A collection of projects

Dipl.-Ing (EE) Alexandros Filotheou, M.Sc.

OCT 2017

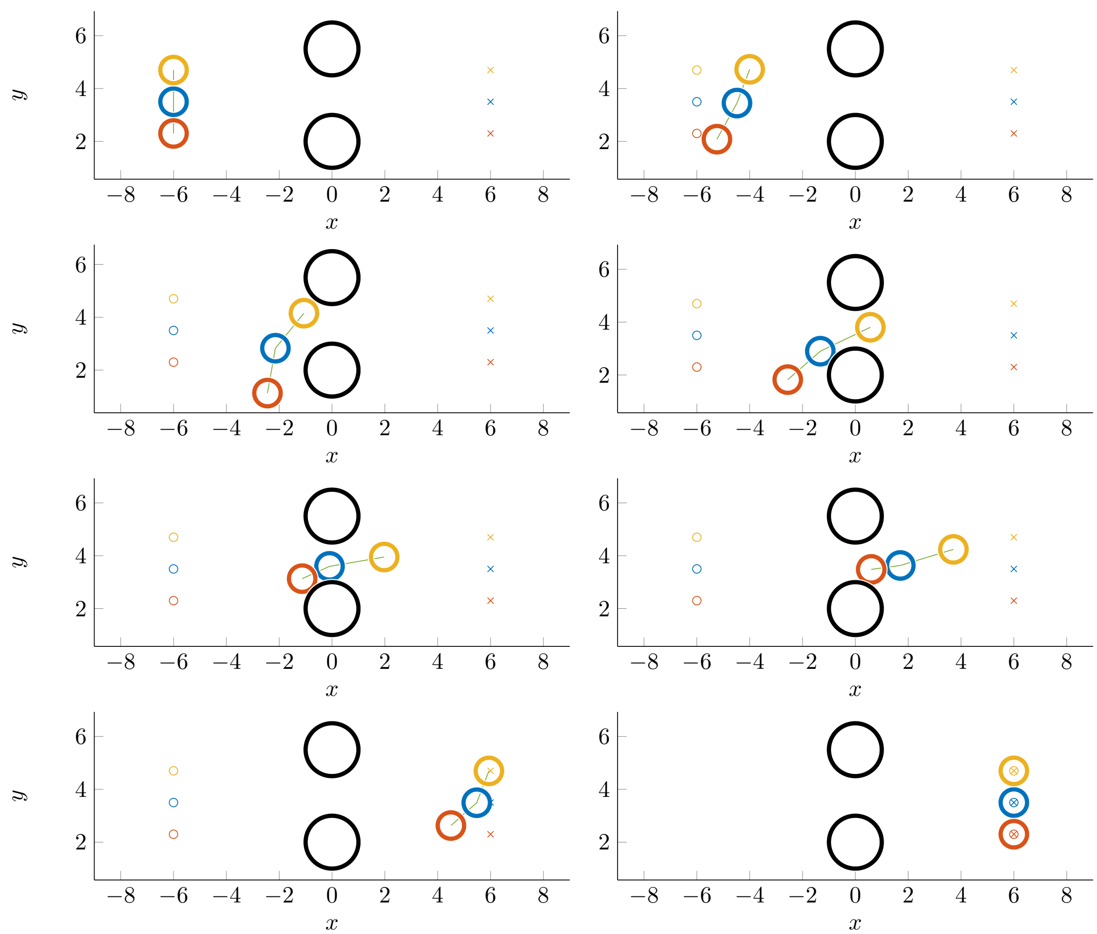
Contents

1 Control Projects

1.1 Robust Decentralized Control of Cooperative Multi-robot Systems

Context: M.Sc. Degree Project, Royal Institute of Technology (KTH), Stockholm, Sweden.

