SUMO - More on... Software Tests

Daniel Krajzewicz

Danilot Tete Boyom

SUMO - More on... Software Tests

Daniel Krajzewicz
Danilot Tete Boyom

$Revision: 3772 $

Table of Contents

1. Introduction 1
    - Document Information 1
    - Document Scope 1
    - Document Audience 1
    - Document Structure 1
2. Acceptance Testing using TextTest 2
    - About 2
    - Install 2
    - Running the Tests 2
    - Setting up new Tests 2
    - Coverage 2
        - DFROUTER 3
        - DUAROUTER 4
        - JTRROUTER 5
        - NETCONVERT 6
        - SUMO 8

List of Tables

2.1. Tests for DFROUTER 3
2.2. Tests for DUAROUTER 4
2.3. Tests for JTRROUTER 5
2.4. Tests for NETCONVERT 6
2.5. Tests for SUMO 8

Chapter 1. Introduction

Document Information

This document is a part of *SUMO* - *S*imulation of *U*rban *MO*bility. SUMO is an open source microscopic road traffic simulation. It can be found under http://sumo.sourceforge.net/.

This document fits to version 0.9.5. We will try to keep it up-to-date with the next versions.

Document Scope

Since version 0.9.3 we support some basic tests which prove the software functionality. The number of the available tests is not that large, yet, but it will hopefuly grow with time. Ths chapter describes how to set up the testing system and how to use it.

Document Audience

This document contains information for *developers* who extend SUMO and want to assure their changes work properly.

Document Structure

At first, the acceptance tests that come with the package are described.

Chapter 2. Acceptance Testing using TextTest

About

We use "TextTest [http://texttest.carmen.se/]" to test whether the sosftware still behaves as expected. What this application does is simply to compare an application's output (both normal and errors) and optionally also the files written by the application with previously generated files.

There are some advantages and disadvantages of this approach. On the one hand, you can guarantee that the application is doing what is wanted by comparing all outputs with files you think are right. But this is also the problem with this approach: you do not have the assurance that the files you are comparing the current outputs to ARE right - you have to check these by hand.

Anyhow, for the current progress stage, this approach should be used, because a) the tests are quite small and errors can be easily seen b) it is working.

Install

At first, you have to download and install "TextTest [http://texttest.carmen.se/]" from http://texttest.carmen.se/. The installation seems to be not that easy at first, but if you just follow the steps described in TestText's documentation, you'll have the system up and running in some minutes. Before continuing, please check whether the tests included with the TextTest distribution run before proceeding.

After installing TextTest, you have to set an environment variables in order to make the system run:

- `TEXTTEST_HOME`

  Set it to the path SUMO-tests are located in (`<SUMO_DIST>/tests` for the tests that come with the distribution)

This environment variable forces TextTest to use the specified directory as source of tests.

This should be all on installing the tests-subsystem so far...

Running the Tests

Within the `<SUMO>/tests` - folder you can find batch-files which start TextTest. "runAllTests.bat" starts TextTest for testing all apllications located in the folder, "`runNetconvertTests.bat`" will only show tests for NETCONVERT, "`runDuarouterTests.bat`" only those for DUAROUTER etc.

Setting up new Tests

Please follow the instructions for TextTest. In the following, the TextTest configuration for each application will be discussed.

Coverage

As said before, further tests have to be added. The following tests are currently available:

DFROUTER

Table 2.1. Tests for DFROUTER

Test Set Description	
Tests whether showing meta-information works	
Path	**Description**
`dfrouter/meta/help`	checks whether printing help works
`dfrouter/trips/noop`	checks what happens when no argument is given
Checks whether the detector types are computed properly	
Path	**Description**
`dfrouter/detector_types/messstrecke`	Example using the dlr-testtrack
Checks whether the detector pois are written correctly	
Path	**Description**
`dfrouter/detector_pois/messstrecke`	Example using the dlr-testtrack
Checks whether the routes are computed correctly	
Path	**Description**
`dfrouter/routes/messstrecke`	Example using the dlr-testtrack
Checks whether error handling works	
Path	**Description**
`dfrouter/errors/false_net`	Tests what happens if an unknown network is passed
`dfrouter/errors/no_output`	Tests what happens if no output is given
`dfrouter/errors/unknown_option`	Tests what happens if an unknown option is passed

Recent changes:

