starts at 6 am and ends at 9 am), this cyclic assumption will score wrongly the plan. For these reasons, an extension of the \textit{CharyparNagelActivityScoring} is necessary for multi-day simulations. It scores plans with any duration in a non cyclic way, but needs the start time of every plan to be specified. It is also useful to design a scoring function which takes into account that activities of the same type performed consecutively with a frequency higher than expected (e.g. shopping groceries more than twice a week), should obtain lower scores. Ordóñez Medina (2015a) proposes an extended version of the standard MATSim utility function.

2.3.3 Re-planning

Re-planning strategies and modules are designed to mutate plans of any size within the MATSim evolutionary algorithm. However, weekly mutations are not implemented. For example, the \textit{Sub-tour Mode Choice} module selects one tour of the plan and changes the transportation mode of the trips performed in that tour. For a period of one day this mutation is significant because an agent performs one or two tours per day. But in a week an agent can perform 14 or more tours, and changing the mode of one of those is not a significant mutation. New re-planning modules and strategies can be designed for weekly mutations.

2.3.4 Analysis

The majority of MATSim standard analyzers do not depend on the simulation total time and no modifications are needed. The \textit{Stop Watch}, \textit{Travel Distance Stats}, \textit{Score stats}, and \textit{Trip durations} modules work just fine on weekly simulations. The only analysis module which depends on the
mobility simulation total time is the Leg Histogram. It has hard-coded the total time as 30 hours and the information is just calculated till this time. A new Leg Histogram Event handler has to be added to obtain departures and arrivals of agents by time of the week.

3 MATSim extension overview

This paper proposes an extension of MATSim to simulate longer periods of time. There are two reasons why the standard MATSim is not ideal for multi-day scenarios. First, the evolutionary algorithm of MATSim takes too long to reach a user equilibrium with longer periods, because of the combinatorial increase in the number possible activity chains to test per agent. Second, MATSim is not ideal for multi-day simulations because synthesize multi-day initial plans from commonly available data is difficult. Techniques to generate daily activity plans are not practical for longer durations. With the proposed extension, MATSim users will be able to prepare realistic multi-day activity-based demand for a synthetic population by means of commonly available data. This section introduces the main concepts of the new strategy.

3.1 Fixed and flexible activities

Human activities can be classified in two categories: fixed or mandatory and flexible or optional ((Arentze and Timmermans; 2000), (Miller, 2005), (Chen and Kwan; 2012), (Doherty et al.; 2002)). This differentiation is commonly used in transport science where activities or locations are classified as primary or secondary. In this paper the two categories will be called fixed and flexible to reinforce the fact that fixed activities are prearranged.

With this classification, activity scheduling can also be divided in two steps: (i) To arrange fixed activities, (ii) To schedule flexible activities within time windows left after arranging fixed activities. Scheduling fixed activities have been extensively studied by transportation modelers and urban planners ((Hansen; 1959), (Small; 1982), (Vovsha and Gupta; 2013), (Ordóñez Medina and Erath; 2013)). They have tried to find spatial patterns within geographical regions like cities, and temporal patterns during one day or longer periods of time. The challenge of the second step is the combinatorial variability of flexible activities. This makes it more difficult to find a small number of patterns that can explain flexible activity decision making. For example, when analysing the household interview travel survey of Singapore (HITS 2012) containing daily activity chains of 1% of the population of a city, there are more than 900 combinations of activities. By contrast, there are just 40 combinations of fixed activities after removing flexible
activities. What’s more, if fixed activities in this dataset are just categorized as Home and Work (i.e. Studying is a type of work), only 10 combinations can be found. Hence, it seems a good strategy to study fixed and flexible activities separately to solve this problem.

In MATSim, activities are not classified as fixed or flexible. To start the MATSim process, it is necessary to provide fully specified activity plans for the whole population. With the proposed extension, incomplete plans with fixed activities can be handled, and when free time windows are found during the simulation, flexible activities will be schedule on-the-fly.

### 3.2 Activity scheduling on-the-fly

In reality, people plan their activities at many different times. Some activities can be planned months or years in advance or just before the activity occurs. In MATSim, activity plans must be fully specified, and activity scheduling decisions taken on-the-fly and under specific circumstances cannot be modeled. Many flexible activity decisions are taken in this way, e.g. the decision of doing groceries after leaving early the work place. The proposed MATSim extension includes activity schedulers for flexible activities that are triggered while the mobility is running. When an agent finds free time in its plan, MATSim will schedule new activities, times, locations and trips to reach activity locations.

