Inexact stabilized Benders decomposition approaches with applications to MIQCQP problems

W. van Ackooij[1] and A. Frangioni[2] and W. de Oliveira[3]

[1]OSIRIS Department
EDF R&D
7 Boulevard Gaspard Monge; 9120 Palaiseau ; France

[2]Dipartimento di Informatica, Università di Pisa

[3]Universidade do Estado do Rio de Janeiro - UERJ

JFRO, CNAM, 23/06/2016

Outline

Motivation I

- In typical applications of energy management, one encounters systems of the form $Ax \leq \xi$, wherein ξ and A are subject to uncertainty.
- Typically x needs to be decided upon prior to observing uncertainty
- Knowledge of the distribution of ξ might be reasonable, but perhaps that of A is less (so).

Motivation II

- To take this into account, assume that the ith row a_i follows the model: $a_i(u) = \bar{a}_i + P_i u$,
- With $\bar{a}_i \in \mathbb{R}^n$, P_i is an $n \times n_i$ matrix, and the *uncertainty set* $u \in \mathcal{U}_i = \{u \in \mathbb{R}^{n_i} : \|u\| \leq \kappa_i\}$ is the ball of radius κ_i.
- We thus express safety of x by using the following “robust chance constraint”:

$$\mathbb{P}\big[A(u)x \leq \xi \quad \forall u \in \mathcal{U}\big] \geq p. \tag{1}$$

Motivation III

- Well established theory from robust optimization gives that this is equivalent to:

$$\mathbb{P}\left[\, \bar{a}_i^{\mathsf{T}} x + \kappa_i \left\| P_i^{\mathsf{T}} x \right\| \leq \xi_i \quad i \in I \,\right] \geq p \tag{2}$$

Motivation IV

An optimization problem involving such a robust chance constraint can be seen as a special case of problems of the type:

General setting

$$f_{\mathtt{inf}} := \min \left\{ f(x) \ : \ \mathbb{P}[\, g(x,\xi) \le 0 \,] \ge p \,,\ x \in X \right\} \tag{3}$$

- where $\xi \in \mathbb{R}^r$ is a random variable,
- $f : \mathbb{R}^n \to \mathbb{R}$ is a convex function,
- $g = [g_i]_{i \in I}$ is a mapping over a finite index set I such that each $g_i : \mathbb{R}^n \times \mathbb{R}^r \to \mathbb{R}$ is convex in the first argument,
- $X \neq \emptyset$ is a bounded convex set.

Motivation V

- ξ follows a discrete distribution with finite support, i.e., $\xi \in \{\xi_1, ..., \xi_S\}$ associated probabilities $\pi_1, ..., \pi_S$

- The constraint can be reformulated as

$$\mathbb{P}[g(x,\xi) \leq 0] \geq p \equiv \left\{ \begin{array}{ll} g_i(x,\xi^s) \leq M_i^s z_s & i \in I \,,\ s \in S \\ \sum_{i \in S} \pi_s z_s \leq 1 - p & \\ z_s \in \{0,1\} & s \in S \end{array} \right\}$$

$$\equiv \left\{ G(x) \leq Tz \,,\ z \in Z \right\}$$

- The motivating problem becomes an MIQCQP problem

Setting

- Consider the following Mixed-Integer Non-Linear Problem (MINLP)

$$\min \{ \ f(x) \ : \ G(x) \leq Tz \ , \ z \in Z \ , \ x \in X \ \} \tag{4}$$

- and for fixed $z \in Z$, the easier problem:

$$v(z) := \min \{ \ f(x) \ : \ G(x) \leq Tz \ , \ x \in X \ \} \ , \tag{5}$$

- as well as the related Benders' master problem:

$$v^* := \min \{ \ v(z) \ : \ z \in Z \ \} \ , \tag{6}$$

Structure

Lemma ([van Ackooij et al.(2016)van Ackooij, Frangioni, and de Oliveira])

The mapping v is proper, convex, and bounded from below. If for a given $z \in \mathcal{D}\mathrm{om}(v)$, the slave problem satisfies some appropriate constraint qualification (e.g., Slater's condition) so that the set $\Lambda(z) \subset \mathbb{R}^p_+$ of optimal Lagrange multipliers of the constraints $G(x) \leq Tz$ in (5) is nonempty, then $\partial v(z) = -T^\top \Lambda(z)$.

Cutting plane methods

- The obvious idea is to replace v, only implicitly known, by its cutting plane approximation:

$$\check{v}_k(z) := \max\{\, v^j + \left\langle w^j, z - z^j \right\rangle \;:\; j \in \mathcal{O}_k \,\} \;\leq\; v(z) \;, \tag{7}$$

- and solve the approximated master problem

Feasibility cuts

- Since it may be so that some $z \notin \mathcal{D}\mathrm{om}(v)$, we need to add feasibility cuts to the master problem. Here for the sake of exposition, they can take the elementary form of "no-good cuts":
- let $S(z) = \{ s \; : \; z_s = 0 \}$ and $S^k = S(z^k)$; then

$$\sum_{s \in S^k} z_s \geq 1 \tag{8}$$

is a feasibility cut that excludes the point z^k from the feasible set of the master problem.

