

Update:

1) memory leaks:

clear_llist() is added in mobility_parser.c to free up heap allocations

so, hashtable.c is fine now.

TBD: to check in trace.c

2) zero mobility sceario:

1	v1	x1	y1	0
3	v1	x2	y2	0
5	v1	x3	y3	0

nodes are put to sleep until 1st second, and is maintained in the this location till 3sec, later on position is updated in update_trace_nodes() to x2,y2 and is maintained till 5 sec and so on.

value of eps is changed from 10.99 to 0.09, node next position is assigned just before the next time step described in the description file.

current time in create_trace_node is started off with 0.10

3) regular scenario works fine.

4) early and late arrival scenario:

1	v1	x1	y1	0
3	v1	x2	y2	2.0

in this case when vehicle starts off at one from x1,y1 to x2,y2 and reaches at 2.5sec, it is put to sleep for 0.5sec.

in cases of late arrival where vehicle may reach after 3 seconds, then time is ignored for rest of the description.

target_time is added in def.h(mobility struct) to keep track of arrival time and to calculate the sleep duration.

5) only lines with data are read.

INPUT required:

the code works fine when sufficient data is provided, but its lost in one point,

line 1266 oaisim.c:

```
if (oai_emulation.info.omg_model_ue == TRACE)
    extract_position_fixed_ue(ue_data, NB_UE_INST);
```

where does this decision take me.

TBD: memory leak in trace.c
default mobility if no file is given
managing large mobility files
out-off boundary coordinates

