Mobile robots

Nour Khaddaj, Anna Hauschild

July 23, 2023

1 Content of this lab

The goal of this lab is to control a 2D mobile robot in order to follow a trajectory. Two mobile robots will be considered:

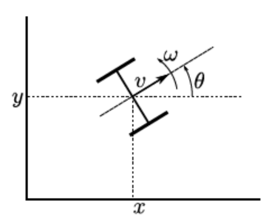
- A 2-0 or unicycle robot, with two actuated, fixed wheels.
- A 1-1 or bicycle-like robot, with two passive fixed wheels and an actuated, steering wheel.

Two control laws will be tested for each robot: static feedback and Lyapunov-based control. In addition, robustness of the control will be analyzed with regards to several practical issues:

- Bad calibration, ie wrong dimensions of the robot model
- Non linearities, which are typically found with saturation on wheel velocities or angles.

The control laws will be tested in simulation with Matlab/Simulink. Trajectory generation is partially given and we assume a localization method is here and provides the current posture (x, y, θ) of the robot.

2 Differential drive 2-0 robot



State	(x, y, θ)
Parameters	(base, radius)
Control	$\mathbf{u} = (v, \omega)$
Kinematic model	$\begin{cases} \dot{x} = v\cos\theta \\ \dot{y} = v\sin\theta \\ \dot{\theta} = \omega \end{cases}$

Figure 1: 2-0 robot

2.1 Static feedback

For

$$\mathbf{x}_p = \begin{cases} x_p = x + d.cos(\theta) \\ y_p = y + d.sin(\theta) \end{cases}$$

we derive

$$\dot{x}_p = \begin{pmatrix} \dot{x}_p \\ \dot{y}_p \end{pmatrix} = \begin{pmatrix} \dot{x} - d.sin(\theta)\dot{\theta} \\ \dot{x} + d.cos(\theta)\dot{\theta} \end{pmatrix} = \begin{pmatrix} vcos(\theta) - d.sin(\theta)w \\ vsin(\theta) + d.cos(\theta)w \end{pmatrix} = K \begin{pmatrix} v \\ w \end{pmatrix}$$

with K $= \begin{pmatrix} vcos(\theta) - d.sin(\theta)w \\ vsin(\theta) + d.cos(\theta)w \end{pmatrix}$ and $u = \begin{pmatrix} v \\ w \end{pmatrix}$ so that

$$K^{-1}\dot{x}_p = u$$

and

$$K^{-1} = \begin{pmatrix} d.cos(\theta) & d.sin(\theta) \\ -sin(\theta) & cos(\theta) \end{pmatrix} 1/d$$

- **What happens when there are calibration errors?**
  Inaccurate Movement: Calibration errors can cause the robot to move in unintended directions, making it difficult to control its movement.

  Poor Turning Performance: Differential drive robots rely on the speed difference between their two wheels to turn effectively. Calibration errors can cause one wheel to move faster or slower than the other, which can result in the robot having difficulty turning or turning too sharply.

  To address these issues K_p and d were tuned to

  $$K_p = 10 \text{ and } d = 0.2$$

  When K_p is too large (see Figure 4) we noticed fluctuations and oscillations in the wheel velocity which means the wheels speeds alternate. If $d = 0$ there is no initial error, but zeroing out d will remove the effect of θ in our formulas causing the robot to never turn and move in a straight line to infinity. Therefore, the error will increase to infinity.

- **What happens when wheel rotation velocity is saturated?**
  As seen in figure 5, the robot is simultaneously drifting on and off the right (x,y) track. This can be explained in figure 6 as the robot can be on the right x while being on the wrong y, like in point 1, or on the right y but on the wrong x (point 2). The wheel fails to catch up with the planned trajectory to the right locations at the right time and then goes to the next needed point from wherever location it is in. This is displayed in the small circle in the actual trajectory. The robot, for example, instructed to move 10° to the left every 2 seconds, will form a smaller final circle than a fast one would after a certain time.