Issues with this scheme

This simple cutting plane method suffers from the usual drawbacks

- instability: even if a current iterate z_k is close to optimal, a next one can be arbitrarily far away
- tailing off effect: slow convergence for high precision

Second, we have assumed available exact information of v, but computing $v(z)$ implies solving a (convex) NLP !

Informative on-demand inexact oracles

To remedy this last point, let us assume available a procedure that, given $z \in Z$, a *descent target* $\mathtt{tar} \in \mathbb{R} \cup \{-\infty, +\infty\}$ and a *desired accuracy* $\varepsilon \geq 0$, returns:

Informative on-demand inexact oracles

- as function information, *two* values $\underline{v}$ and $\bar{v}$ such that $\underline{v} \leq v(z) \leq \bar{v}$
- as first-order information, a vector $w \in \mathbb{R}^p$ such that

$$v(\cdot) \geq \underline{v} + \langle w, \cdot - z \rangle \quad . \tag{9}$$

  holds
- under the condition that, if $\underline{v} \leq \mathtt{tar}$ then $\bar{v} - \underline{v} \leq \varepsilon$

How to obtain such oracles

- dual approach: In several situations the problem defining v can be solved by moving to the dual. We always have a valid lower bound and linearization : eventually we will generate a primal feasible iterate (e.g., [van Ackooij and Malick(2016)])
- Primal-dual approach: The problem defining v can be solved by a primal dual method (e.g., interior points): we typically dispose of primal dual pairs, which after a while becoming feasible.

Two ideas of stabilization: trust region

- The first possibility is to consider a trust-region stabilization:

$$z^k \in \arg\min \left\{ \check{v}_k(z) \ : \ z \in Z \ , \ \left\| z - \hat{z}^k \right\|_1 \leq \mathcal{B}_k \right\}, \qquad (10)$$

- this amounts to a local branching constraint:

$$\sum_{s \,:\, \hat{z}^k_s = 1} (1 - z_s) + \sum_{s \,:\, \hat{z}^k_s = 0} z_i \leq \mathcal{B}_k \qquad (11)$$

- One can force the master problem to explore the complement of past regions

Two ideas of stabilization: level

- Disposing of a lower and upper bound on the optimal value $(v_{\mathtt{low}}, v_{\mathtt{up}})$, and level parameter $v_{\mathtt{lev}} \in (v_{\mathtt{low}}, v_{\mathtt{up}})$ we define the level set:

$$\mathbb{Z}_k := \left\{\, z \in Z \;:\; \check{v}_k(z) \leq v^k_{\mathtt{lev}} \,\right\}.$$

- We set aside a past iterate called stability center $\hat{z}^k$.
- Using a stability function $\varphi(\cdot;\hat{z}^k)$ (for instance $\varphi(\cdot;\hat{z}^k) = \left\|. - \hat{z}^k\right\|_2$) we solve

$$z^k \in \arg\min \left\{\, \varphi(z;\hat{z}^k) \;:\; z \in \mathbb{Z}_k \,\right\} \tag{12}$$

Some remarks

- The trust region master problem is a MILP (even with reverse region constraints)
- The level master (although "linear" here) is a (MI)QP in general. It can have an empty feasible set. This is important information : the level parameter is a valid lower bound for the optimal value

The trust region method I

- Initialization: Choose some initial parameters $\varepsilon_1 \geq 0$, $\gamma > 0$, $\hat{v}^1 = \infty$, $\beta > 0$, pick some $\hat{z}^1$.
- Master problem: solve the trust region master problem to produce z^k
- Stopping test: Set $\Delta_k := v_{\mathtt{up}}^{k-1} - v_{\mathtt{low}}^{k}$. If $\Delta_k > \delta$, keep trust region radius. Else ($\Delta_k < \delta$), if the trust region radius is m, then stop z^k is δ-optimal. Otherwise increase trust region.

The trust region method II

- Oracle call : Call the oracle and receive $(\underline{v}_k, \bar{v}_k, w_k)$.
  - If the slave problem is infeasible, add a feasibility cut. Else set $v^k_{\mathrm{up}} = \min\left\{v^{k-1}_{\mathrm{up}}, \bar{v}^k\right\}$ and potentially update z_{up} (the best solution)
  - If $\bar{v}_k \leq \hat{v}_k - \beta$, then set $\hat{z}^{k+1} = z^k$, $\hat{v}_{k+1} = \bar{v}_k$, choose arbitrary ε_{k+1} go back to the master problem
  - If $\underline{v}^k \leq v^k_{\mathtt{low}} + \gamma$, then choose $\varepsilon_{k+1} \in [0, \varepsilon_k)$, otherwise arbitrarily.
- return to the master problem