The proposed utility-maximization algorithm to schedule new activities, finds an approximately optimal activity-trip chain for a given time window according to activity performing utilities and traveling dis-utilities. It allows to include intrinsic variables from the decision maker and external information from the place and time where and when the decision is taken. It is recursive and it doesn’t prioritize any scheduling dimension. It returns the sequence of activities without imposing any size and the trips between them. For each activity the method calculates start time, duration, location and type; and for each trip the algorithm returns the travel time and the mode. It is possible to calibrate the number and duration of specific activities varying the corresponding utility parameters. Spatial, temporal and tour restrictions can also be imposed. A full description of this algorithm can be found in Ordóñez Medina (2015a).

### 4 Passive planning

In this section, details of the implementation of the introduced MATSim extension are presented. The name of this new module is Passive planning, referring to the fact that agents are not
active in the mobility simulation while they are scheduling new flexible activities. With this approach, a different activity scheduling optimization problem is solved on the fly at each free time window of each agent. This reduces the size of the problem which doesn’t depend anymore on the total simulation time, but on the biggest free time window. As mode choice, destination choice, route choice and time planning happen when solving each scheduling problem, it’s not necessary to include re-planning modules, at least for flexible activities. As the processes to schedule activities are computationally expensive, a parallelization framework is also proposed and implemented.

4.1 Overview

The first step of the extension is to represent incomplete plans in MATSim. Plan elements in MATSim can be activities and legs. Activity times can be defined by a duration or by an end time, while leg times are defined by a travel time. An incomplete plan has at least one period of free time without a defined plan element. These free time windows have to be in between two activities of the plan, because no leg can be defined without a defined location.

To practically represent incomplete plans in the MATSim framework the idea is to model free time windows as a special type of Leg called *EmptyTime*. Thus, when an activity finishes while the mobility simulation is running and the next leg is an *EmptyTime*, a new module of MATSim will take care of that agent, new plan elements will be inserted, and the agent will be returned to the mobility simulation to continue its simulation. The travel time attribute of these new objects is used to model the size of the time window. Following, details of this approach are presented. The figures in this section are UML diagrams, a graphic language introduced by Booch et al. (2005) to represent models from the *Object Oriented Programming* paradigm.

4.2 Passive planning model

4.2.1 MATSim population model extension

The right side of Figure 1(a) shows the extension mentioned above: the new interface *EmptyTime* extends *Leg*. In this manner, MATSim can handle incomplete plans as sequences of activities and legs. It is also necessary to differentiate incomplete plans from executable plans. In Figure 1(a), the new interface *BasePerson* extends the standard *Person* model incorporating a new method called *getBasePlan()*). Thus, a *BasePerson* object maintains a collection of plans (person’s
memory), and adds a BasePlan object, which is a plan with Activity and EmptyTime objects fixed in time, and no standard legs or flexible activities. In this way, the activities added to BasePlan objects define which activities are fixed.

### 4.2.2 Passive Planning agents

A simple extension of the MATSim agent must be implemented to integrate the proposed new processes. The new MobsimAgent must be aware of EmptyTime legs, so it doesn’t trigger leg events, and it has to contain a Planner object as shown on the top of Figure 1(b). A Planner is a new object in charge of taking planning decisions when an agent has free time. More specifically, it replaces an EmptyTime plan element for new flexible activities and legs. It contains a collection of DecisionMaker objects which solve the different dimensions for activity and trip scheduling. The Planner object takes these results and arrange the new plan elements for the agent. On the right of Figure 1(b), the type hierarchy of the DecisionMaker is presented. This approach allows to model individual DecisionMaker objects which take individual choices based on discrete choice theory, or one object which takes all the decisions like the AgendaDecisionMaker. The multi-activity scheduling algorithm introduced above is used by AgendaDecisionMaker objects.

### 4.2.3 Planning engine and Planner Manager

When an agent finishes an activity and the next plan element is an EmptyTime object, the agent will enter to a new state: PLANNING. A Planning engine object is in charge of running a series of processes for agents in this new state. Just one Planning engine object is instantiated during the whole MATSim process. This object has three functions as shown on the top of Figure 1(c):

1. DepartureHandler object: These objects are in charge of initialize agents when they start a Leg. As EmptyTime plan elements are also legs, the Planning engine object will take any agent which initializes an EmptyTime, it will register it in an internal collection (planningAgents), and it will send the Planner object of that agent to perform the activity-trip scheduling.

2. MobsimEngine object: A MobsimEngine object in MATSim has a method called doSimStep(time) which is called every time step of the mobility simulation. In this method, the Planning engine reviews agents which have finished their planning process, and put
them back into the mobility simulation. It also handles agents with unsuccessful planning, teleporting them to the future activity location and adding a penalty in their score.