- Test for `--version` was removed in version 0.9.5

DUAROUTER

duarouter/trips/increasing_time_typedefined_withclass	same as increasing_time_typedefined, but the type has a class
duarouter/trips/increasing_time_unsorted	same as increasing_time, but with `--unsorted` set
duarouter/trips/increasing_time_withcolor	same as increasing_time, a color is given partially
duarouter/trips/one_time	100 trips, one departure time
duarouter/trips/one_time_notype	same as increasing_time, but no type is given in the trips
duarouter/trips/one_time_typedefined	same as increasing_time, but a type is given and defined in the trips
duarouter/trips/one_time_unsorted	the same, but with `--unsorted` set
duarouter/trips/unsorted_time	100 trips, random departure time, `--unsorted` set
duarouter/trips/unsorted_time_notype	same as unsorted_time, but no type is given in the trips
duarouter/trips/unsorted_time_typedefined	same as unsorted_time, , but a type is given and defined in the trips

Checks whether the import of routes (SUMO) works

Path	Description
duarouter/routes/increasing_time	100 routes (old style), departure time is increasing
duarouter/routes/one_time	100 routes (old style), one departure time
duarouter/routes/unsorted_time	100 routes (old style), random departure time, `--unsorted` set
duarouter/routes/newstyle_withid_increasing_time	100 routes (new style, routes have an id), departure time is increasing
duarouter/routes/newstyle_withid_one_time	100 routes (new style, routes have an id), one departure time
duarouter/routes/newstyle_withid_unsorted_time	100 routes (new style, routes have an id), random departure time, `--unsorted` set
duarouter/routes/newstyle_noid_increasing_time	100 routes (new style, routes have no id), departure time is increasing
duarouter/routes/newstyle_noid_one_time	100 routes (new style, routes have no id), one departure time
duarouter/routes/newstyle_noid_increasing_time	100 routes (new style, routes have no id), increasing departure time
duarouter/routes/newstyle_noid_unsorted_time	100 routes (new style, routes have no id), random departure time, `--unsorted` set
duarouter/routes/newstyle_increasing_colors	same as newstyle_noid_increasing_time, but colors are supplied partially
duarouter/routes/newstyle_with_vehtype	Another (!!!) color check
duarouter/routes/newstyle_with_vehtype	newstyle_withid_one_time, but the with a defined vehicle type
duarouter/routes/newstyle_with_vehtype_nodef	newstyle_withid_one_time, but the with an undefined vehicle type
duarouter/routes/newstyle_with_vehtype_withclass	newstyle_with_vehtype, but the vehicle type has a class

Checks whether the DUA works

Recent changes:

Path	Description
duarouter/dua/newstyle_dua3s_step1	the first step of the dua3s-example using current (new style) output; converts trips to routes and alternatives
duarouter/dua/newstyle_dua3s_step2	the second step of the dua3s-example using current (new style) output; converts alternatives

- Test for `--version` was removed in version 0.9.5

JTRROUTER

Table 2.3. Tests for JTRROUTER

Test Set Description	
Tests whether showing meta-information works	
Path	**Description**
`dfrouter/meta/help`	checks whether printing help works
`dfrouter/trips/noop`	checks what happens when no argument is given
Checks plain functioning	
Path	**Description**
`dfrouter/plain/plain`	A basic test
`dfrouter/plain/no_turnarounds`	As plain, but turning around is prohibited (by setting the probability to zero)
`dfrouter/plain/straight_only`	As plain, but only moving forward is allowed (by setting all other probabilities to 0)
`dfrouter/plain/straight_only`	As straight_only, but the ending edge is a sink
Checks whether the processing of turn definitions works	
Path	**Description**
`dfrouter/turns/definite_xml`	Description of two routes using turn definitions in XML
`dfrouter/turns/definite_csv`	Description of two routes using turn definitions in CSV
Checks whether error handling works	
Path	**Description**
`dfrouter/errors/no_net`	Tests what happens if the net is not given
`dfrouter/errors/false_net`	Tests what happens if the net is not available
`dfrouter/errors/no_output`	Tests what happens if the output is not given
`dfrouter/errors/false_output`	Tests what happens if the output can not be written
`dfrouter/errors/no_input`	Tests what happens if the input is not given
`dfrouter/errors/unknown_option`	Tests what happens if an unknown option is passed

Recent changes:

- Test for `--version` was removed in version 0.9.5

Table 2.4. Tests for NETCONVERT

Path	Description
netconvert/meta/help	checks whether printing help works
netconvert/trips/noop	checks what happens when no argument is given

NETCONVERT

Checks whether --speed-in-kmh works (in XML)

