MotionSynthesize

$$\begin{aligned}
\texttt{MotionSynthesize}(z_{i,k}(t_0'), t_0', t_f', y) &:= \\
\arg\min_{x:[t_0',t_f']\to\mathbb{R}} \int_{t_0}^{t_f} |x(t)|dt & \\
\text{subject to } & \ddot{x}(t) = u(t), \text{ for all } t \in [t_0', t_f']; \\
& 0 \leq \dot{x}(t) \leq v_m, \text{ for all } t \in [t_0', t_f']; \\
& |u(t)| \leq a_m, \text{ for all } t \in [t_0', t_f']; \\
& |x(t) - y(t)| \geq l, \text{ for all } t \in [t_0', t_f']; \\
& x(t_0') = x_{i,k}(t_0'); \quad \dot{x}(t_0') = \dot{x}_{i,k}(t_0'); \\
& x(t_f') = 0; \quad \dot{x}(t_f') = v_m,
\end{aligned}$$

where initial state $z_{i,k}(t_0') = (x_{i,k}(t_0'), \dot{x}_{i,k}(t_0'))$.

AMPL implementation

Using the AMPL modeling language, we can almost immediately implement the above linear program such that it can be read by a modern solver.

MATLAB implementation

We start by expressing $v = (v_1, \ldots, v_N)^T$ in terms of the decision variables $u = (u_0, \ldots, u_{N-1})^T$. From the initial condition $v_0 = v_m$ and the relation $v_{i+1} = v_i + u_i \cdot \Delta t$, we obtain

$$v = v_m \mathbb{1} + Au,$$

with the lower triangular matrix

$$A = \begin{pmatrix} \Delta t & 0 & & \\ \Delta t & \Delta t & 0 & \\ \vdots & \ddots & \ddots & \ddots \end{pmatrix},$$

which can be constructed using the following Matlab code:

```
A = tril(delta_t * ones(N))
```

Similarly, we express $x = (x_1, \ldots, x_N)^T$ in terms of v. From $x_0 = -L$ and the relation $x_{i+1} = x_i + (v_i + v_{i+1}) \cdot \Delta t/2$, we obtain

$$x = -L\mathbb{1} + Bv,$$

with the matrix

$$B = \Delta t/2 \cdot \begin{pmatrix} 1 & 1 & & & \\ 1 & 2 & 1 & & \\ 1 & 2 & 2 & 1 & \\ \vdots & \vdots & \vdots & & \ddots \end{pmatrix},$$

which can be constructed using the following Matlab code:

```
B = delta_t / 2 * (tril(2 * ones(N), 1) - diag(ones(
    N - 1, 1), 1) - [ ones(N, 1) zeros(N, N-1) ])
```

Constraints

We can now start constructing the matrix C and the righ-hand side b.

The acceleration constraints

$$-a_m \le u_i \le a_m$$

can simply be encoded as

$$\begin{aligned} C_1 = I,\ b_1 = a_m \mathbb{1}, \\ C_2 = -I,\ b_2 = a_m \mathbb{1}. \end{aligned}$$

The constraints on the velocities

$$0 \le v_i \le v_m$$

are encoded as

$$\begin{aligned} C_3 = A,\ b_3 = 0, \\ C_4 = -A,\ b_4 = v_m \mathbb{1}. \end{aligned}$$

In order to encode constraint $x_N = 0$, observe that

$$\begin{aligned} x &= -L\mathbb{1} + Bv \\ &= -L\mathbb{1} + B(v_m \mathbb{1} + Au), \\ &= -L\mathbb{1} + v_m B\mathbb{1} + BAu. \end{aligned}$$

Let $M[N]$ denote the N-th row of matrix M, then

$$x_N = -L + v_m(B\mathbb{1})[N] + (BA)[N]u,$$

so the constraint is encoded as

$$\begin{aligned} C_5 = (BA)[N],\ b_5 = L - v_m(B\mathbb{1})[N], \\ C_6 = -(BA)[N],\ b_6 = -L + v_m(B\mathbb{1})[N]. \end{aligned}$$

In order to encode constraint $v_N = v_m$, observe that

$$v_N = v_m + A[N]u,$$

so the constrain is encoded as

$$\begin{aligned} C_7 = A[N],\ b_7 = 0, \\ C_8 = -A[N],\ b_8 = 0. \end{aligned}$$

The constraints for keeping a safe distance to the vehicle ahead, given by

$$x_i \le y(t_0 + i \cdot \Delta t) - l, \quad \text{for all } i$$

Objective

Finally, the objective is encoded as

$$\begin{aligned} \max_u \sum_{i=0}^{N} x_i &= \max_u \mathbb{1}^T x \\ &= \max_u \mathbb{1}^T(-L\mathbb{1} + v_m B\mathbb{1} + BAu) \\ &= \max_u \mathbb{1}^T BAu. \end{aligned}$$