Learning for Adaptive and Reactive Robot Control
Instructions for exercises of lecture 6

Professor: Aude Billard
Contact: aude.billard@epfl.ch

Spring Semester

1 Theoretical exercises [1h]

1.1

Book correspondence: Ex8.1, p222

Consider the nominal DS $\dot{x} = Ax$ with $A = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}$. Construct a matrix $M(x)$ that is locally active. Additionally:

1. make the entire system diverge away from the attractor (unstable)
2. make the attractor a saddle point
3. create a limit cycle around the attractor, (a) keep the system stable at the attractor or (b) make the system diverge from the attractor and converge to the limit cycle (see Fig. 1)
4. invert the direction of the initial dynamics at a fixed point x^*.
   Make sure your modulation matrix is smooth.

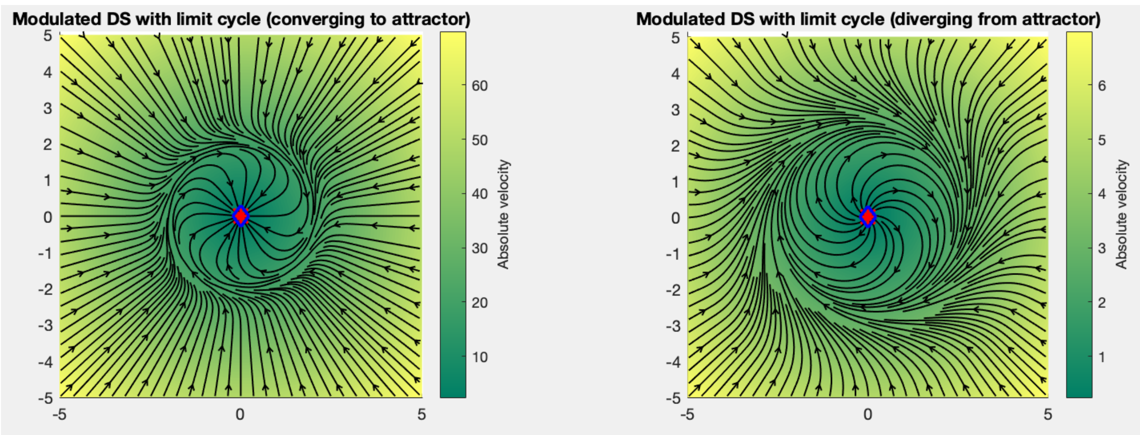

Figure 1: Two limit cycles at the attractor

1.2

Book correspondence: Ex8.2, p222

Show that if $M(x)$ is full rank for all x, the modulated dynamics has the same equilibrium point as the nominal dynamics.

1.3

(Optional) *Book correspondence: Ex8.3, p222*

Show that if the nominal dynamics is bounded and $M(x)$ is locally active in a compact-set $\chi \subset \mathbb{R}^N$. Then the modulated dynamics is bounded.

Boundedness A DS is bounded if for each $\delta > 0$, there exists $\epsilon > 0$ such that

$$\|x(t_0)\| < \delta \Rightarrow \|x(t)\| < \epsilon, \forall\, t > t_0 \tag{1}$$

1.4

(Optional)*Book correspondence: Ex8.4, p222*

Consider a system $\dot{x} = f(x)$ that has a single equilibrium point. Without loss of generality, let this equilibrium point be placed at the origin. Assume further that the equilibrium point is stable and the modulated dynamics is bounded and has the same equilibrium point as the nominal dynamics. Show that if χ does not include the origin, the modulated system is stable at the origin.

2 Programming exercises [1h]

2.1 Programming Exercise : Modulated DS

Book correspondence: p.222

Open MATLAB and set the directory to the following folder:

```
lecture6-modulating-ds/exercises
```

Open `lect6_ex1.m`. This code generates a nominal linear DS and a modulated DS as example. You will now add your implementation of two other kind of modulated DS.

1. First implement a local rotation at point $x^1 = [2, 1]^T$.
2. Then implement two local rotations in the same DS, at points $x^1 = [2, 1]^T$ and $x^2 = [3, -2]^T$.

IMPORTANT : Do not forget to uncomment the `plot_ds()` function call after the loop to plot your modulations.

Hint : You can use the examples in the code to guide your implementation.

2.2 Programming Exercise : Modulated DS

Book correspondence: p.223

Keep your directory in the following folder:

```
lecture6-modulating-ds/exercises
```

Open `lect6_ex2.m`. This file generates the nominal DS proposed in theoretical exercise 1.1. You will now code your solutions from this exercise :

1. Implement a limit cycle around the attractor.
2. Using a modulation matrix, create a locally unstable point while keeping the system stable at the origin.

