Lund, 7 December 2011

CasADi tutorial – Nonlinear programming using IPOPT

Joel Andersson   Johan Åkesson

Department of Electrical Engineering (ESAT-SCD) &
Optimization in Engineering Center (OPTEC)
Katholieke Universiteit Leuven

Yesterday

- Optimization problems: LP - QP - NLP - OCP
- Automatic/algorithmic differentiation (AD)
  - We *didn't* discuss AD implementation

    (Tricky especially for adjoint mode, complete sparse Jacobians)
  - Good news: AD works "out-of-the-box"
- More good news: Forward/adjoint sensitivity analysis for ODE/DAEs also works (almost) "out-of-the-box"
- Yet more good news: LP & QP also work (almost) "out-of-the-box"

Today

- Nonlinear programming (NLP)
  - Bad news: It doesn't quite work out of the box, always

Recall: Nonlinear programming (NLP)

$$\begin{array}{ll} \underset{x \in \mathbb{R}^N}{\text{minimize}} & f(x) \\ \text{subject to} & x_{\min} \leq x \leq x_{\max} \\ & g_{\min} \leq g(x) \leq g_{\max} \end{array} \tag{1}$$

- $x_{\min}, g_{\min} \in \mathbb{R} \cup \{-\infty\}, \quad x_{\max}, g_{\max} \in \mathbb{R} \cup \{\infty\}$
- Equality constraints: $x_{\min,k} = x_{\max,k}$ for some k
- Formulating used by NLP solvers (e.g. IPOPT)

Equivalent formulation

$$\begin{array}{ll} \underset{x \in \mathbb{R}^N}{\text{minimize}} & f(x) \\ \text{subject to} & g(x) = 0, \quad h(x) \geq 0 \end{array} \tag{2}$$

Without inequalities

$$\begin{array}{ll} \underset{x \in \mathbb{R}^N}{\text{minimize}} & f(x) \\ \text{subject to} & g(x) = 0 \end{array} \tag{3}$$

Optimality conditions

For an optimal solution x^* there exist multipliers λ^* such that:

$$\nabla_x \mathcal{L}(x^*, \lambda^*) = 0 \tag{4}$$

$$g(x^*) = 0 \tag{5}$$

where $\mathcal{L}(x, \lambda) = f(x) - \lambda^{\mathsf{T}} g(x)$ is the *Lagrangian*

How to solve nonlinear equations?

$$\nabla_x \mathcal{L}(x^*, \lambda^*) = 0 \tag{6}$$
$$g(x^*) = 0 \tag{7}$$

Newton's method

At current guess (x^k, λ^k), linearize:

$$\nabla_x \mathcal{L}(x^k, \lambda^k) + \nabla_x^2 \mathcal{L}(x^k, \lambda^k)\Delta x - \nabla_x g(x^k)\Delta\lambda = 0 \tag{8}$$
$$g(x^k) + \nabla_x g(x^k)^\mathsf{T} \Delta x = 0 \tag{9}$$

Solve for Δx and $\Delta\lambda$

Next iterate (τ scaling factor, ideally 1):

$$x^{k+1} = x^k + \tau \Delta x^k, \quad \lambda^{k+1} = \lambda^k + \tau \Delta\lambda^k \tag{10}$$

Iterate until optimality conditions satisfied

With inequalities

$$\begin{array}{ll} \underset{x \in \mathbb{R}^N}{\text{minimize}} & f(x) \\ \text{subject to} & g(x) = 0, \quad h(x) \geq 0 \end{array} \tag{11}$$

Optimality conditions: Karush-Kuhn-Tucker (KKT)

For an optimal solution x^* there exist multipliers λ^* and μ^*such that:

$$\nabla_x \mathcal{L}(x^*, \lambda^*) = 0 \tag{12}$$

$$g(x^*) = 0 \tag{13}$$

$$h(x^*) \geq 0 \tag{14}$$

$$\mu^* \geq 0 \tag{15}$$

$$h(x^*)^\mathsf{T} \mu^* = 0 \tag{16}$$

where $\mathcal{L}(x, \lambda) = f(x) - \lambda^\mathsf{T} g(x) - \mu^\mathsf{T} h(x)$ is the *Lagrangian*

Two ways to solve KKT

- Sequential quadratic programming (SQP)
  - Instead of linear system of equations, solve QP in each iteration
- Interior point methods (IP)
  - Penalize inequalities by barrier function $\tau \log(h(x))$

$$\begin{array}{ll} \underset{x \in \mathbb{R}^N}{\text{minimize}} & f(x) - \tau \log(h(x)) \\ \text{subject to} & g(x) = 0 \end{array} \qquad (17)$$

  - Solve as in equality constrained case, with $\tau \to 0$

IPOPT

- Interior Point OPTimizer
- Open-source project: `www.coin-or.org/Ipopt`, written by Carl Laird, Andreas Wächter
- Can solve very large NLPs (millions of variables/constraints)
- Linear system solved with: MA27, MA57, Mumps, Paradiso, ...
- Large & active user community: Subscribe to mailing list
- CasADi contains a complete interface
  - Calculates $\frac{\partial g}{\partial x}$, $\nabla_x \mathcal{L}$, $\nabla_x^2 \mathcal{L}$ using efficient AD

Exercise take-away

- Formulating and solving NLPs using CasADi
- IPOPT – Primal-dual interior point method
- Last exercise also hand-in problem!