

Content

Preface.....	1
Data.....	1
Traffic lights.....	4
Traffic flow.....	5
Implementation.....	5
Example with phase request.....	5
UI in python.....	6
Experiences and Pitfalls.....	6
No detection of gaps.....	6
Pedestrian running in a row.....	6
Wrong distance of detectors.....	6

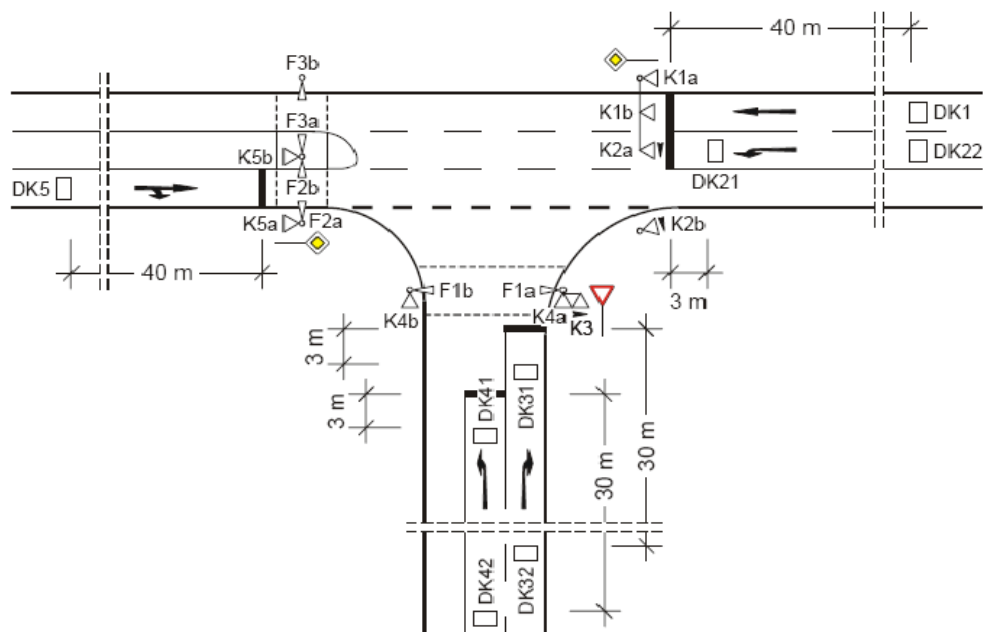
Preface

This document is written in English, because

- I used English in the last years for technical writing,
- Many students came from other countries
- I might want to publish this doc as SUMO example

Data

This example realizes the data from a lecture at TU Braunschweig (with some simplifications)



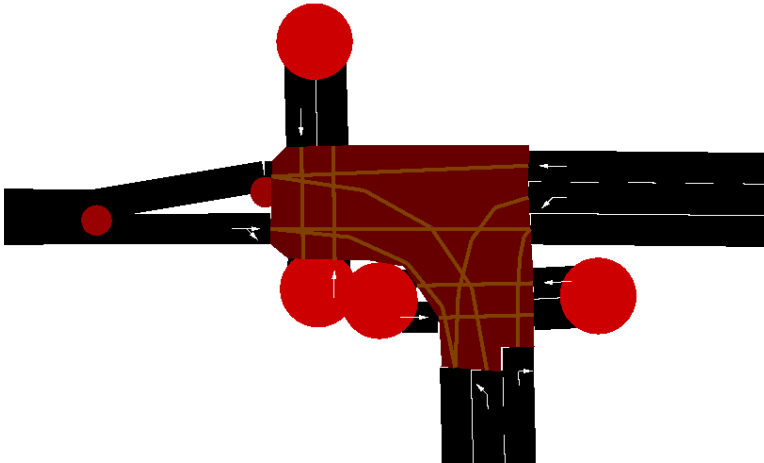
Quelle: Beispielsammlung zu den Richtlinien für Lichtsignalsteuerung – RiLSA 2010; FGSV

Note that the distance of the detectors is different (30m vs. 40m).