3. **EventHandler object:** Finally the **PlanningEngine** works as an **EventHandler**, which are object that react to specific events triggered by the mobility simulation. The **PlanningEngine** will collect useful information for the agents that are planning, e.g. current traffic conditions or locations’ crowdedness.

Hence, the **PlanningEngine** object works as a bridge between the standard mobility simulation and the new on-the-fly planning processes. However this object is not in charge of running these planning processes. The **Planner** object of agents in the **PLANNING** state are handled by a different object of type **PlannerManager**. This object is also instantiated once in the whole MATSim simulation. In the center of Figure 1(c), it is shown that the relationship between the **PlanningEngine** and the **PlannerManager** is 1-to-1. The reason of decoupling the planning processes from the mobility simulation is computational. Planning processes can be computationally expensive, but unlike the mobility simulation, they are fully parallelizable. Thus, while a small number of threads execute the mobility simulation, a massive number of threads can execute planning processes at the same time. The relation between the **PlanningEngine** and **PlannerManager** objects maintains the synchronicity between the mobility simulation and the planning processes. Thus, if the mobility simulation is very fast compared with the planning processes the **PlanningEngine** can pause the mobility simulation.

Consequently, a **PlannerManager** object receives **Planner** objects at different times, executes them, and informs the **PlanningEngine** object when agents are ready to continue. The current implementation of the **PlannerManager** has the following characteristics:

1. It creates the available number of threads for planning processes before the mobility simulations starts at every iteration. The available number of threads is the total number minus the number of threads used by the mobility simulation.
2. It rebalances the number of planning processes for each thread. In other words, it tries to maintain the same number of **Planner** objects in each thread while new **Planner** objects are received.
3. It can cancel planning processes because of their computation duration according to a specified maximum duration.
4. It informs the **PlanningEngine** object whether an agent is ready to continue the simulation or failed its planning process.
Figure 1: MATSim extension for passive planning.

(a) MATSim population model extension for passive planning.

(b) New agent types for passive planning.

(c) Planning engine and Planner manager relationship.
5 Application of Passive Planning

In this section, results of two case studies using the new MATSim extension are presented. The first compares a standard MATSim process with a Passive Planning process in terms of the evolution of the utilities and computation issues. The second presents a multi-day scenario with an unexpected event where agents plan on-the-fly. For both experiment a 1% sample of the MATSim Singapore scenario described by Erath et al. (2012) was used.

5.1 Comparison between Passive Planning and standard MATSim

First, two evolutionary processes are executed with similar conditions. Results of the extended MATSim incorporating Passive planning are compared with results of a MATSim process using standard re-planning.

Set-up The extended version of MATSim presented in this paper (Passive Planning), allows agents to start the process with incomplete activity plans, i.e. a plan with only fixed activities specified, or a skeleton as called by Doherty et al. (2002). Using a 1% sample of the Singapore scenario, skeletons of fixed activities for the 37,425 agents were generated by deleting flexible activities and trips between all activities from their plans. It is expected that after the first iteration (a mobility simulation of one day) every agent constructs a complete plan by filling free time windows with new flexible activities. After every iteration 30% of the agents are selected at random to start the next mobility simulation with incomplete plans (i.e. with the skeleton of fixed activities).

The complete plans generated after the first iteration of the Passive Planning process are used as initial plans for a standard MATSim process (normal re-planning). The same population with these plans is simulated using three re-planning strategies: Re-Route (10%), Time Allocation (10%) and Subtour Mode Choice (10%). The idea is to compare the evolution of the daily utilities executing these two strategies.

When comparing the evolution of the utilities some conditions shown in Table 1 must be highlighted. The standard MATSim process just mutate three dimensions: routes, times, and modes, while the Passive Planning process mutates five: routes, times, modes, locations and activity types. The standard MATSim process can mutate durations and end times of fixed activities while the Passive Planning process can not. For the standard MATSim process agents
Multi-day activity models: an extension of the Multi-Agent Transport Simulation (MATSim).

August 2016

Table 1: Comparison conditions between Standard MATSim and Passive Planning.

