Tutorial 5: Basic Platooning Implementation

Basic Platooning Implementation

Prof. Sangyoung Park

Module ”Vehicle-2-X: Communication and Control“

Configuration the Update Period

- (Continued from page 18)
- Currently, the update period of vehicle movements is 1s
- Of course this is too large and the vehicle will not properly follow
- Update of the vehicle position, velocity data is managed by veins/src/veins/modules/mobillity/traci/TraCIScenarioManager.cc
  - In function initialize(), you will see the following line
  - It reads the parameter from the omnetpp.ini file

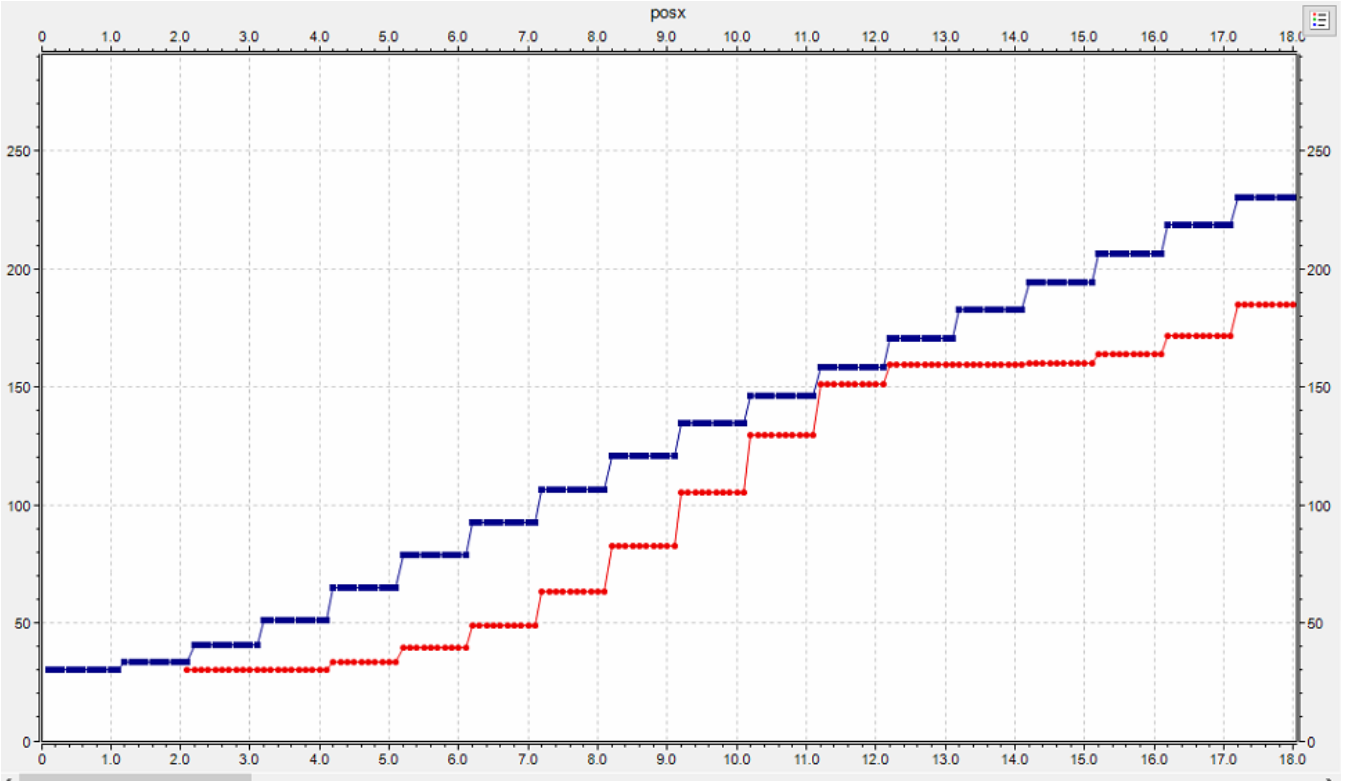
```
updateInterval = par("updateInterval");
```

  - So, let‘s update the omnetpp.ini file such that Veins update the mobility information more frequently

```
############################################################
#            TraCIScenarioManager parameters               #
############################################################
*.manager.updateInterval = 0.1s
*.manager.host = "localhost"
*.manager.port = 9999
*.manager.autoShutdown = true
*.manager.launchConfig = xmldoc("straight.launchd.xml")
```

Configuration the Update Period

- But, is this it?
  - We see that the number of data points has increased to 10 points/sec, but the actual value is not changing (plateaus)
  - It means that TraCI is reading from SUMO at 10 Hz, but the data in SUMO is not changed



Configuration the Update Period

- Update period from the SUMO side can be updated from sumocfg file
- Let’s add a line <step-length value=“0.1”\> to the file