Source: Beispielsammlung zu den Richtlinien für Lichtsignalanlagen : RiLSA
Forschungsgesellschaft für Straßen- und Verkehrswesen / Arbeitsgruppe Verkehrsmanagement. -
Ausg. 2010. - Köln : FGSV-Verl., 2010

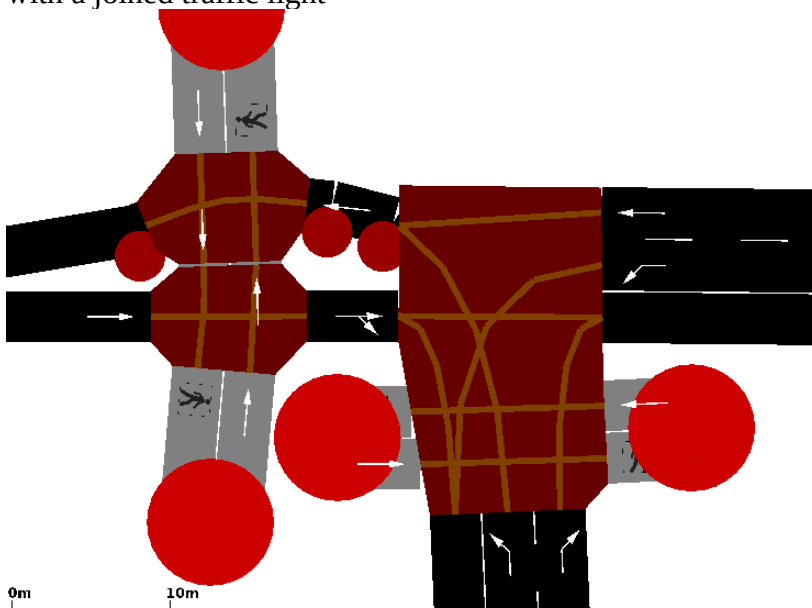
The first example (Freigabezeitanpassung) is from chapter 3.2, P.22 f

Here is the view of the connection in netedit (file junction.net.xml)