<table>
<thead>
<tr>
<th>Condition</th>
<th>Standard MATSim</th>
<th>Passive Planning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modules</td>
<td>ReRoute, TimeAllocation, SubtourModeChoice</td>
<td>Flexible Activities</td>
</tr>
<tr>
<td>Population mutated</td>
<td>10%, 10%, 10%</td>
<td>Scheduling</td>
</tr>
<tr>
<td>Type of mutations</td>
<td>1 Best response 2 Random</td>
<td>1 Best response with 5 dims.</td>
</tr>
<tr>
<td>Scheduling dimensions</td>
<td>routes, times, and modes</td>
<td>routes, times, modes, locations and activity types</td>
</tr>
<tr>
<td>Fixed act. mutations</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Memory size</td>
<td>5 Plans</td>
<td>1 Plan</td>
</tr>
<tr>
<td>Public transport</td>
<td>Teleported</td>
<td>Teleported</td>
</tr>
<tr>
<td>Threads</td>
<td>4 mobsim 24 re-planning</td>
<td>4 mobsim 20 planning</td>
</tr>
</tbody>
</table>

Results  Results of these initial experiments show firstly that the proposed approach is computationally feasible using parallel processes in charge of several scheduling tasks. As shown in Figure 2, the first iteration took almost 3:30 minutes, it is the longest as every agent has an incomplete plan, while in the other iterations just 30% of them use the scheduler. The average computation time was 95 seconds.

Lighter lines in Figure 3 show the evolution of the utilities after running 100 iterations of the standard MATSim, while darker lines show the same evolution using the Passive Planning MATSim. For the Passive planning process, agents schedule and experience different flexible activities. As only one plan is saved in each agent memory, the evolutionary algorithm performance consists of improving flexible activities and trips with updated information, and not in the selection of remembered experiences. In the final iterations, reaching user equilibrium, the worst plan is almost always the executed one. Additionally, scheduling events are highly heterogeneous due to the different conditions of the agents when scheduling flexible activities.

Analyzing differences between the standard MATSim and Passive Planning in Figures 2 and 3, the following conclusions are presented:

• When optimizing daily experiences, to schedule flexible activities affects significantly the average final utility of a population. After 100 iterations the final average utility is
more than 140 utils scheduling flexible activities while it is just 120 utils. without this scheduling.

• Just one execution of the proposed algorithm over the entire population generates a very optimal result in terms of routes, times and modes. The standard MATSim process just improved the utility of the initial plans from 100 to less than 120 utils. after 100 iterations mutating routes, times and modes.

• The computation of the standard MATSim process is 16% faster in the initial iterations, and 47% faster in the final iterations. This happens because the standard mobility simulation speeds up with less congestion.

• The standard MATSim needs 66% less RAM memory (Standard MATSim: 10GB, Passive Replanning: 30GB). This figure should change with longer periods of time because more memory will be needed to store experienced plans with the standard MATSim, and because the mental map with agents in Passive planning doesn’t grow.
Figure 3: Evolution of the daily utilities of 37,425 agents within the MATSim evolutionary algorithm.

Source: Lighter lines show the evolution of the utilities running 100 iterations of the standard MATSim, while darker lines show the evolution of the utilities using the Passive Planning MATSim.

5.2 Passive Planning: Multi-day scenario with an unexpected event

In this scenario, four similar week days (Monday to Thursday) are simulated with an unexpected event to evaluate how differently agents schedule flexible activities using Passive planning. The idea is to decrease the utility of performing shopping activities during a specific period of time and at a specific geographical area. Expected results should show how agents decide on-the-fly not to shop there.

Set-up As the first application, this experiment is conducted with a small sample of the Singapore scenario. Almost forty thousand agents (1% sample) start the simulation with four-days long incomplete plans of fixed activities. They can schedule two types of flexible activities: shop and eat. The shop facilities were categorized in 2 groups according to their size. Thus, locations with more than 100 stores were categorized as Big shops and the rest as Small
Multi-day activity models: an extension of the Multi-Agent Transport Simulation (MATSim).

August 2016

Figure 4: Big shops in the Singapore scenario.

Source: The oval marks the geographic area where the unexpected event occurs.

shops. In the Singapore’s scenario collection of activity facilities, 37 places were categorized as Big Shops and 12,983 places as Small shops. Black pentagons in Figure 4 represent the geographic location of Big shops. Eating places and Small shops can be assumed to be located everywhere. The marginal utility of shopping in a big shop is set to 12 utils/h, of shopping in a small shop is set to 3 utils/h, and the marginal utility of eating is 6 utils/h. These values are used in the utility maximization algorithm mentioned previously for activity-trip scheduling.

To test the Passive Planning module, an unexpected event is introduced from 4pm to 9pm on Wednesday. It affects the shopping activity at facilities located inside the oval shown in Figure 4. To represent this event, shopping activities at these locations during the specified time period have a marginal utility of 0 utils/h. The simulation is executed in a computer with 130 GB of RAM memory and 12 cores with multi-threading. With these conditions, four threads were reserved for the mobility simulations and 20 threads for Passive Planning.

