Using computers to predict the effect of stopping automation algorithms early

Automation algorithms are used all around you even if you are not aware of it. These algorithms control an automation process which you want to behave in a certain way but completely automatically. Take for example the cruise control in a car which is set to keep a certain speed whether it travels uphill or downhill. What the algorithms do is compute a control signal based on the output, which for this case is the speed of the vehicle. If the speed is low, the control signal tells the process to increase the throttle, and if it is too high it does the opposite.

The algorithms used in the thesis, ”Study of early termination of MPC algorithms”, are slightly complex and in order to determine which control signal they need, they solve an optimization problem. It takes a certain number of iterations for the algorithm to find the correct control signal and if, for some reason, the algorithm does not have time to complete all iterations, the control signal might be different than if it had time to perform all iterations.

If we simplify this for the cruise control example, two things can happen. Either the control signal is good enough and the cruise control does its job and follows the set speed, or it is not good enough and the cruise control makes the car accelerate too much or slow down too much; which leads to discomfort, and possibly dangers, during your ride. With the help of computers it is possible to predict whether or not the control signal will be good enough for its task when the algorithm does not have time to finish. In order for the computers to be able to predict this, they use something called Machine Learning. The method trains a computer on a set of data which gives it the capability to make predictions on a new set of data it has not seen before.

Machine learning makes it easier to choose an algorithm depending on the task at hand. If it is a task where the algorithm often does not have time to perform all its iterations, the computer could then pick a suitable algorithm that matches the task well even if the algorithm gets stopped too early.

In the thesis mentioned earlier this is described in detail. Several tests with different tasks, which were not real life cases but a computer model of them, with different algorithms were performed. The computer then learned from these tests and tried to predict for a new test if the algorithm can be stopped earlier. For one particular algorithm, the prediction worked on 97.3% of all the different scenarios, which means that it can tell if the control signal will be good or bad if the algorithm does not have time to finish in 36 cases out of 37!

Written by: Gustav Henriks