Lecture 17 Convex Relaxations

Can Li

ChE 597: Computational Optimization
Purdue University

Overview

- Overview of global optimization algorithms
- Convex relaxations
  - McCormick envelopes
  - SDP
  - Difference of convex
  - concave functions
  - factorization
- Piecewise linear approximation and SOS2
- Outer approximation

Convex relaxations for nonconvex functions

Convex function $g(x)$ underestimate nonconvex function $f(x)$

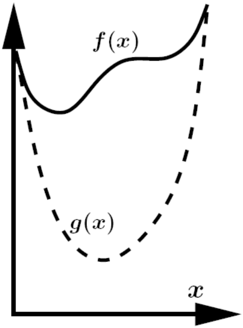


Figure: ref: Tawarmalani & Sahinidis

The convex underestimator can provide a lower bound if we are minimizing $f(x)$.

Spatial branch-and-bound

Need to perform spatial branching on the continuous variable to obtain global optimality

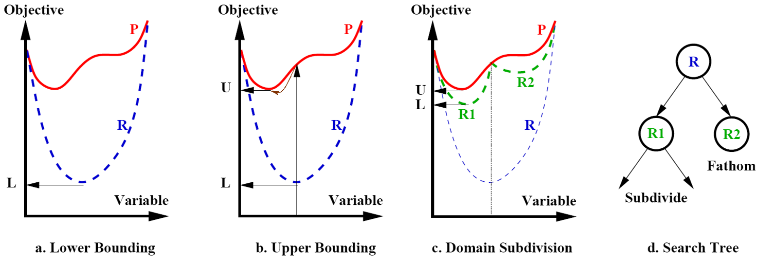
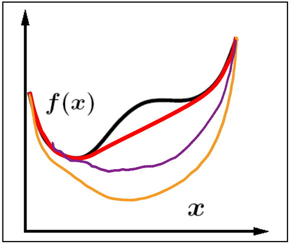


Figure: ref: Tawarmalani & Sahinidis

- Convergence in the limit: imagine each interval becomes "small" enough.
- "Tighter" convex relaxations give rise to smaller tree size.
- Key research question: how to derive tight convex relaxations

Convex Envelope



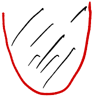


Figure: ref: Tawarmalani & Sahinidis

- Given a nonconvex function $f(x), g(x)$ is the convex envelope of $f(x)$ for $x \in \mathcal{S}$ if
  - $\underline{g}(x)$ is convex underestimator of $f(x)$
  - $\underline{g}(x) \geq \underline{h}(x)$ for all convex underestimators $\underline{h}(x)$
- The convex envelope is the tightest possible convex underestimator of a function.
- An equivalent statement is $\text{epi}(\underline{g}(x)) = \text{conv}(\text{epi}(f(x)))$. The epigraph of $\underline{g}(x)$ is the convex hull of the epigraph of $f(x)$.

$$\text{epi}(f) = \{(x, y) \in X \times \mathbb{R} \mid y \geq f(x)\}$$

Concave envelope

Figure: dotted line: concave envelope. dashed line: convex envelope. ref: Tawarmalani & Sahinidis

- Given a function $f(x), \overline{g}(x)$ is the concave envelope of $f(x)$ for $x \in \mathcal{S}$ if
  - $\overline{g}(x)$ is **concave overestimator** of $f(x)$
  - $\overline{g}(x) \geq h(x)$ for all concave overestimators $h(x)$
- The concave envelope is the tightest possible concave overestimator of a function.
- An equivalent statement is $\text{hypo}(\overline{g}(x)) = \text{conv}(\text{hypo}(f(x)))$. The hypograph of $g(x)$ is the convex hull of the hypograph of $f(x)$.

$$\text{hypo}(f) = \{(x, y) \in X \times \mathbb{R} \mid y \leq f(x)\}$$

Convex and concave envelopes of $w = xy$ (bilinear)

Consider the set

$$P = \left\{(w, x, y) | w = xy, x^L \leq x \leq x^U, y^L \leq y \leq y^U\right\}$$

- Goal: find the convex hull of P.
- Due to the definition of the convex and concave envelopes, it is equivalent to finding the convex and concave envelopes of $w(x, y) = xy$ over the domain $x^L \leq x \leq x^U, y^L \leq y \leq y^U$.

