Exercises for Lecture Course on Modelling and System Identification (MSI)
Albert-Ludwigs-Universität Freiburg – Winter Term 2015

Exercise 7: Nonlinear Least Squares for Output Error Minimization (OEM)

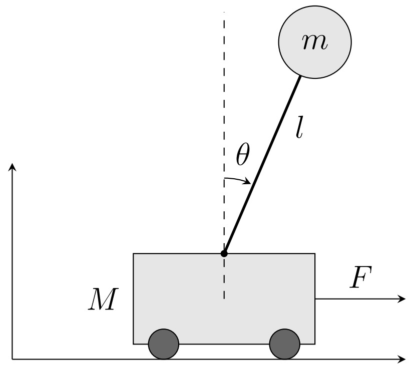
(to be returned on Jan 19, 2016, 8:15 in HS 26, or before in building 102, 1st floor, 'Anbau')

Prof. Dr. Moritz Diehl, Robin Verschueren, Jesus Lago and Fabian Girrbach

Your MATLAB solution has to run from a main script called `main.m`, which can call other functions/scripts, but when running this script all the necessary results and plots should be clearly visible.Compress the folder in a `.zip` file and send it to `jesuslagogarcia@gmail.com`. Please state also your name and the names of your team members in the e-mail.

Exercise Task

Consider a car-pendulum system as the one depicted below:



This system is controlled by an external force F applied to the car, and its dynamics are defined by three parameters: m[kg] (mass of the pendulum), l[m] (length of the pendulum) and M[kg] (mass of the car).

The position of the car is denoted by p and the pendulum configuration is described by the angle θ, using that $\theta = \pi\,\mathrm{rad}$ corresponds to the pendulum hanging down. The system dynamics are described by the following implicit ODE system

$$\begin{aligned}(M+m)\ddot{p} - ml(\ddot{\theta}\cos(\theta) - \sin(\theta)\dot{\theta}^2) - F = 0,\\ l\ddot{\theta} - \ddot{p}\cos(\theta) - g\sin(\theta) = 0.\end{aligned} \tag{1}$$

where g is the gravitational acceleration and assumed to be constant and equal to $9.81\ \mathrm{m/s^2}$. By solving for the differential state derivatives $\ddot{p}$ and $\ddot{\theta}$, one can obtain the following explicit ODE formulation which is mathematically equivalent

$$\begin{aligned}\ddot{p} &= \frac{-ml\sin(\theta)\dot{\theta}^2 + mg\cos(\theta)\sin(\theta) + F}{M+m-m(\cos(\theta))^2},\\ \ddot{\theta} &= \frac{-ml\cos(\theta)\sin(\theta)\dot{\theta}^2 + F\cos(\theta) + (M+m)g\sin(\theta)}{l(M+m-m(\cos(\theta))^2)}.\end{aligned} \tag{2}$$

1. **System simulation** In the first part of the exercise we will implement a simulation routine to calculate the response of the system.

(a) Given the system state $x = [p, \theta, \dot{p}, \dot{\theta}]^\top$, implement a function `[xdot] = carpole_ode(t,x, F, eta)` which evaluates the right-hand side of the ODE $\dot{x} = f(x, F, \eta)$, with $\eta = [M, m, l]^\top$. Use the following parameters: $M = 1$ kg, $m = 0.1$ kg and $l = 0.8$ m. Validate your function by comparing it with the given black-box function `rhs(t,x,F,eta)`, with the same function definition as `carpole_ode`. (2 point)

(b) Implement one step of an Euler integration method `[x_next] = euler_step(x0,u,deltaT, eta,@ode)`, which performs one integration step for a general ODE $\dot{x} = f_{\text{ode}}(x, u, \eta)$ starting at x_0, with input u, parameters η and a integration interval ΔT. (2 points)

(c) Load the dataset from the website. On it you will find 4 vectors: $F_\text{m}, p_\text{m}, \theta_\text{m}$ and t_m, where p_m and θ_m represent the measurements of the car and pendulum positions obtained when the system is excited with a set of inputs $F_\text{m} = [F_\text{m}(1), \ldots, F_\text{m}(N)]$ on the timegrid $t_\text{m} = [0, \ldots, (N-1)\Delta T]$. Use the implemented function `[x_next] = euler_step(x0, u, deltaT, eta, @ode)` to build a function `[x_sim] = carpole_sim(x0, F, t, eta)` which simulates the system response to a set of inputs F. Starting at $x_0 = [0, \pi, 0, 0]$, and using F_m and $\eta = [1, 0.1, 0.8]$, simulate the system and plot the simulated p_s and θ_s together with p_m and θ_m as a function of time. Does the pendulum swing up? Use the `visualize` function. (2 points)

(d) Extra: Repeat the last two tasks but using a Runge-Kutta integrator of order 4 instead of Euler. *Hint: check last section on page 50 on the script on numerical integration methods.* (2 bonus points)

2. **Parameter estimation for output error minimization** In the second part of the exercise we will use the function `lsqnonlin` of MATLAB to perform an estimation of the system parameters η.

`lsqnonlin` takes as input a vector function $f(\eta) = [f_1(\eta), \ldots, f_N(\eta)]$ with parameter η, and minimizes $\|f(\eta)\|_2^2$ with respect to η. You can find more information on: `www.mathworks.com/help/optim/ug/lsqnonlin.html`

Assuming that the car-pole system has only output errors, and that these errors are Gaussian with noise variances $\sigma_p = 0.1$ m and $\sigma_\theta = 0.2$ rad , then the Maximum Likelihood Estimation problem to estimate η is:

$$\eta^* = \arg\min_{\eta} \|[p_\text{m}^\top, \theta_\text{m}^\top]^\top - M(x_0, F_\text{m}, t_\text{m}, \eta)\|_{\Sigma^{-1}}^2 \quad (3)$$

Here, $M(x_0, F, t, \eta) = [p_\text{s}^\top, \theta_\text{s}^\top]^\top$ represents the simulated values of p and θ in a vector shape, $[p_\text{m}^\top, \theta_\text{m}^\top]^\top - M(x_0, F_\text{m}, t_\text{m}, \eta)$ are the residuals between measurements and simulation and Σ the covariance matrix of $[p_\text{m}^\top, \theta_\text{m}^\top]^\top$.

(a) Implement a function `res = residuals(eta)` which computes the residual vector between the given measurements p_m and θ_m and the simulated values p_s and θ_s obtained from `[x_sim] = carpole_sim(x0, F, t, eta)`, again with $x_0 = [0, \pi, 0, 0]$ and using `F_m2` as input, given in the dataset. (2 point)

(b) Adapt your function `residuals` in order to incorporate the measurement variances correctly, i.e. weight the cost function in the right way. (1 point)

(c) Use `lsqnonlin` to estimate η^*. (2 points)

(d) Plot the simulated model with η^* versus the measurements. (1 point)

(e) Extra: Can you find a estimate for the covariance of your estimator η^*? *Hint: linearize your residual function and use it to give an approximation of the covariance.* (2 bonus points)

This sheet gives in total 12 points and 4 bonus points