

## Simulation Script Format V1.1

A simulation script is a text file that contains the information for setting up simulations. A script file is required when SMARTS runs as a distributed system without visualizing simulations at the server.

A script contains one or more blocks. Each block describes a simulation or a set of repeated simulations. A block is ended with a line that only contains three hashes, **###**. A comment line starts with a pair of slashes, **//**.

Let us use an example to describe the script format.

Line 1	maxNumSteps 10000
Line 2	numRandomBackgroundPrivateVehicles 10
Line 3	numRandomBackgroundTrams 10
Line 4	numRandomBackgroundBuses 10
Line 5	foregroundVehicleFile routeCBD.txt
Line 6	backgroundVehicleFile -
Line 7	outputInitialRoute ALL
Line 8	outputTravelTime ALL
Line 9	outputTrajectory ALL
Line 10	outputSimulationLog false
Line 11	allowReroute false
Line 12	numRuns 1
Line 13	lookAheadDistance 100
Line 14	numStepsPerSecond 1
Line 15	// A comment line
Line 16	serverBased false
Line 17	openStreetMapFile -
Line 18	trafficLightTiming FIXED
Line 19	routingAlgorithm DIJKSTRA
Line 20	trafficReportStepGapInServerlessMode 1
Line 21	driveOnLeft true
Line 22	###
Line 23	numRandomBackgroundPrivateVehicles 1000
Line 24	###
Line 25	###
Line 26	numRandomBackgroundPrivateVehicles 100
Line 27	foregroundVehicleFile -
Line 28	###

This example script includes several blocks. Each line within a block shows a parameter setting or a comment. For a parameter setting, the format is **parameter name + space + parameter value**. For example, Line 1 is “maxNumSteps 10000”. The supported parameters are detailed later in this document.

SMARTS sets all the parameters to their default values initially. When the system reads a parameter setting from the script, the parameter value is overwritten with the newly read value. The parameter setting won't change until a new value of the parameter is read from the script. For example, the

second block contains only one line, “numRandomBackgroundPrivateVehicles 1000”. Based on this line, the system will change the value of numInternalNonPubVehicles from 10 to 1000. For all other parameters, the values will be the same as in the first simulation. For example, maxNumSteps will still be 10000, as set in the first block. If there was no parameter setting within a block, the system ignores the block. For example, the empty block between Line 24 and Line 25 will be ignored. The following pages shows the supported parameters.

Parameter	Default Value	Acceptable Values
<p>maxNumSteps</p> <p>A simulation starts from step 1 and ends at the step specified by this parameter.</p>	10000000	A positive integer
<p>numRandomBackgroundPrivateVehicles</p> <p>This is the number of random private vehicles that SMARTS will try to maintain during a simulation. Any randomly generated vehicle is a <b>background</b> vehicle.</p>	100	An integer that is equal to or larger than 0
<p>numRandomBackgroundTrams</p> <p>This is the number of random trams that SMARTS will try to maintain during a simulation. This setting has no effect if the OpenStreetMap data, from which the road map is built, does not contain any tram route. Any randomly generated vehicle is a <b>background</b> vehicle.</p>	0	An integer that is equal to or larger than 0
<p>numRandomBackgroundBuses</p> <p>This is the number of random buses that SMARTS will try to maintain during a simulation. This setting has no effect if the OpenStreetMap data, from which the road map is built, does not contain any bus route. Any randomly generated vehicle is a <b>background</b> vehicle.</p>	0	An integer that is equal to or larger than 0
<p>foregroundVehicleFile</p> <p>This is the file containing the route plan of <b>foreground</b> vehicles. Different to random vehicles, the vehicles generated based on this file follow the pre-defined routes.</p>	An empty string	<ul style="list-style-type: none"> <li>Hyphen sign -, which will set the parameter to an empty string</li> <li>A string showing the path of the file</li> </ul>