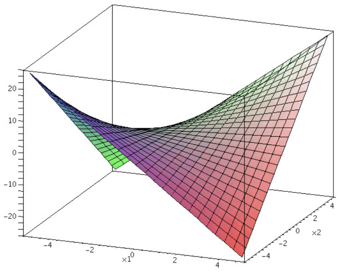


Figure: The bilinear surface $w\left(x_1, x_2\right) = x_1 x_2$. ref: Costa and Liberti

McCormick lower and upper envelopes

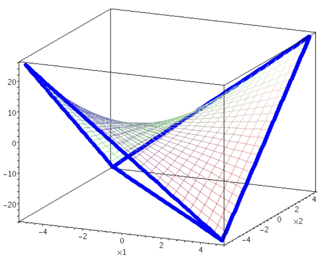


Figure: Lower convex (left) and upper concave (right) envelopes for the bilinear term. ref: Costa and Liberti

Observation: the convex and concave envelopes for bilinear function are linear.

Derivation of McCormick envelopes

$$a = \left(x - x^L\right) \;\; b = \left(y - y^L\right) \quad a \times b \geq 0$$

$$a \times b = \left(x - x^L\right)\left(y - y^L\right) = xy - x^L y - x y^L + x^L y^L \geq 0$$

$$w \geq x^L y + x y^L - x^L y^L$$

$$a = \left(x^U - x\right) b = \left(y^U - y\right) \quad a \times b \geq 0$$

$$w \geq x^U y + x y^U - x^U y^U$$

$$a = \left(x^U - x\right) b = \left(y - y^L\right) \quad a \times b \geq 0$$

$$w \leq x^U y + x y^L - x^U y^L$$

$$a = \left(x - x^L\right) b = \left(y^U - y\right) \quad a \times b \geq 0$$

$$w \leq x y^U + x^L y - x^L y^U$$

The underestimators of the function are represented by:

$$w \geq x^L y + x y^L - x^L y^L; w \geq x^U y + x y^U - x^U y^U$$

The overestimators of the function are represented by:

$$w \leq x^U y + x y^L - x^U y^L; w \leq x y^U + x^L y - x^L y^U$$

Apply McCormick envelopes to QCQPs

$$\text{(QCQP):} \quad \min \quad x^T Q_0 x + q_0^T x$$
$$\text{s.t.} \quad x^T Q_k x + q_k^T x \le b_k \quad k = 1, \ldots, K$$
$$l \le x \le u$$

$x^T Q x = \sum_{i,j} x_i x_j \cdot Q_{ij}$

$$\text{(Lifted QCQP):} \quad \min \quad Q_0 \cdot X + q_0^T x$$
$$\text{s.t.} \quad Q_k \cdot X + q_k^T x \le b_k \quad k = 1, \ldots, K$$
$$l \le x \le u$$
$$X = xx^T$$

$X_{ij} = x_i \cdot x_j \quad \forall\, i,j$

McCormick (LP) Relaxation: replace $X = xx^T$ above by applying McCormick envelope to each bilinear term $X_{ij} = x_i x_j$:

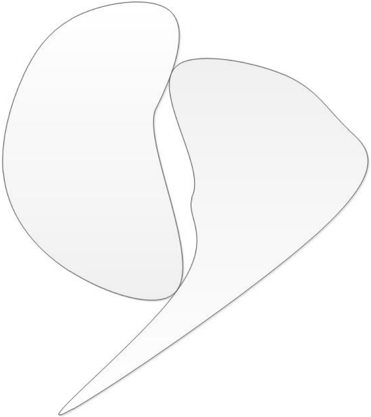
$l_i \le x_i \le u_i$

$$X_{ij} \ge l_i x_j + l_j x_i - l_i l_j$$
$$X_{ij} \ge u_i x_j + u_j x_i - u_i u_j$$
$$X_{ij} \le l_i x_j + u_j x_i - l_i u_j$$
$$X_{ij} \le u_i x_j + l_j x_i - u_i l_j$$