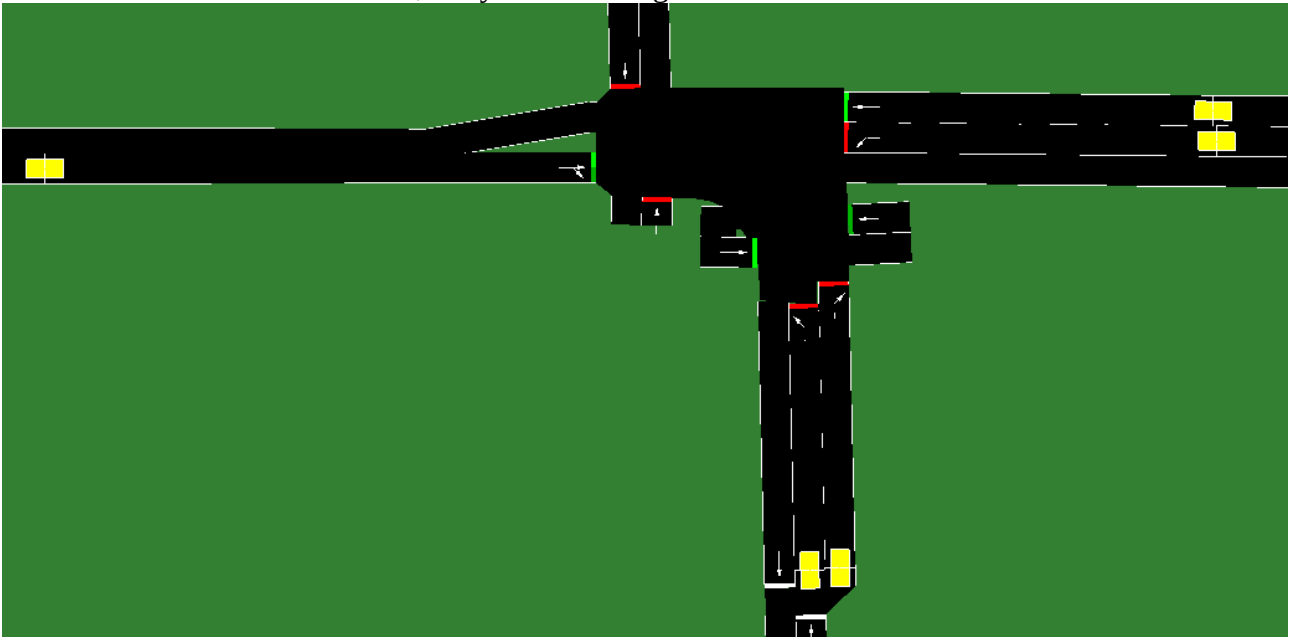


The passenger crossing in the left side is simplified.

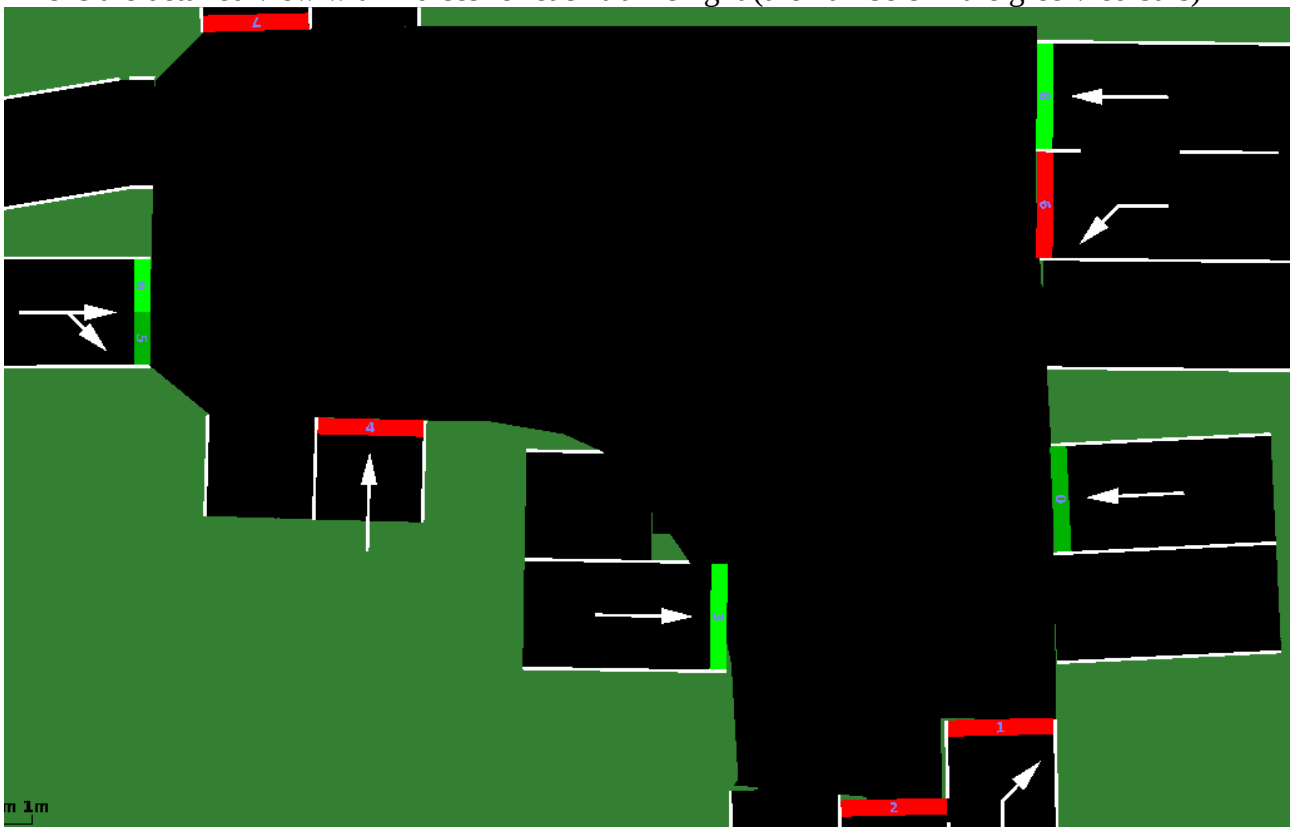
The alternative modeling in Sumo is shown below (file junction2.net.xml), it uses three junctions with a joined traffic light