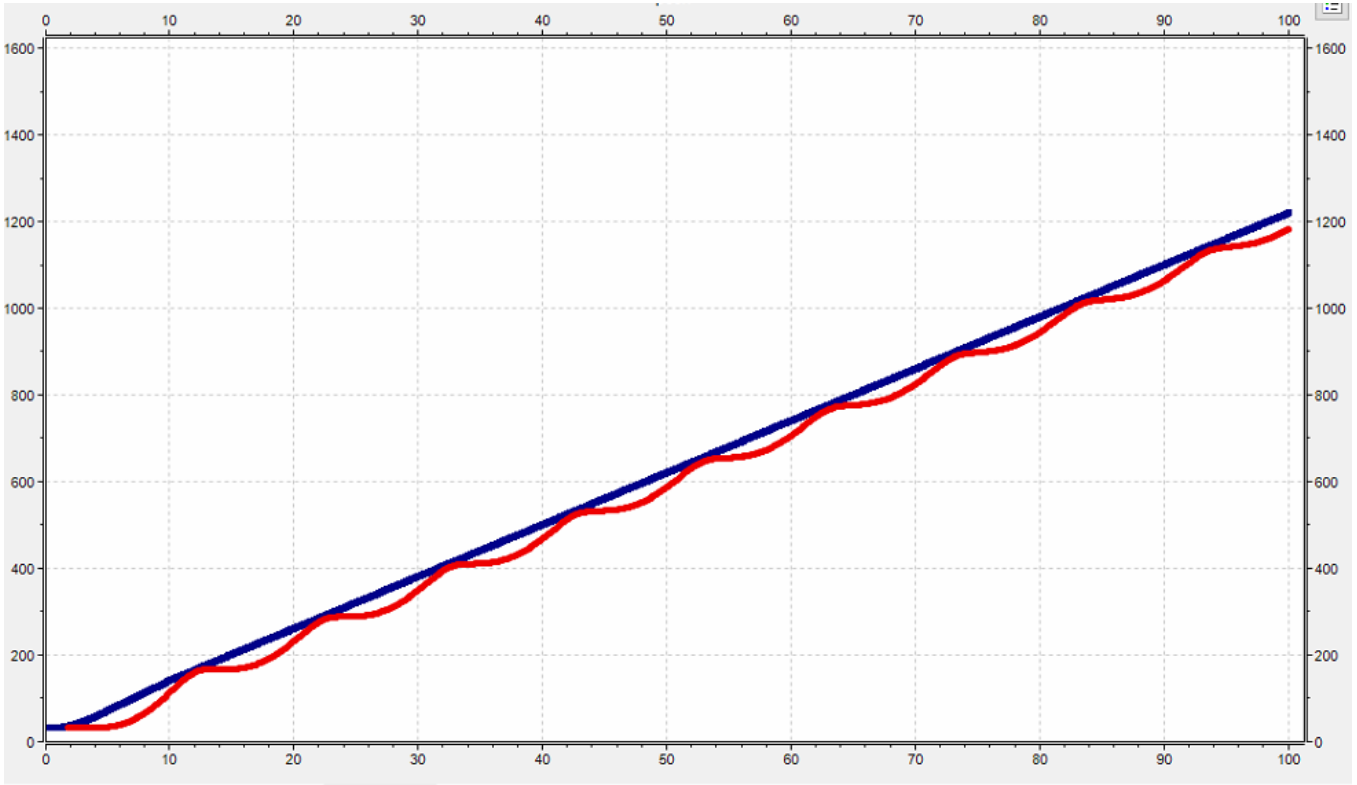
```
<configuration>
   <input>
      <net-file value="straight.net.xml"/>
      <route-files value="straight.rou.xml"/>
   </input>

                  <step-length value="0.1" />
   <time>
       <begin value="0"/>
      <end value="100"/>
   </time>

</configuration>
```

Result Graph

- X position vs time graph on a straight road
- It doesn't look that stable: Needs parameter tuning by you!



A Few More Steps

- We'd better have the speed mode to 0x06
  - It regards max acceleration and deceleration and nothing else
  - If you use 0x1f like last time, the vehicle's maximum speed is clamped to 14 m/s, which is boring..
  - Also, for some reason, setDecel() is ineffective, but rather the vehicle follows "emergency deceleration" value, we should implement setEmergencyDecel();, which is shown on the next page
  - slowdown() seems to work better than setSpeed(), so I modified it as well

```
if ( distance - desiredDistance > 1){
            traciVehicle->setSpeedMode(0x06);
            traciVehicle->setAccel(acc);
            //traciVehicle->setSpeed(14);
            //double check = traciVehicle->getMaxAcc();
            traciVehicle->slowDown(100.0, 0);
        }
        else if (  distance - desiredDistance < -1 ) {
            traciVehicle->setSpeedMode(0x06);
            //traciVehicle->setDecel(1);
            //double check = traciVehicle->getMaxDecel();
            traciVehicle->setEmergencyDecel(-acc*5);
            //traciVehicle->setSpeed(0);
            traciVehicle->slowDown(0.0, 0);
        }
    }
```

Set Emergency Deceleration

- Add the following function to the TraCICommandInterface.cc
- Of course you should edit the header file as well (remember last tutorial?)
- VAR_EMERGENCY_DECEL is also not defined
  - In TraCIConstants.h, add the #define Macro (0x7b)