- Do we get the convex hull of the set $P = \left\{(x, X) | Q_k \cdot X + q_k^T x \leq b_k \; k = 1, \ldots, K; l \leq x \leq u; X = xx^T\right\}$ by applying the McCormick envelopes? In other words, do we get the convex hull of the feasible region of the QCQP?
- The answer is no in general. The reason is that the McCormick envelopes is convexifying each constraint $X_{ij} = x_i x_j$, respectively, which does not give the convex hull of the whole set.
- Intuitively, it can be seen from this result

$$\mathrm{conv}(A \cap B) \subseteq \mathrm{conv}(A) \cap \mathrm{conv}(B)$$

Geometric intuition $\mathrm{conv}(A \cap B) \subseteq \mathrm{conv}(A) \cap \mathrm{conv}(B)$



$\mathrm{conv}(A \cap B) = \emptyset$

clearly $\mathrm{conv}(A) \cap \mathrm{conv}(B) \neq \emptyset$

For convex sets A and B, $\mathrm{conv}(A \cap B) = \mathrm{conv}(A) \cap \mathrm{conv}(B)$

Applications of McCormick envelopes

- We can apply McCormick envelopes to applications discussed in the previous lecture including packing problem, continuous facility location, k-means clustering, pooling problem. Another well-studied application is AC optimal power flow problem.
- The tightness of the McCormick envelopes depends on the problem and also the formulation.
  - The McCormick relaxation of the PQ formulation is tighter than the McCormick relaxation of the P formulation and the Q formulation. That's why we prefer the PQ formulation.
  - Typically, for pooling problems you will observe that the McCormick relaxation is close to the global optimum (typically less than 5% gap).
  - For other problems like packing problem, continuous facility location, k-means clustering, AC optimal power flow problems, McCormick relaxation can be weak.

SDP relaxation for QCQP

Relax the nonconvex constraint $X = xx^T$ by the PSD constraint

$$X \succeq xx^T$$

$X - xx^T$ is PSD

Equivalent to

$$\begin{pmatrix} X & x \\ x^T & 1 \end{pmatrix} \succeq 0$$

Due to Schur's lemma.

- SDP relaxation works well for problems like AC optimal power. However, it is more expensive to solve.
- In general, SDP relaxation is incomparable to the McCormick relaxation, i.e., we cannot say one is tighter than another or vice versa.

Practical considerations of SDP relaxation

- Solving SDP relaxations using interior point solvers like Mosek can be slow. On the other hand, LP solvers are much faster and more robust.
- In QCQP solvers like Gurobi, linear cuts are generated to "outer approximate" the PSD cone.

$$v^T X v \geq 0$$

  where v is any vector in $\mathbb{R}^n$
- How to generate the "good" cuts is still an active research problem.
- Add cuts to improve the bound as much as possible while keeping the vector v sparse. Dense cuts can slow down the LP solvers.

Difference of convex (DC) relaxation

- A function $f(x)$ is d.c. (difference of convex functions) if there exist convex functions $p(x)$ and $q(x)$ such that $f(x) = \underbrace{p(x)}_{\text{convex}} \boxed{-\, q(x)}$ → convex underestimator of $-q(x)$,
- An underestimator of $f(x)$ is $p(x) + Q(x)$, where $Q(x)$ is an underestimator of the concave function $-q(x)$
- One possible d.c. decomposition of $f(x_1, x_2, \ldots, x_n)$ is

$$f = \underbrace{f + \mu \sum_i x_i^2}_{p(x)} - \mu \sum_i x_i^2 \qquad H\Big(\sum_{i=1} x_i^2\Big) = \begin{bmatrix} 2 & & & \\ & 2 & & \\ & & 2 & \\ & & & \ddots \end{bmatrix}$$

  for a sufficiently large value of μ for which the eigenvalues of the Hessian of the first two terms of the sum become positive. where $p(x) = f = f + \mu \sum_i x_i^2 \quad q(x) = \mu \sum_i x_i^2$
  $q(x)$ can be relaxed by its McCormick envelopes.
- An example of calculating the μ is the αBB algorithm (Floudas et al.).