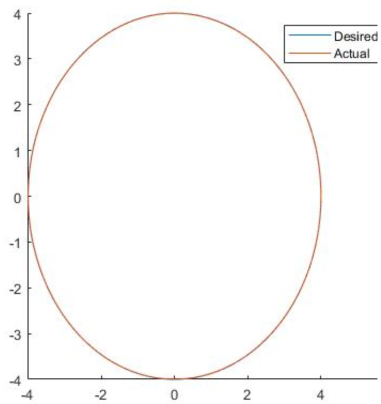


Figure 2: Static feedback path

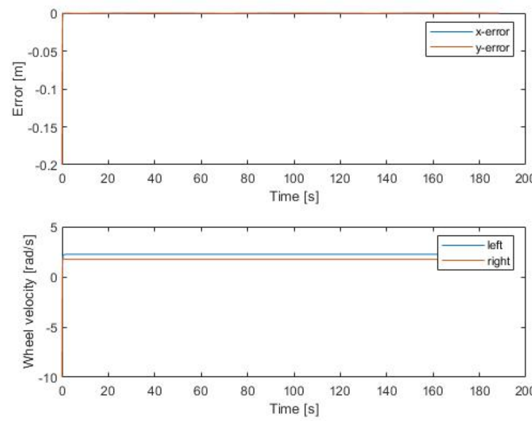


Figure 3: Static feedback error

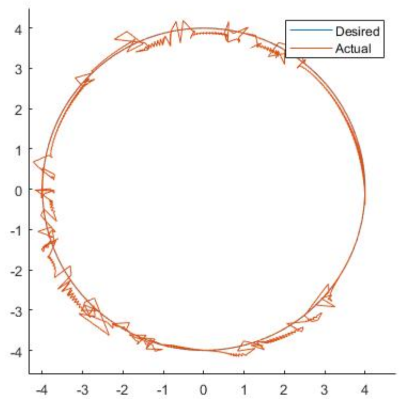


Figure 4: Static feedback with $K_p = 25$

2.2 Lyapunov-based control law

- **What happens when there are calibration errors?**
  What is noticed is that decreasing K_y results in a $0.2m/s$ decrease in the wheel velocity, resulting in a smaller circle trajectory. Increasing K_x gives us fluctuations in the wheel velocity graph. Due to high K_x the linear velocity v increases in comparison with the angular velocity w. So the robot moves forward faster than it can turn. On the other hand, increasing K_θ results in faster angular velocity relative to the linear velocity, giving a jagged trajectory.

- **What happens when wheel rotation velocity is saturated?**

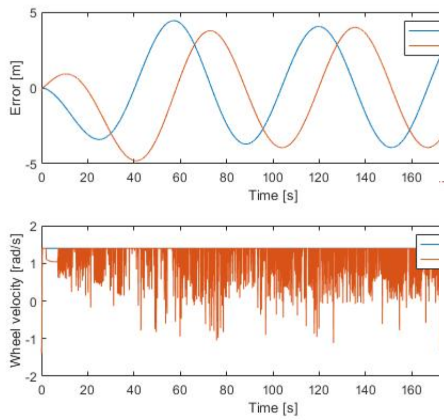


Figure 5: Static feedback error

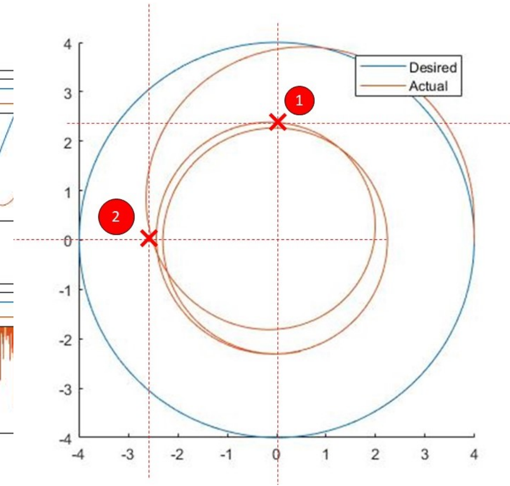


Figure 6: Static feedback path

The results are similar to the static feedback behavior.

3 Bicycle-like 1-1 robot

For

$$\mathbf{x}_p = \begin{cases} x_p = x + L.cos(\theta) + d.cos(\theta + \beta) \\ y_p = y + L.sin(\theta) + d.sin(\theta + \beta) \end{cases}$$

we derive

$$\dot{\mathbf{x}}_p = \begin{pmatrix} \dot{x}_p \\ \dot{y}_p \end{pmatrix} = \begin{pmatrix} \dot{x} - \dot{\theta} sin(\theta) L - (\dot{\theta} + \dot{\beta}) d.sin(\theta + \beta) \\ \dot{y} + \dot{\theta} L.cos(\theta) + (\dot{\theta} + \dot{\beta}) d.cos(\theta + \beta) \end{pmatrix}$$
$$= \begin{pmatrix} vcos(\theta) vcos(\theta) - vsin(\theta) sin(\beta) - (vsin(\beta)/L + u_2) d.sin(\theta + \beta) \\ vsin(\theta) cos(\beta) + vsin(\beta) cos(\theta) + (vsin(\beta)/L + u_2) d.cos(\theta + \beta) \end{pmatrix} = K \begin{pmatrix} v \\ \dot{\beta} \end{pmatrix}$$

with $K = \begin{pmatrix} cos(\theta)cos(\beta) - sin(\beta)sin(\theta) - sin(\beta)/Ld.sin(\theta+\beta)) & -d.sin(\theta+\beta) \\ sin(\theta)cos(\beta) + sin(\beta)cos(\theta) + sin(\beta)/Ld.cos(\theta+\beta)) & d.cos(\theta+\beta) \end{pmatrix}$
and $u = \begin{pmatrix} v \\ \dot{\beta} \end{pmatrix}$ so that