Problem statement: suppose that in a given 3D physical workspace there are a number of objects, and a number of agents whose control we are after, and whose motion is described by non-linear continuous time dynamics. All agents are constrained (a) in being within the workspace boundaries and (b) in avoiding collisions with each other and with the obstacles in the workspace. Some agents are constrained in keeping certain distance bounds from other agents. Given this setting and these constraints, the goal is for the whole multi-agent system to navigate itself from some initial configuration to a goal configuration and to stay there (be stable there), regardless of disturbances affecting the agents. Figure 1 shows a three-agent system passing through a narrow gap between two objects before reaching its intended configuration and without violating its specified constraints.



Trajectories of three agents in the $x-y$ plane. A faint green line connects agents who are to stay within distance certain bounds from one-another. The obstacles are black. Mark O denotes equilibrium configurations. Mark X marks desired configurations.

One solution of this problem involves the design of a robust decentralized model predictive control regime for the team of cooperating robot systems. The problem involves agents whose dynamics are independent of one-another, and its solution couples their constraints as a means of capturing the cooperative behaviour required.

In this work, analytical proofs are given to show that, under the proposed control regime: (a) Subject to initial feasibility, the optimization solved at each step by each agent will always be feasible, irrespective of whether or not disturbances affect the agents. (b) Each (sub)system can be stabilized to a desired configuration, either asymptotically when uncertainty is absent, or within a neighbourhood of it, when uncertainty is present, thus attenuating the affecting disturbance. In this context, disturbances are assumed to be additive and bounded.

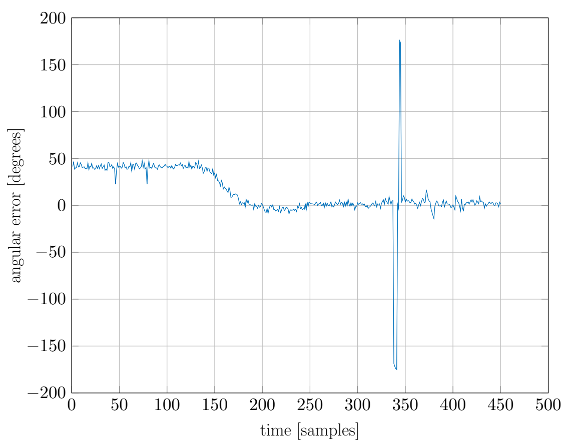
Simulations verify the efficacy of the proposed method over a range of different operating environments.

Resources / tools involved: Predictive Control, MATLAB, git

1.2 Tracking the centerline of a lane with a RC car

Context: Automatic Control Project Course EL2425, Royal Institute of Technology (KTH), Stockholm, Sweden.

Problem statement: suppose a remotely controlled vehicle chassis of Ackermann steering, equipped with a laser rangefinder and a computing unit. Given a straight path (something that simulates a road lane, e.g. a corridor), the goal is for the vehicle to navigate the path while always staying in the middle of it. The solution to the problem involves solving two distinct and independent sub-problems, centered around the (a) translational error, and (b) the rotational error, with respect to the middle line of the path. Furthermore, the problem can be approached by two ways: through the design of a PID controller, which is easier to setup, faster in execution, but potentially difficult to tune, and through that of a MPC controller, which requires more work, is slower in execution, but is more robust than the PID controller.



The angular error of the vehicle controlled via PID control.

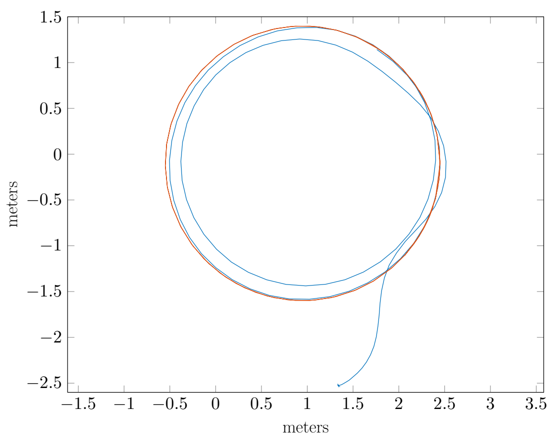
Sample results can be found in the following video sequences:
https://www.youtube.com/watch?v=w3Wnw5SLmss
https://www.youtube.com/watch?v=937OZez1iN8?t=142

Resources / tools involved: PID / MP control, ROS (Linux), Python, git, MATLAB.