Results Figure 5(a) shows the number of trips to shop and eat planned by agents from 4pm to 9pm by day of the week. In general agents plan more trips to eat because the vast majority of shopping places are categorized as Small shops with a low marginal utility. Many agents prefer to eat than to travel long distances to crowded Big shops. The ratio between the number of eating and shopping activities changed on Wednesday comparing to the other days. When agents got the information about the event, they preferred to plan more eating activities than...
shopping activities on Wednesday. This change in the ratio can be appreciated more by checking the height of the lighter bars of the Figure because these bars represent the number of trips originated inside the affected area. This means that agents located inside this area changed their activity decision more than the other agents. These results demonstrate how Passive planning can model activity type choices.

Similarly, Figure 5(b) shows the number of trips to shop from 4pm to 9pm every day at different shopping location types. On Monday, Tuesday and Thursday, agents travel to Big shops much more to obtain high utilities, but this changes drastically on Wednesday when many more shop activities are performed on Small shops. As 25 of the 37 Big shops are closed on Wednesday, and the closed facilities are located at a central location, the number of trips to shop at Big shops is approximately reduced by half. The destinations of almost all of these trips are Big shops located outside the affected area (pink bar), but unexpectedly, 23 trips are still planned to Big shops inside, where every agent knows that no activity performing utility will be obtained. After analyzing these 23 cases it was found that 19 of them were trips planned to reach before 4pm, but because of traffic conditions the arrival time was at most 15 minutes later. The other 4 cases happened for a similar reason, agents planned to reach Big shops in the affected area after 9pm, but they traveled faster arriving at most 6 minutes earlier. These results demonstrate how Passive planning can model activity location choices.

Finally, Figure 6 shows four travel time distributions of trips to shop originated inside the affected area. It represents how agents located in the affected area decide to travel longer to perform shopping on Wednesday during the affected period.

6 Conclusion and future work

Few modifications are necessary to simulate multi-day scenarios in MATSim. The real challenges are related to the synthesis of the activity demand. To simplify this problem, a new approach which differentiate fixed activities and flexible activities was proposed. An extension of MATSim to initialize agents with incomplete plans of fixed activities and schedule on-the-fly flexible activities was designed implemented and tested.

The design of Passive Planning in MATSim is based on extensions of the population model, and the development of a new module. Two key elements in this new module are in charge of decoupling and synchronizing the mobility simulation and the planning processes: Planning engine and Planner manager. As planning processes are fully paralellizable, this approach allows
Figure 5: Number of trips from 4pm to 9pm by day of the week.

(a) Shop vs. Eat

(b) Big shops vs. Small shops

Lighter colors represent trips originated inside the affected area while darker colors represent all the trips.

To run in parallel any number of available threads for planning while the mobility simulation is executed.

Two case studies were prepared to evaluate the new approach. Passive Planning results show that the process is computationally feasible. The first iteration (where all agents plan flexible activities on-the-fly) takes 3.5 minutes for a 1% sample (37,425 agents) of the Singapore MATSim scenario. This is achieved running 20 threads for parallel flexible activities planning, and using 30GB of RAM memory.
Figure 6: Travel time distributions of trips to shop from 4pm to 9pm originated in the affected area by day of the week.

In the first case study, the new Passive Planning process, i.e. planning secondary activities on-the-fly, is compared with a standard MATSim evolutionary process. In 100 iterations, Passive Planning improves the average utility from 100 to 140 utils., while the standard MATSim from 100 to 120 utils. This means 100% more improvement. This happens because agents can schedule new flexible activities with new conditions of travel times every iteration. However, the evolutionary process needs 66% more RAM memory, and it is 47% slower than the standard MATSim.

In the second case study a multi-day scenario with an unexpected event is simulated. Agents react on the fly to the event by planning different activities at different places and/or different times, while a few others make planning mistakes traveling to the affected area. Comparative analyses between the affected day and the other days are performed.

With this framework, a full weekly Singapore model can be prepared from real data. Ordóñez Medina (2015b) and Ordóñez Medina (2016) present how Fixed and Flexible activities can be synthesized from commonly available data sources. Thus, realistic multi-day studies can be perform and validated. More efforts to reduce RAM memory and computation time can be made by simplifying activity scheduling tasks, or compressing mental maps of agents.
7 Acknowledgments

The author would like to thank the following Singaporean Authorities for providing access to data and valuable review: Land Transport Authority, Urban Redevelopment Authority, Singapore Land Authority (SLA Digitised Land Information). Financial support comes from the National Research Foundation (NRF) of Singapore and ETH Zurich research fund.

8 References