$$K^{-1} \dot{x}_p = u$$

3.1 Static feedback

- **What happens when there are calibration errors?**

  In a bicycle-like 1-1 robot, calibration errors in the sensors could cause the robot to incorrectly estimate its position or orientation. This could result in the robot drifting off course or not being able to move in a straight line. The robot may also have difficulty maintaining balance, which is crucial for a two-wheeled robot like this. Starting with the tuning with the values $K_p = 10$ and $d = 0.2$ as in the 2-0 robot gives us already good results as seen in Figure 8. The robot is following the desired graph with only a small error.

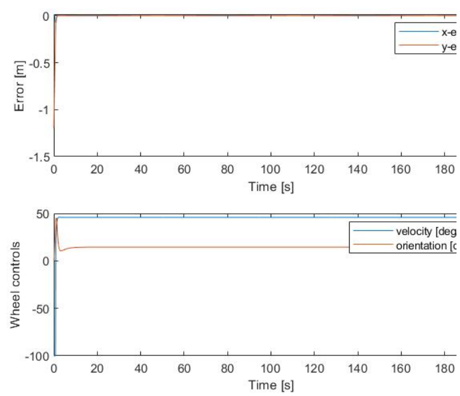


Figure 7: Static feedback error

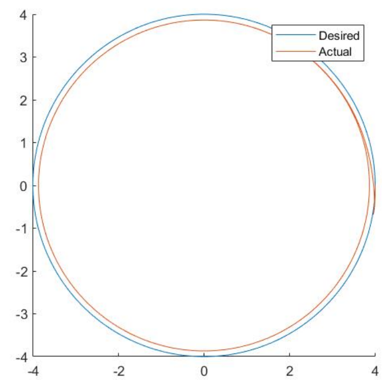


Figure 8: Static feedback path

- **What happens when wheel rotation velocity or wheel orientation is saturated?**
  If the wheel rotation velocity saturates, it means that the motor is unable to increase the speed of the wheel any further. This can cause the robot to move more slowly, or to lose traction and skid.

  Similarly, if the wheel orientation saturates, it means that the robot is unable to turn the wheel any further in a certain direction. This can cause the robot to turn less than intended, or to turn in the opposite direction.

3.2 Lyapunov-based control law

- **What happens when there are calibration errors?** If the wheel rotation velocity saturates, it means that the actual velocity of the wheel cannot match the desired velocity commanded by the control system. This can cause the robot to deviate from the desired trajectory or even become unstable. In a Lyapunov-based control system, the Lyapunov function used to analyze the stability of the system may not hold anymore, leading to the failure of the control system.

  Similarly, if the wheel orientation saturates, it means that the robot's turning radius cannot match the desired radius commanded by the control system. This can cause the robot to turn less or more than intended, leading to trajectory deviation or instability. Again, the Lyapunov function used to analyze the stability of the control system may not hold anymore, causing the control system to fail.

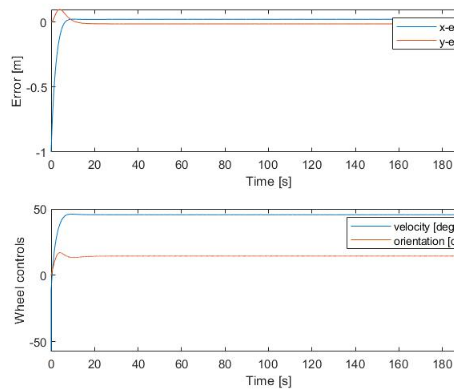


Figure 9: Lyapunov error

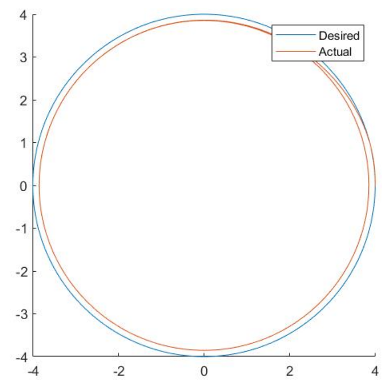


Figure 10: Lyapunov path

- **What happens when wheel rotation velocity or wheel orientation is saturated?** If the wheel rotation velocity saturates, it means that the actual velocity of the wheel cannot match the desired velocity commanded by the control system. This can lead to trajectory deviation or even instability in the system, especially in the case of a bicycle-like robot where maintaining balance is crucial. Similarly, if the wheel orientation saturates, it means that the robot's turning radius cannot match the desired radius commanded by the control system. This can also lead to trajectory deviation or instability, as the robot may turn more or less than intended.