Path	Description
netconvert/XML2/speed_in_kmh/cross_notypes_kmh	no types, speed in kmh
netconvert/XML2/speed_in_kmh/cross_notypes_kmh_v	the same, with verbose output
netconvert/XML2/speed_in_kmh/cross_usingtypes_kmh	types, speed in kmh
netconvert/XML2/speed_in_kmh/cross_usingtypes_kmh_v	the same, with verbose output

Checks whether types may be used to describe edges (in XML)

Path	Description
netconvert/XML2/types/cross_notypes	no types
netconvert/XML2/types/cross_notypes_v	the same, with verbose output
netconvert/XML2/types/cross_usingtypes	with types
netconvert/XML2/types/cross_usingtypes_v	the same, with verbose output

Checks whether description of edge shapes works (in XML)

Path	Description
netconvert/XML2/shapes/hokkaido-japan	a simple test for the hokkaido-network

Checks whether setting connections and disabling addition of turnaround work

Path	Description
netconvert/XML2/connections/cross3l_edge2edge_conns	a simple test for setting connections edge-wise
netconvert/XML2/connections/cross3l_lane2lane_conns	a simple test for setting connections lane-wise
netconvert/XML2/connections/cross3l_no_turnarounds	a simple test to check whether appending turnarounds may be disabled

Checks whether error handling works

Path	Description
netconvert/errors/false_output	Tests what happens if the output can not be written
netconvert/errors/no_output	Tests what happens if no output is given
netconvert/errors/xml_false_edg	Tests what happens if xml-edges can not be read
netconvert/errors/xml_no_edg	Tests what happens if no xml-edges are given
netconvert/errors/xml_false_nod	Tests what happens if xml-nodes can not be read
netconvert/errors/xml_no_nod	Tests what happens if no xml-nodes are given
netconvert/errors/unknown_option	Tests what happens if an unknown option is passed

- Test for `--version` was removed in version 0.9.5

Checks whether variable speed signs work

Path	Description
`sumo/extended/variable_speed_signs/b0_offset_100`	a scenario with begin time=0, first change at time=100
`sumo/extended/variable_speed_signs/b100_offset_0`	a scenario with begin time=100, first change at time=0
`sumo/extended/variable_speed_signs/b100_offset_50`	a scenario with begin begin time=100, first change at time=50
`sumo/extended/variable_speed_signs/b100_offset_100`	a scenario with begin begin time=100, first change at time=100
`sumo/extended/variable_speed_signs/b100_offset_200`	a scenario with begin time=100, first change at time=200
`sumo/extended/variable_speed_signs/begin_0`	a scenario with begin time=0, first change at time=0

Checks whether other emitters work

Path	Description
`sumo/extended/emitter/10_time_0_1_start0_0types_1route_noid`	ten vehicles, one second between starting at 0, random vehicle type (from 1), one route, no id
`sumo/extended/emitter/10_time_0_1_start5_0types_1route_noid`	ten vehicles, one second between starting at 0, random vehicle type (from 1), one route, no id, sim start at 5 (check whether previous vehicles are discarded)
`sumo/extended/emitter/base_vehtype1from2`	as 10_time_0_1_start0_0types_1route_noid, but with a chosen vehicle type
`sumo/extended/emitter/base_vehtype2from2`	as 10_time_0_1_start0_0types_1route_noid, but with two randomly chosen vehicle type
`sumo/extended/emitter/base_withid`	as 10_time_0_1_start0_0types_1route_noid, but with ids
`sumo/extended/emitter/base_period10`	as 10_time_0_1_start0_0types_1route_noid, but with a period of 10s
`sumo/extended/emitter/base_otherroute`	as 10_time_0_1_start0_0types_1route_noid, but another route is used
`sumo/extended/emitter/base_2randomroutes`	as 10_time_0_1_start0_0types_1route_noid, but a random route is used
`sumo/extended/emitter/base_exp_routes`	as 10_time_0_1_start0_0types_1route_noid, with explicit routes

Checks whether other additional structures work

Path	Description
`sumo/extended/rerouter`	Tests whether rerouter work; checks also whether vehroutes work properly when vehicles are rerouted

Checks whether error handling works

Path	Description
`sumo/errors/unknown_option`	Tests what happens if an unknown option is passed
`sumo/errors/meandata_edges_missing_begin1`	Tests what happens if the first begin time is missing
`sumo/errors/meandata_edges_missing_begin2`	Tests what happens if the second begin time is missing

- Test for `--version` was removed in version 0.9.5