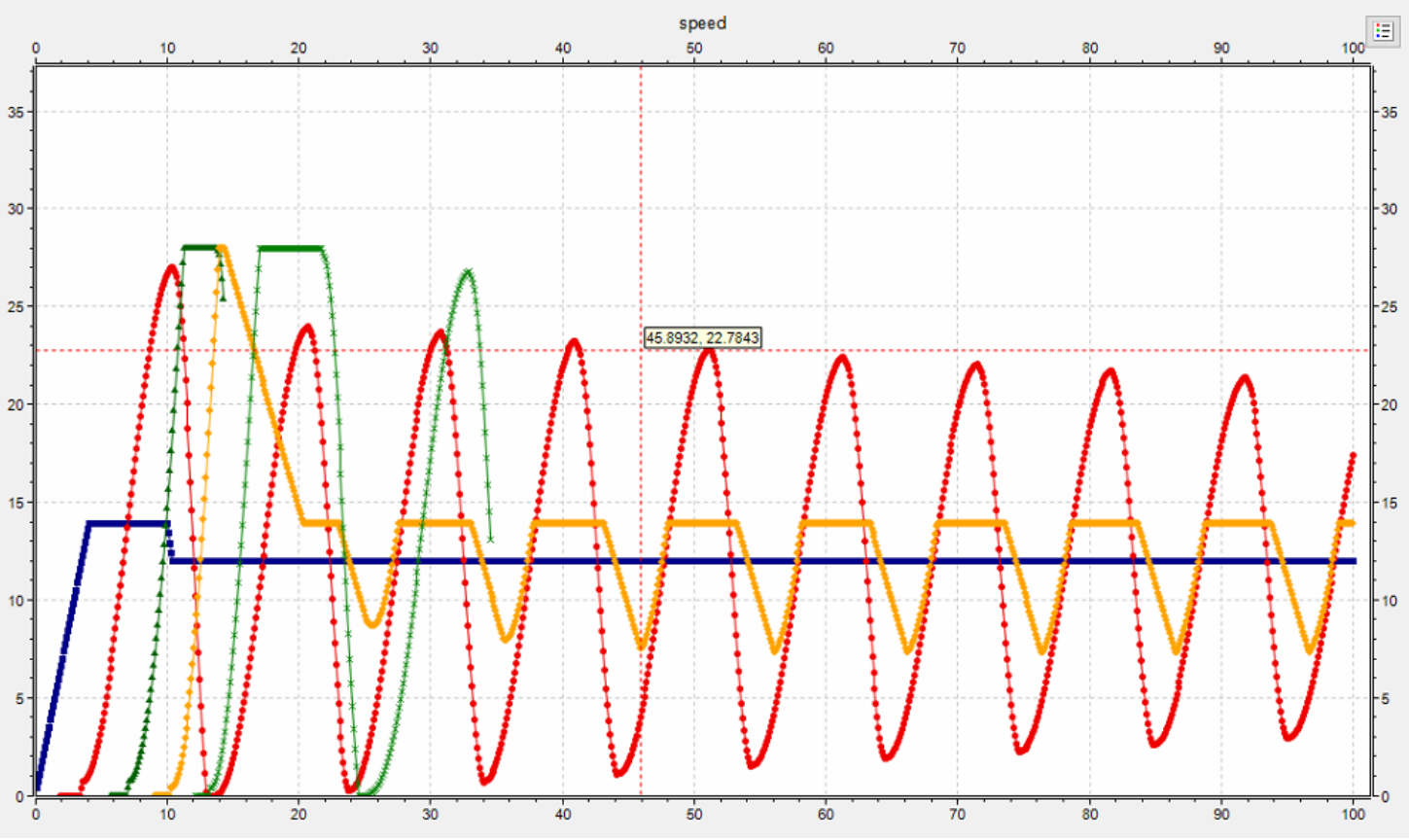
```
void TraCICommandInterface::Vehicle::setEmergencyDecel(double decel) {
    uint8_t variableId = VAR_EMERGENCY_DECEL;
    uint8_t variableType = TYPE_DOUBLE;
    TraCIBuffer buf = traci->connection.query(CMD_SET_VEHICLE_VARIABLE,
TraCIBuffer() << variableId << nodeId << variableType << decel);
    ASSERT(buf.eof());
}
```

```
// max. deceleration (get: vehicle types)
#define VAR_DECEL 0x47

// added by spark
#define VAR_EMERGENCY_DECEL 0x7b
```

Let’s increase the number of vehicles

- I’ve increased the number of vehicles to five by editing .rou.xml file
- Uh oh, it’s not string stable. Perturbation in the first vehicle is propagated and the third and fifth vehicle crashes



Parameter Tuning

- How do we make the platoon stable?
- Could you tune the parameters?
  - Use different coefficients for acceleration and deceleration?
  - Use PI control instead of P control?
  - Should we take “velocity” into consideration as well?
- Try various things yourself now!