Applications of DC programming to QCQP

- Uniform perturbation of Q: For $f(x) = x^T Qx + q^T x$

$$f(x) = x^T Qx + q^T x + \mu \sum_i x_i^2 - \mu \sum_i x_i^2$$

We can let $\mu = -\lambda_{\min}(Q)$.

- Nonuniform perturbation of Q

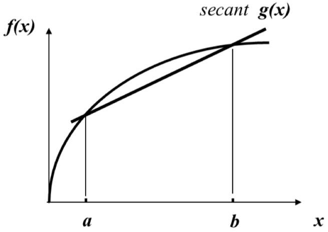
$$\begin{bmatrix} \alpha_1 & & & \\ & \alpha_2 & & \\ & & \ddots & \\ & & & \alpha_n \end{bmatrix}$$

$$\begin{aligned} f(x) &= x^T Qx + q^T x + x^T \mathrm{Diag}(\alpha)x - x^T \mathrm{Diag}(\alpha)x \\ &= x^T (Q + \mathrm{Diag}(\alpha))x + q^T x - x^T \mathrm{Diag}(\alpha)x \end{aligned}$$

where Diag(α)) is a matrix with $\alpha \in \mathbb{R}^n$ in the diagonal. Choose α such that $Q + \mathrm{Diag}(\alpha) \succeq 0$.

Convex envelope of univariate concave function

- Secant underestimator



$$g(x) = f(a) + \frac{[f(b) - f(a)]}{b - a}(x - a)$$

- This is applicable to functions including
  - $\log(x)$
  - x^α, $0 < \alpha < 1$.

Procedure for bounding factorable programs

Introduce variables for intermediate quantities whose envelopes are not known

Example $f(x, y, z, w) = \sqrt{\exp(xy + z \ln w) z^3}$

$$\overbrace{\Big(\underbrace{\overbrace{\exp(\underbrace{\underbrace{xy}_{x_1} + \overbrace{z \underbrace{\ln w}_{x_2}}^{x_3}}_{x_4})}^{x_5} \overbrace{z^3}^{x_6}}_{x^7}\Big)^{0.5}}^{f}$$

$$x_1 = xy$$
$$x_2 = \ln(w)$$
$$x_3 = z x_2$$
$$x_4 = x_1 + x_3$$
$$x_5 = \exp(x_4)$$
$$x_6 = z^3$$
$$x_7 = x_5 x_6$$
$$f = \sqrt{x_7}$$

Figure: ref:Tawarmalani & Sahinidis

Factor multilinear and polynomial functions

- Multi-linear function

$x_1 x_2 x_3 x_5$

$$M(x_1, \ldots, x_n) = \sum a_t \prod_{i=1}^{p_t} x_i, L_i \leq x_i \leq U_i, i = 1, \ldots, n$$

- Polynomial functions

$x_1^2 x_3^3 x_5^5$

$$P(x_1, \ldots, x_n) = \sum a_t \prod_{i=1}^{p_t} x_i^{\alpha_i}, L_i \leq x_i \leq U_i, i = 1, \ldots, n$$

where $\alpha_i \in \mathbb{Z}^+$

- Example $z = x_1^2 x_2 x_3$ can be factored as

$$y_1 = x_1 x_2, y_2 = y_1 x_1, z = y_2 x_3$$

- Factorization is not unique.
- Practical implication: any polynomial optimization problem can be converted to QCQP and solved using Gurobi.

