AA-project-documentation-Qin

Qin Lin

July 2019

1 Reachability using flow*

1.1 Model and control

$$\dot{x} = cos(\psi) \cdot v \tag{1}$$

$$\dot{y} = sin(\psi) \cdot v \tag{2}$$

$$\dot{\psi} = \frac{tan(\delta) \cdot v}{vehicle.L} \tag{3}$$

$$\delta = \arctan(\frac{2 \cdot vehicle.L \cdot sin(\alpha)}{Lf}) \tag{4}$$

$$Lf = k \cdot v + Lfc \tag{5}$$

$$\alpha = \arctan(\frac{y_d - y}{x_d - x}) - \psi \tag{6}$$

where $vehicle.L$ length of vehicle, x_d, y_d reference position, Lfc look-ahead-distance

replace δ in (4) to (3):

$$\dot{\psi} = \frac{2 \cdot vehicle.L \cdot sin(\alpha)}{Lf \cdot vehicle.L} \cdot v \tag{7}$$

replace Lf in (5) to (8):

$$\dot{\psi} = \frac{2 \cdot vehicle.L \cdot sin(\alpha)}{(k \cdot v + Lfc) \cdot vehicle.L} \cdot v \tag{8}$$

$$av = \arctan(\frac{y_d - y}{x_d - x}) = \frac{\pi}{4} \cdot \frac{y_d - y}{x_d - x} + w \tag{9}$$

$$av = \arctan(\frac{y_d - y}{x_d - x})' = \frac{\pi}{4} \cdot \frac{-\dot{y}(x_d - x) + (y_d - y)\dot{x}}{(x_d - x)^2} \tag{10}$$

According to the initial values of x, y, x_d, y_d, we can get the initial interval of av using interval arithmetic.

$$\begin{cases} \dot{x} = cos(\psi) \cdot v, \\ \dot{y} = sin(\psi) \cdot v, \\ \dot{\psi} = \frac{2 \cdot vehicle.L \cdot sin(av - \psi)}{(k \cdot v + Lfc) \cdot vehicle.L} \cdot v, \end{cases} \tag{11}$$

1.2 Installation of flow* in RC car

Environment: RC car
Ubuntu 16.04 LTS
Memory 7.7 GiB
Processor ARMv8 Processor rev 3 (v8l) * 4
Graphics NVIDIA Tegra X2 (nvgpu)/integrated OS type 64-bit

1. Refer the libraries dependency at Table 1 in flow* manual[1];

2. Go to gmplib website download gmp-6.1.2.tar.lz; extrat, ./configure make sudo make install.
   Mpfr website; mpfr-4.0.2.tar.xz; then same operation as the gmplib.
   Gsl? Seems that already installed;
   glpk ; glpk-4.65.tar.gz;
   Bison already installed?
   Flex already installed?
   sudo apt-get install gnuplot

3. Download flow* 2.1.0; make
   An error: libmpfr.so.6 cannot open shared object file no such file ordirectory.
   solution: Sudo apt-get update
   Sudo apt-get install libmpfr4
   Sudo apt-get install –reinstall libmpfr4

4. Testing
   Go to flowstar file, run Flow < neuron.model (download from flow start website)
   When Computation finished, several files will be created such as images, outputs, counterexamples
   For visulization: run gnuplot "/outputs/neuron.plot", an eps figure will be generated in images file

1.3 Integrating flow* into ROS

Flow* file is normally for either static analysis (constant control input) or the case that the controller can be written as differential equation. As shown in Equa. (11), the control input is x_d, y_d, v. We chunk these target points in a piece-wise way (the frequency TBD). The general continuous reachability analysis is as follows:

1. write a template of flow* file define $y_d = Y_D$, $x_d = X_D$, $v_d = V_D$;

2. Replace X_D, Y_D, V_D with values in each iteration and store as a new file;

3. Execute the new flow* file;

[1] https://www.cs.colorado.edu/∼xich8622/manual/manual-2.0.0.pdf

4. Visualize the reachable state;

5. Goto step 1

The above framework is implemented into ROS, ROS is able to obtain the current states of the vehicle (this part of communication will be implemented soon).

2 Reachability using CORA2018

1. Without installing any extra package, copy the whole file into your Matlab work directory.
2. The main file is in CORA_2018/examples/contDynamics/RC_CAR 3. The model file is in models/Cora/RCeq_kinematic_pid

I tested lots of simulations on CORA but unfortunately I'm not very happy about the run-time performance. Here I only uploaded a simple example.