IMPORTANT : Do not forget to uncomment the `plot_ds()` function call after the loop to plot your modulations.

2.3 Programming Exercise : Local Modulation

Book correspondence: Programming Ex8.3, p.226

Set the directory to the following folder:

```
lecture6-modulating-ds/exercises/Local_Modulation
```

Then run

```
modulating_dynamical_systems.m
```

This file displays a GUI in which you can generate a local rotation by first clicking on the plot, then on the 'reshape' button.

Modify the code to generate a rotation with smaller angles that varies over time. You can do so in the designated section at the end of the file.

Use the function `simulate_lmds()` to visualize your modulated DS, as shown in the code. You can modify this function call in the code or directly set your input arguments when running `modulating_dynamical_systems()` in command, as shown at the top of the file. You can also call `simulate_lmds()` in a loop to simulate a time-varying parameter. If so, do not forget to set the position of the modulation as a 7^{th} parameter to deactivate the user input in the GUI.

2.4 Programming Exercise : Learning modulations

Book correspondence: Programming Ex8.4, p.228

Set the directory to the following folder:

```
lecture6-modulating-ds/exercises/Learning_modulations
```

MATLAB file `locally_modulating_dynamical_systems.m` allows you to draw some new trajectories that will generate a local modulation. Test the effect of changing the parameters of the kernel on the precision of the reconstruction of the training trajectories. The kernel `A` can be set as an input argument when calling the function in the command window. Answer the following questions:

1. Would it be possible to have an S-shape trajectory while
   (a) the target is in the middle of the S-shape trajectory?
   (b) the target is far away from the S-shape trajectory?
2. Does the direction of the demonstrated trajectories matter?
3. What happens if the nominal DS is very stiff (large values in A) or unstable?

2.5 Programming Exercise : Externally learning modulations

Book correspondence: Programming Ex8.5, p.234

Set the directory to the following folder:

```
lecture6-modulating-ds/exercises/Externally_learning_modulations
```

The file `externally_modulating_dynamical_systems.m` allows you to define a modulation function around a specific point. The modulation is a rotation matrix, with angle modulated by $h_x(\mathbf{x})$, a RBF kernel centered at the modulation point and $h_s(\mathbf{x})$, an external signal.

$$M = \begin{bmatrix} cos(\Phi(\mathbf{x}, s)) & sin(\Phi(\mathbf{x}, s)) \\ -sin(\Phi(\mathbf{x}, s)) & cos(\Phi(\mathbf{x}, s)) \end{bmatrix} \quad \Phi(\mathbf{x}, s) = h_s(\mathbf{s}) h_x(\mathbf{x}) \phi_c \tag{2}$$

The external signal is defined as the Euclidean distance between the target and the state ($h_s = ||x - x^*||$), such that

$$
h_s = \begin{cases} 1 & ||x - x^*|| = 0 \\ 1 + \frac{r}{2} - \frac{r}{1+(\frac{r-2}{r+2})^{||x-x^*||/u}} & 0 < ||x - x^*|| \leq u \\ 0 & u < ||x - x^*|| \end{cases} \tag{3}
$$

with the parameter $r > 2$ defining how fast the external signal goes to zero and the parameter $u > 0$ defining the region of modulation around the attractor.

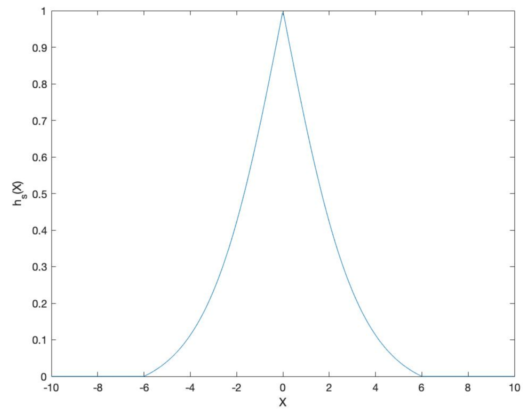

Figure 2: External signal function in 1D, with u = 6, r = 2.1, and the attractor at zero

The modulation parameters can be set as an input arguments when calling the function in the command window, as shown at the top of the file. The modulation point can be set by clicking on the plot with the mouse.
Answer the following questions:

1. What values should be set for the scalars r and u, such that the modulated function is not activated for any $x \in \mathbb{R}^2$?

2. What would happen if h_s was not a monotonically decreasing function?

3. What happens if the nominal DS is very stiff or unstable?

4. Apart from $||x - x^*||$, can you think of other external variables?

References

[1] Aude Billard, Sina Mirrazavi, and Nadia Figueroa. *Learning for Adaptive and Reactive Robot Control: A Dynamical Systems Approach.* MIT press, 2022.