Recall: piecewise linear approximation of a nonconvex function

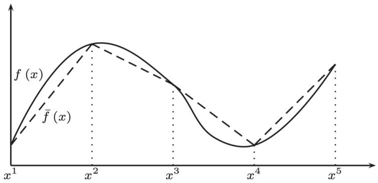


Figure: ref: Wolsey

1. Practical implication: for functions that are complicated but have low dimension, one can use piecewise linear approximation. The advantage is that we can make use of the MILP solvers.

SOS2

Definition

A set of variables of which at most two can be positive. If two are positive, they must be adjacent in the set.

- Typically modeled using special ordered sets of type 2.
- The adjacency conditions of SOS2 are enforced by the solution algorithm.
- All commercial solvers allow you to specify SOS2 constraints.

$$\bar{f}(x) = \sum_{i=1}^{k} \lambda_i f(x^i)$$

$$\sum_{i=1}^{k} \lambda_i = 1$$

$$x = \sum_{i=1}^{k} \lambda_i x^i$$

$$\lambda_i \geq 0 \quad \forall i$$

Outer approximation (OA) of convex nonlinear function

- Motivation: interior point solvers for convex NLPs are not as robust as the LP solvers.
- In solvers like Gurobi and BARON, OA cuts are generated convex nonlinear functions.

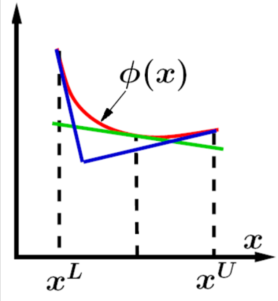


Figure: Outer approximate (underestimate) the convex nonlinear functions by linear cuts. ref: Tawarmalani, & Sahinidis

Where to add the cuts?

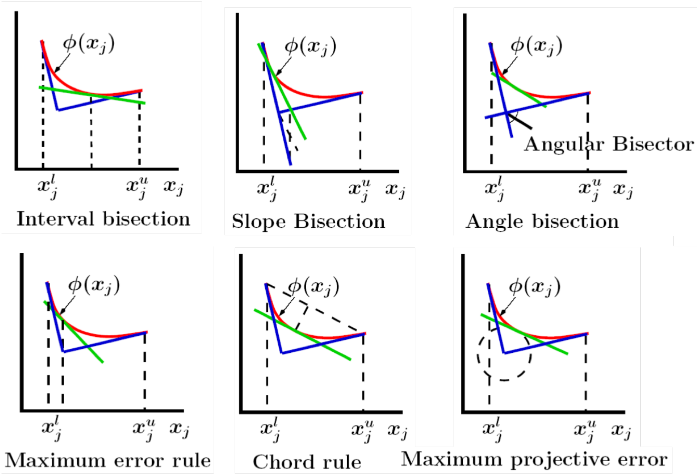


Figure: Heuristics for generating OA cuts. ref: Tawarmalani, & Sahinidis

Automatic detection of convexity

- Global solvers like BARON can automatically detect convex functions and apply OA cuts.
- These detection algorithms reply on identifying univariate convex functions and composition rules that preserve convexity (recall lecture 2).
    - **Nonnegative linear combination:** $f_1, \ldots, f_m$ convex implies $a_1 f_1 + \ldots + a_m f_m$ convex for any $a_1, \ldots, a_m \geq 0$
    - **Pointwise maximization:** if f_s is convex for any $s \in S$, then $f(x) = \max_{s \in S} f_s(x)$ is convex. Note that the set S here (number of functions f_s) can be infinite
    - **Partial minimization:** if $g(x, y)$ is convex in x, y, and C is convex, then $f(x) = \min_{y \in C} g(x, y)$ is convex
    - **Affine composition**: if f is convex, then $g(x) = f(Ax + b)$ is convex.

References

- Tawarmalani, M., & Sahinidis, N. V. (2013). Convexification and global optimization in continuous and mixed-integer nonlinear programming: theory, algorithms, software, and applications (Vol. 65). Springer Science & Business Media.
- Horst, R., & Tuy, H. (2013). Global optimization: Deterministic approaches. Springer Science & Business Media.