Improved Drive Cycle Following with ILC Supported Driver Model

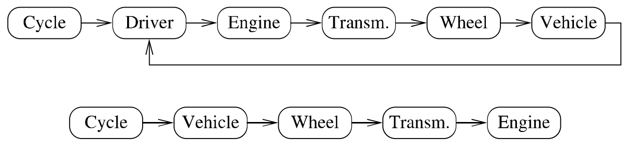
Lars Eriksson[1] and Mikael Norrlöf[1,2]

[1]Dept. of E.E., Linköping University, [2]ABB Robotics, Västerås

Background

A driving cycle is a vehicle speed as a function of time. Driving cycles are produced by different countries and organizations to assess the performance of the vehicles in different ways.

In the literature there are essentially two approaches to find the input variables to the car given a driving cycle and schematic illustrations of the two approaches are shown below.

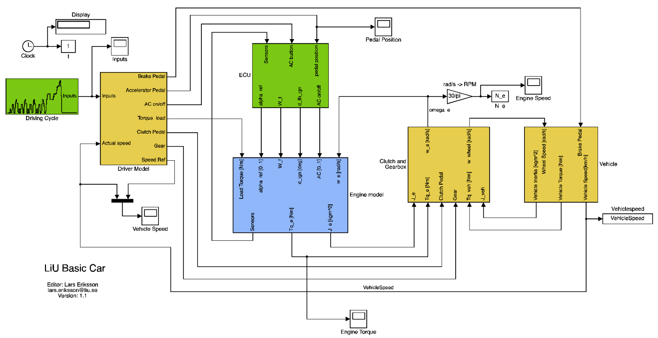


(Above)Forward simulation. (Below) Inverse simulation, also called Quasi Steady Simulation (QSS)

Vehicle model

A complete vehicle model for longitudinal motion that includes significant nonlinear elements is used. The vehicle model is composed by:

1. A naturally aspirated engine (rate limited throttle movement, filling and emptying intake manifold dynamics, volumetric efficiency for engine flow, torque model with ignition influence).
2. Driveline (clutch, stiff driveline, rolling conditions at the wheel). The clutch model contains the logics for break up and lock-up of the clutch and its logic keeps track of the wheel speed as well as the engine speed.
3. Vehicle model (longitudinal mass model, rolling resistance, air drag model).



Vehicle model implemented in Simulink.

ILC Improved Driver

The basic idea in the ILC framework is to use the error from one iteration of a repetitive action and based upon this information compute an update to the reference or control input to the system.

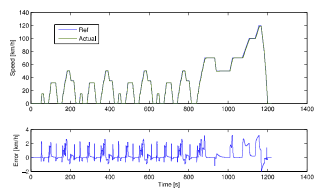
The input to the system is the speed reference and the output is the actual speed of the vehicle. In each iteration k, $t \in [0, T_f]$, for some final time T_f. It is also assumed that the states are reset so that $x_k(0) = x_0(0)$ in all iterations $k > 0$. A general first order ILC algorithm is used,

$$u_{k+1}(t) = Q(q)(u_k(t) + L(q)e_k(t))$$

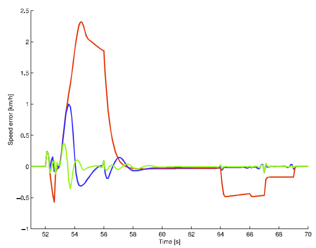
where the error is defined as $e_k(t) = r(t) - y_k(t)$. The control signal $u_k(t)$ is applied as a correction term to the speed profile, which acts as the input to the driver model. $L(q)$ is chosen as $L(q) = \gamma q^{\kappa}$ and $u_0(t) = 0$.

Simulation results

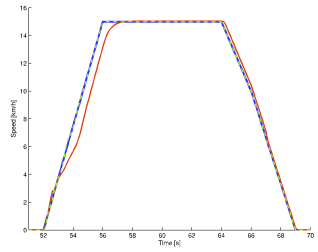
To find the values of the parameters γ and κ some simple design rules are applied. The static gain from speed reference to actual speed is close to one, therefore the gain from error to speed reference compensation, γ, is chosen close to one, here $\gamma = 0.95$. The parameter κ relates to the delay of the system and it is chosen to compensate for it in a non-causal way. The 2-norm of the error is shown for different choices of κ. $\kappa = 2$ is chosen because the 2-norm of the error is monotonically decreasing as a function of iteration.



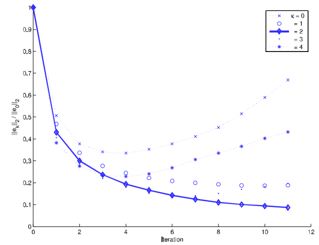
Top– NEDC speed profile, Bottom – speed tracking error.



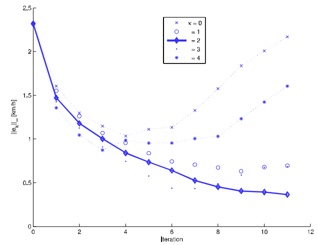
Error without ILC, e_0 (red), after 3 iterations, e_3 (blue), and after 11 iterations, e_{11} (green).



Speed tracking, reference speed (solid blue), nominal speed profile, y_0 (solid red), and the speed profile after 11 iterations, y_{11} (dashed green).



Normalized 2-norm of the error, $\frac{||e_k||_2}{||e_0||_2}$, as a function of iteration for different choices of κ in the ILC algorithm.



Maximum error, $||e_k||_\infty$, as a function of iteration for different values of the parameter κ in the ILC algorithm.

Results using ILC

- With ILC the speed converges to the desired profile in a few iterations
- The final level of the 2-norm after 11 iterations is less than 10% of the value without ILC
- The error is decreased from a level of nearly 2.5 km/h to below 1 km/h after 3 ILC iterations, that is, one full drive cycle
- High performance is achieved with very little development and tuning effort

Future work

- Extensions to cope with a decoupled driveline
- ILC design based upon time weighted optimization
- Make experiments in both the vehicular systems engine lab as well as the vehicular systems vehicle propulsion lab on a complete vehicle driveline with a driver robot