Parameter	Default Value	Acceptable Values
backgroundVehicleFile  This is the file containing the route plan of <b>background</b> vehicles. Different to random vehicles, the vehicles generated based on this file follow the pre-defined routes.	An empty string	<ul style="list-style-type: none"> <li>Hyphen sign -, which will set the parameter to an empty string</li> <li>A string showing the path of the file</li> </ul>
outputSimulationLog  A log file contains some general information about completed simulations, such as simulation time and the average speed of all vehicles.	false	<ul style="list-style-type: none"> <li>true</li> <li>false</li> </ul>
outputTrajectory  A trajectory file contains the time-stamped GPS positions of vehicles. Note that a trajectory does not necessarily follow the initial route if <b>rerouting</b> was enabled.	NONE	<ul style="list-style-type: none"> <li>NONE</li> <li>FOREGROUND</li> <li>BACKGROUND</li> <li>ALL</li> </ul>
outputInitialRoute  A route file contains the initial routes of vehicles. Note that the initial routes may be different to the actual routes if <b>rerouting</b> was enabled.	NONE	<ul style="list-style-type: none"> <li>NONE</li> <li>FOREGROUND</li> <li>BACKGROUND</li> <li>ALL</li> </ul>
outputTravelTime  A travel time file contains time length of vehicles' trips.	NONE	<ul style="list-style-type: none"> <li>NONE</li> <li>FOREGROUND</li> <li>BACKGROUND</li> <li>ALL</li> </ul>
allowReroute  The system can change the route of a vehicle in certain circumstances, such as being stuck in traffic congestion for too long. Reroute can happen to any vehicle, including the vehicles generated based on pre-defined routes.	false	<ul style="list-style-type: none"> <li>true</li> <li>false</li> </ul>
lookAheadDistance  This controls the distance, within which a vehicle looks for traffic lights, front vehicles, conflict traffic at intersection.	50	A positive double value

Parameter	Default Value	Acceptable Values
<p>numStepsPerSecond</p> <p>This setting determines the step length. For example, if there are 5 steps per second, the simulation progresses with 0.2 second step length. That means a vehicle, which is travelling at 10 metres per second, moves 2 metres at each time step. If the number of steps per second increases to 10, the vehicle moves 1 metre at each time step. A high number of steps per second can result in a smooth but slow progression of traffic.</p>	5	A positive double value
<p>serverBased</p> <p>When a simulation is server-based, all the workers must report to server at each time step. The server instructs all the workers what to do next at each step. On the contrary, in serverless mode, where the server is not involved in synchronization, the workers do not need to be instructed by the server at each time step. The workers only report the simulated traffic to server at a certain interval. Generally, simulations can run faster in serverless mode. However, the workers may not be able to respond to users' input immediately as in server-based mode during a simulation.</p>	true	<ul style="list-style-type: none"> <li>• true</li> <li>• false</li> </ul>
<p>openStreetMapFile</p> <p>The file may be an original OpenStreetMap data file downloaded to the disk. It may also be a customized data file in OSM XML format.</p>	An empty string	<ul style="list-style-type: none"> <li>• Hyphen sign -, which will set the parameter to an empty string</li> <li>• A string showing the path of the file</li> </ul>
<p>trafficLightTiming</p> <p>This controls how traffic lights change colours. If this is set to NONE, traffic lights are not used.</p>	FIXED	<ul style="list-style-type: none"> <li>• FIXED</li> <li>• DYNAMIC</li> <li>• NONE</li> </ul>
<p>routingAlgorithm</p> <p>The algorithm is used for generating routes of random private vehicles.</p>	DIJKSTRA	<ul style="list-style-type: none"> <li>• DIJKSTRA</li> <li>• RANDOM_A_STAR</li> </ul>

Parameter	Default Value	Acceptable Values
<p>trafficReportStepGapInServerlessMode</p> <p>If the server needs to output information of the simulated traffic, such as vehicle trajectories, the workers need to send the information to the server during the simulation. Workers can send the information at a specified interval. The interval can vary between one step to many steps. For example, if the interval is 10, the workers report to the server at step 10, step 20, etc. The interval should not be larger than the maximum number of steps (see parameter maxNumSteps). If the interval is larger than the maximum number of steps, the workers won't send the information before the simulation ends. This gap should be preferably small if trajectories needed to be exported from the simulation.</p>	1	A positive integer
<p>numRuns</p> <p>This is the number of times that the simulation described in the current block should be repeated. If this is larger than 1, the simulation described in the current block will be repeated numRuns-1 times.</p>	1	A positive integer
<p>driveOnLeft</p> <p>If this is true, vehicles use keep-left rule. Otherwise, vehicles use keep-right rule.</p>	true	<ul style="list-style-type: none"> <li>• true</li> <li>• false</li> </ul>