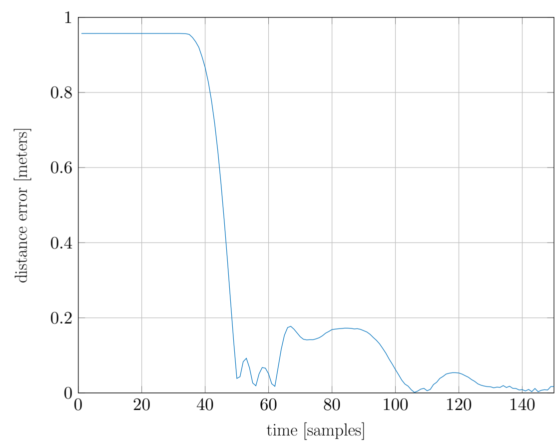
1.3 Tracking the circumference of a circle with a RC car

Context: Automatic Control Project Course EL2425, Royal Institute of Technology (KTH), Stockholm, Sweden.

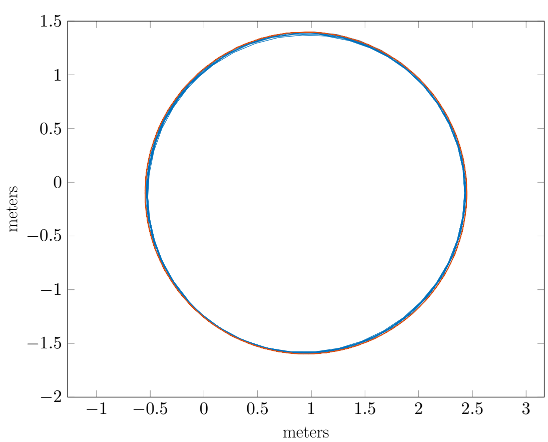
Problem statement: suppose a remotely controlled vehicle chassis of Ackermann steering that can be localized in space, equipped with a computing unit. Given a circular path, the goal is for the vehicle to navigate the path as close as possible. The solution was sought after using Model Predictive Control with variable reference poses within the horizon of the optimization problem. Figures 3 and 5 show the trajectory of the vehicle from a top view during the transient phase and the steady-state phase respectively. Accordingly, figures 4 and 6 show the respective errors in each phase as a function of time. These figures refer to actual experiments, not simulations. Notably, the steady-state error does not exceed **3.5**cm, that is, the center of gravity of the vehicle does not diverge more than **3.5**cm from the reference circular trajectory.



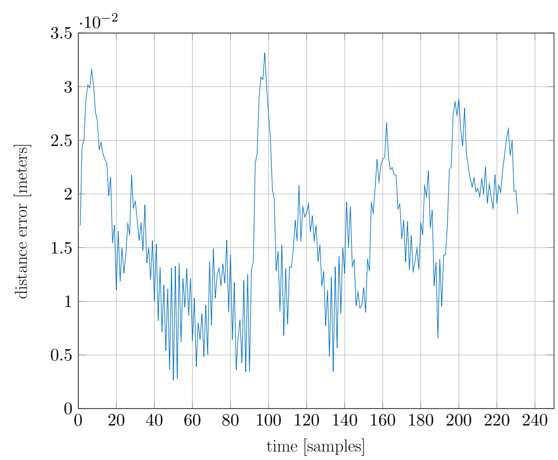
Reference trajectory (red) and trajectory of the vehicle (blue), in the transient phase.



The discrepancy in distance between the trajectory of the vehicle and the reference trajectory in the transient phase.



Reference trajectory (red) and trajectory of the vehicle (blue), in steady state.



The discrepancy in distance between the trajectory of the vehicle and the reference trajectory in steady state.

Sample results can be found in the following video sequences:
https://www.youtube.com/watch?v=Vh1huYlyD_8
https://www.youtube.com/watch?v=937OZez1iN8?t=69

Resources / tools involved: Model Predictive Control, ROS (Linux), Python, git, MATLAB.

2 Estimation Projects

3 Machine Learning Projects