Numerical Optimal Control, August 2014

Exercise 7: Direct shooting methods

Joel Andersson   Joris Gillis   Greg Horn   Rien Quirynen   Moritz Diehl

University of Freiburg – IMTEK, August 7th, 2014

Direct single vs. direct multiple shooting

We continue to work with the simple OCP from Exercise 5 and 6:

$$\begin{array}{ll} \underset{x,u}{\text{minimize}} & \displaystyle\int_0^T x_1(t)^2 + x_2(t)^2 + u(t)^2 \, dt \\ \text{subject to} & \dot{x}_1 = (1 - x_2^2)\, x_1 - x_2 + u, \qquad x_1(0) = 0 \\ & \dot{x}_2 = x_1, \qquad x_2(0) = 1 \\ & -1 \le u(t) \le 1, \end{array} \tag{1}$$

where $T = 10$ as earlier.

Tasks:

6.1 On the course webpage as well as on Gist[1] you find a solution of this problem using direct single shooting using the now familiar RK4 integrator. Go through the script and make sure that you understand the code. Run the script.

6.2 Modify the script to so that it implements the direct multiple shooting method. The control parametrization and the definition of the integrator can remain the same. Tip: Start by replacing the line:

```
nv = N
```

with

```
nv = 1*N + 2*(N+1)
```

Make sure that you get the same solution.

6.3 Compare the IPOPT output for both scripts. How did the change from direct single shooting to direct multiple shooting influence:

- The number of iterations
- The number of nonzeros in the Jacobian of the constraints
- The number of nonzeros in the Hessian of the Lagrangian
- The total solution time

[1] https://gist.github.com/jaeandersson/2c12faa4cef5b7e99e8a

Sequential quadratic programming (SQP)

Above we solved the NLP using IPOPT, a popular primal-dual interior point code employing so-called filter line-search to ensure global convergence. Other NLP solvers that can be used from CasADi include SNOPT, WORHP and KNITRO. These three solvers implement SQP (KNITRO implements both SQP and IP). In the following, we will write our own simple SQP code to solve (1).

Gauss-Newton SQP

SQP employs a sequence of quadratic approximations to solve the NLP and solves these with a QP solver. For an NLP of the form:

$$\begin{array}{ll} \underset{x}{\text{minimize}} & f(x) \\ \text{subject to} & x_{\text{lb}} \leq x \leq x_{\text{ub}}, \quad g(x) = 0, \end{array} \tag{2}$$

these quadratic approximations take the form:

$$\begin{array}{ll} \underset{\Delta x}{\text{minimize}} & \frac{1}{2}\, \Delta x^{\text{T}}\, \nabla_x^2 \mathcal{L}(x^{(k)}, \lambda^{(k)})\, \Delta x + \nabla_x f(x^{(k)})^{\text{T}}\, \Delta x \\ \text{subject to} & x_{\text{lb}} - x^{(k)} \leq \Delta x \leq x_{\text{ub}} - x^{(k)}, \quad \dfrac{\partial g}{\partial x}(x^{(k)})\, \Delta x + g(x^{(k)}) = 0, \end{array} \tag{3}$$

where $(x^{(k)}, \lambda^{(k)})$ is a guess of the primal-dual solution to (2) and $\mathcal{L}(x, \lambda) = f(x) + \lambda^{\text{T}}\, g(x)$ is the Lagrangian. The solution of this QP gives the step in Δx and a new approximation of the multipliers λ.

Tasks:

6.4 For problems with a quadratic objective function $f(x) = \frac{1}{2}\, \|F(x)\|_2^2$, like the NLPs arising from both direct single shooting and direct multiple shooting transcription of (1), a popular variant is to use a *Gauss-Newton* approximation of the Hessian of the Lagrangian:

$$\nabla_x^2 \mathcal{L}\left(x^{(k)}, \lambda^{(k)}\right) \approx \frac{\partial F}{\partial x}\left(x^{(k)}\right)^{\text{T}} \frac{\partial F}{\partial x}\left(x^{(k)}\right) \tag{4}$$

and $\nabla_x f(x) = \frac{\partial F}{\partial x}\left(x^{(k)}\right)^{\text{T}} F\left(x^{(k)}\right)$.

What are the main advantages and disadvantages of such an approximation?

6.5 **Extra:** Implement a Gauss-Newton method to solve the problem. Use algorithmic differentiation as in Exercise 4 to calculate $\frac{\partial F}{\partial x}$ and $\frac{\partial g}{\partial x}$ and solve the QP subproblem using the `qpsolve` script from Exercise 1.