This is the view in the SumoGui, the yellow rectangles are the detectors



This is the detailed view with indices for each traffic light (the numbers in the green/red bars)



Traffic lights

The idea is to create a single static program for the complete long phases.

With the TraCi interface the logical phases 1, 2 or 3 are shortened if necessary.

Here is the long program as input for the (modified) `tls_csv2SUMO.py` tool

```

key;junction1
subkey;Nachmittag
offset;0
link;F1;gneE22_0;-gneE23_0;0          #junction1;000
link;K3_4;-gneE24_0;gneE4_0;0          #junction1;001
link;K4;-gneE24_1;gneE5.14_0;0         #junction1;002
link;F1;gneE23_0;-gneE22_0;0          #junction1;003
link;F3;-gneE17_0;-gneE12_0;0         #junction1;004
link;K5;-gneE5.0_0;gneE24_0;0         #junction1;005
link;K5;-gneE5.0_0;gneE4_0;0          #junction1;006
link;F3;gneE12_0;gneE17_0;0          #junction1;007
link;K1;-gneE4_0;gneE5.14_0;0         #junction1;008
link;K2;-gneE4_1;gneE24_0;0          #junction1;009
siggrpphase;K1;0;g;54;y;57;r;84;g;89
siggrpphase;K2;0;r;39;g;54;y;57;r;89
siggrpphase;K3_4;0;r;40;g;79;y;82;r;89
siggrpphase;K4;0;r;59;g;79;y;82;r;89
siggrpphase;K5;0;g;34;y;37;r;84;g;89
siggrpphase;F1;0;g;30;r;83;g;89
siggrpphase;F2;0;r;54;g;80;r;89
siggrpphase;F3;0;r;59;g;82;r;89

```

This is turned into

```

<phase duration="30" state="grrGrgGrGr"/> logical Phase 1
<phase duration="4" state="rrrrrGGrGr"/>  -+
<phase duration="3" state="rrrrryyrGr"/>   |  PÜ1.2
<phase duration="2" state="rrrrrrrrGr"/>   |
<phase duration="1" state="rrrrrrrrGG"/>   -+
<phase duration="14" state="rGrrrrrrGG"/> logical phase 2
<phase duration="3" state="rGrrrrrryy"/>   -+  PÜ2.3
<phase duration="2" state="rGrrrrrrrr"/>   -+
<phase duration="20" state="rGGrgrrgrrr"/> logical phase 3
<phase duration="1" state="ryyrgrrrgrrr"/> -+
<phase duration="2" state="ryyrgrrrgrrr"/> |
<phase duration="1" state="rrrrrrrrrr"/>   |  PÜ3.1
<phase duration="1" state="GrrGrrrrrr"/>   |
<phase duration="5" state="grrGrgGrGr"/>   -+

```

This is the logic:

```

if tl.inPhase(1) and tl.phaseRuns() >= 6 and gapFromWest():
    tl.terminateCurrentPhase()
elif tl.inPhase(2) and tl.phaseRuns() >= 4 and gapFromEast():
    tl.terminateCurrentPhase()
elif tl.inPhase(3) and tl.phaseRuns() >= 5 and gapFromSouth():
    tl.terminateCurrentPhase()

```

Traffic flow

The traffic flow is generated by the python script on the fly .

```

flows = stream.TrafficDef()
flows.add("westen-osten-", green, "passenger", ausWesten, nachOsten, 20)
flows.add("westen-sueden-", red, "passenger", ausWesten, nachSueden, 20)
flows.add("osten-westen-", blue, "passenger", ausOsten, nachWesten, 150)
flows.add("osten-sueden-", red, "passenger", ausOsten, nachSueden, 40)
flows.add("sueden-osten-", green, "passenger", ausSueden, nachOsten, 20)
flows.add("sueden-westen-", blue, "passenger", ausSueden, nachWesten, 20)
flows.generate_routefile('passenger2.flow.xml', float(args.factor))

```

The numbers in the last parameter are the number of cars per 10 min.

