Summary of TTK18

Morten Fyhn Amundsen

October 4, 2016

Contents

1 Bi-level programming in constrained control

Def: Opt prob depending on solution of another opt prob. Called “upper level” (UL) and “lower level” (LL) problems.

$$
\begin{aligned}
\min_x f_{\mathrm{UL}}(x, z) \\
G_{\mathrm{UI}}(x, z) \leq 0 \\
G_{\mathrm{UE}}(x, z) = 0 \\
z = \arg\min_z f_{\mathrm{LL}}(x, z) \\
G_{\mathrm{LI}}(x, z) \leq 0 \\
G_{\mathrm{LE}}(x, z) = 0
\end{aligned} \tag{1}
$$

Usually assume LL convex and regular. Linear constraints, linear/quadratic objective functions.

KKT for LL:

$$\mathcal{L}(x, z, \lambda, \mu) = f_{\mathrm{LL}}(x, z) + \lambda^T G_{\mathrm{LI}}(x, z) + \mu^T G_{\mathrm{LE}}(x, z) \tag{2}$$

$$
\begin{aligned}
\nabla_z \mathcal{L}(x, z, \lambda, \mu) = 0 \\
G_{\mathrm{LI}}(x, z) \leq 0 \\
G_{\mathrm{LE}}(x, z) = 0 \\
\lambda \geq 0 \\
\lambda \times G_{\mathrm{LI}}(x, z) = 0
\end{aligned} \tag{3}
$$

The overall problem is then

$$
\begin{aligned}
\min_{x,z,\lambda,\mu} \; & f_{\mathrm{UL}}(x,z) \\
G_{\mathrm{UI}}(x,z) &\leq 0 \\
G_{\mathrm{UE}}(x,z) &= 0 \\
\nabla_z \mathcal{L}(x,z,\lambda,\mu) &= 0 \\
G_{\mathrm{LI}}(x,z) &\leq 0 \\
G_{\mathrm{LE}}(x,z) &= 0 \\
\lambda &\geq 0 \\
\lambda \times G_{\mathrm{LI}}(x,z) &= 0
\end{aligned}
\tag{4}
$$

Small problems can be solved with e.g. YALMIP with the standard UL/LL formulation.

1.1 Big-M notation

Big-M formulation replaces nonlinear complementarity constraints with linear constraints using binary variables $s \in \{0,1\}$ to indicate activeness for inequality constraints:

$$
\begin{aligned}
G_{\mathrm{LI}}(x,z) &\leq 0 \\
G_{\mathrm{LI}}(x,z) &\geq -M^u(1-s) \\
\lambda &\geq 0 \\
\lambda &\leq M^\lambda s
\end{aligned}
\tag{5}
$$

This notation fulfills the complementarity constraints, and given large enough M^u and M^λ, the solution is unchanged. When $s = 1$ we get $G_{\mathrm{LI}}(x,z) \geq 0$ (inequality constraint active), and when $s = 0$ we get $\lambda = 0$ (inequality constraint inactive).

The overall problem formulation becomes

$$
\begin{aligned}
\min_{x,z,\lambda,\mu} \; & f_{\mathrm{UL}}(x,z) \\
G_{\mathrm{UI}}(x,z) &\leq 0 \\
G_{\mathrm{UE}}(x,z) &= 0 \\
\nabla_z \mathcal{L}(x,z,\lambda,\mu) &= 0 \\
G_{\mathrm{LI}}(x,z) &\leq 0 \\
G_{\mathrm{LE}}(x,z) &= 0 \\
\lambda &\geq 0 \\
G_{\mathrm{LI}}(x,z) &\geq -M^u(1-s) \\
\lambda &\leq M^\lambda s \\
s &\in \{0,1\}
\end{aligned}
\tag{6}
$$

Linear f_{ul} gives MILP, quadratic f_{ul} gives MIQP. Nonconvex, *np*-hard, but efficient software exists. Three methods make the problem easier:

- Using *branch and bound* to remove known symmetries. (Correct?)

- Restrict some combinations of binary variables.
- Prefer small M^u and M^λ. Necessary for numerical solution, which is all we have anyway. The Ms are diagonal matrices, so each constraint can be "weighted". M^λ most difficult. M^u can be found by solving an LP. May need trial and error for M^λ, and it should never constrain λ.

2 Linear matrix inequalities

Many problems in systems and control can be reduced to optimization problems involving LMIs, for which there exist efficient numerical solvers.

A strict LMI has the form

$$F(x) \triangleq F_0 + \sum_{i=1}^{m} x_i F_i > 0 \tag{7}$$

where $x \in \mathbb{R}^m$ is the variable, and $F_i = F_i^T \in \mathbb{R}^{n \times n}$, $i = 0, \ldots, m$. Multiple LMIs can be rewritten as a single LMI by stacking each matrix on the diagonal:

$$\operatorname{diag}\left(F^{(1)}(x), \ldots, F^{(p)}(x)\right) > 0 \tag{8}$$