Some comments

- One can notice that only when the algorithm shows signs of convergence, we need to ask for any precision. Otherwise, (almost none) is requested
- One can establish finite convergence of the method to a δ-optimal solution

The Level-stabilized method I

- Initialization: Choose some initial parameters $\varepsilon_1 \geq 0$, $\gamma > 0$, $\hat{v}^1 = \infty$, $\beta > 0$, pick some $\hat{z}^1 \in \mathcal{D}\mathrm{om}(v)$.
- Stopping test: Set $\Delta_k := v_{\mathtt{up}}^{k-1} - v_{\mathtt{low}}^k$. If $\Delta_k < \delta$, then stop z^k is δ-optimal
- Master problem: Pick any $\hat{z}^k$, $v_{\mathtt{lev}} \in [v_{\mathtt{low}}^k, v_{\mathtt{up}}^k - \delta)$. Solve the level master. If it is infeasible, pick $v_{\mathtt{low}}^{k+1} \in [v_{\mathtt{lev}}^k, v^*]$ and return to master problem. Else we have produced z^k

The Level-stabilized method II

- Oracle call I: Call the oracle and receive $(\underline{v}_k, \bar{v}_k, w_k)$.
    - If the slave problem is infeasible, add a feasibility cut. Else set $v_{\mathrm{up}}^k = \min\left\{v_{\mathrm{up}}^{k-1}, \bar{v}^k\right\}$ and potentially update z_{up} (the best solution)
    - If $\underline{v}^k \leq v_{\mathtt{lev}}^k$, then choose $\varepsilon_{k+1} \in [0, \varepsilon_k)$, otherwise arbitrarily.
- Return to master problem

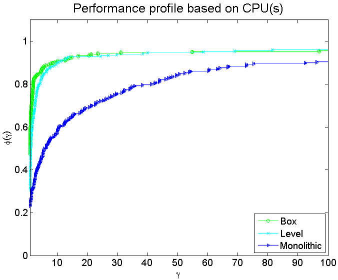
Some comments

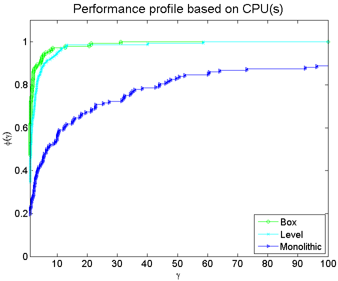
- One can notice that only when the algorithm shows signs of convergence, we need to ask for any precision. Otherwise, (almost none) is requested
- One can establish finite convergence of the method to a δ-optimal solution

Set up

- We consider instances coming from the motivating example
- We have generated several (random) instances with varying degrees of sparsity, number of rows, columns, and scenarios, yielding a total of 252 instances
- We benchmarked the methods against “monolithic” approach: Cplex

High precision $\delta = 10^{-4}$

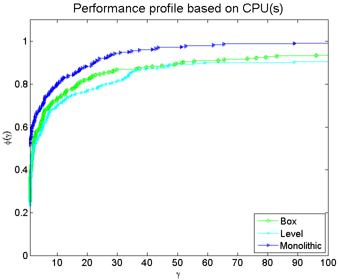


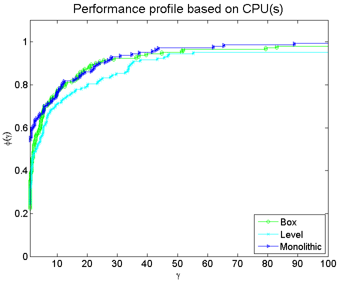


Results with all instances

Results for high sparsity instances

Lower precision $\delta = 10^{-3}$





Results with all instances

Results for high sparsity instances

Comments

- These preliminary results show the advantage of decomposition methods, even without thoroughly testing precision and target management
- The presence of sparsity seems to have a beneficial effect on the Benders type of methods.

Summary

In this talk we have discussed several generalizations of stabilized Benders decomposition methods with oracles not requiring exactly solving the subproblems. Nearly minimal requirements on handling accuracy were presented: The discussed reference is:

W. van Ackooij, A. Frangioni, and W. de Oliveira. Inexact stabilized benders' decomposition approaches: with application to chance-constrained problems with finite support.
Computational Optimization And Applications, To appear:1–24, 2016.
doi: 10.1007/s10589-016-9851-z

Bibliography I

[[van Ackooij and Malick(2016)]]W. van Ackooij and J. Malick.
Decomposition algorithm for large-scale two-stage unit-commitment.
Annals of Operations Research, 238(1):587–613, 2016.
doi: 10.1007/s10479-015-2029-8.

[[van Ackooij et al.(2016)van Ackooij, Frangioni, and de Oliveira]]W. van Ackooij, A. Frangioni, and W. de Oliveira.
Inexact stabilized benders' decomposition approaches: with application to chance-constrained problems with finite support.
Computational Optimization And Applications, To appear:1–24, 2016.
doi: 10.1007/s10589-016-9851-z.