The colors of the cars are chosen according to the destination.

Implementation

There are two run scripts for each example (runner.py and run4phases.py).

They use classes defined in util.py

TrafficDef (and) Stream to create vehicle flows

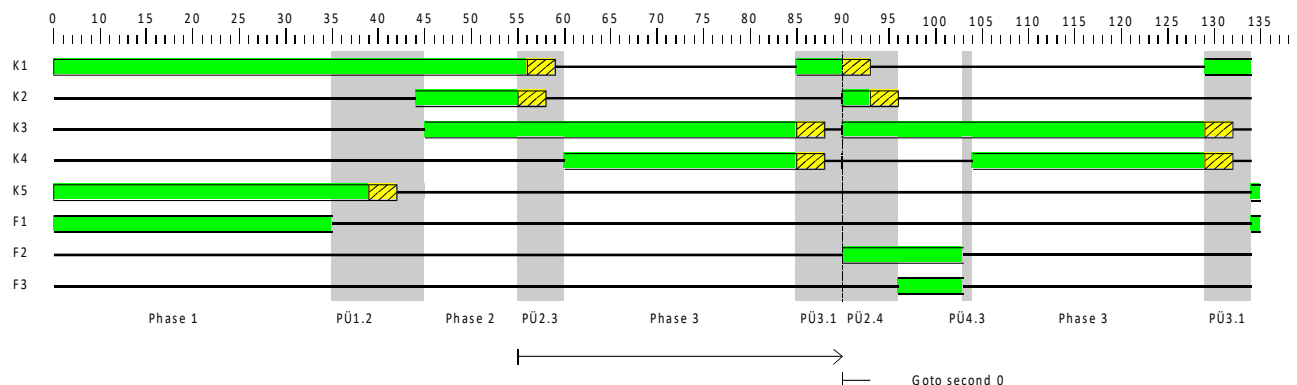
TrafficLight contains all logic for one traffic light.

Example with phase request

This example is based on RiLSA, chapter 3.4, p.25f

The idea is to combine the two phases in one SUMO TLS program. The normal loop goes from second 0 to 90. At the end of PÜ3.1 TraCi executes an unconditional jump to the beginning.

If a pedestrian request is pending, TraCi jumps at the end of (logical) phase 2 to the extend phase.



The logic looks like this

```
if tl.inSumoPhase(10) and tl.remainingPhaseDuration() == 0 and \
    tl.pedestrianGreenRequested:
    tl.clearPedestrianRequest()
    tl.gotoSumoPhase(13)
if tl.inSumoPhase(12) and tl.remainingPhaseDuration() == 0:
    tl.gotoSumoPhase(0)
```

UI in python

To run a UI, you need to install Tk/Tcl. One way for windows is: do it via cygwin (www.cygwin.com), here always consistent packages are installed. To run the example, you must first start the X-Window system (startxwin and set the display variable export DISPLAY=:0).

Another way is to use ActivePython (2.7.x) and ActiveTcl from activestate.com.

You have to add the PIL resp. pillow library for displaying buttons via

```
python -m pip install pillow
```

Experiences and Pitfalls

No detection of gaps

If the traffic stops due to congestion and the cars are standing outside the detectors, a gap is detected but the road is full of cars.

Pedestrian running in a row

You must model pedestrians as pedestrians, not as vehicle.

If they run in a row due to modeling as vehicles, they may block the road

Wrong distance of detectors

If the detectors are placed too closed to the traffic light, the cars make an emergency stop in SUMO.