Tightening McCormick Relaxations Via Reformulation of Intermediate Functions into Schema

Matthew Wilhelm, Robert Ernst, Matthew Stuber

2018 AIChE Annual Meeting
Pittsburgh, PA, October 30th

Process Systems and Operations Research Laboratory

UCONN SCHOOL OF ENGINEERING

Outline

Background

Classification Algorithm

Numerical Results

Motivation

Process Systems and Operations Research Laboratory

- Simulation is a primary task for chemical engineers. Generally, optimization problems involving simulation have the following special form[1]:

$$f^* = \min_{\mathbf{p} \in P} f(\mathbf{x}(\mathbf{p}), \mathbf{p}) \quad (1)$$
$$\text{s.t. } \mathbf{g}(\mathbf{x}(\mathbf{p}), \mathbf{p}) \leq \mathbf{0}$$

- **Kinetic parameter estimation:** solve a system ODEs for fixed parameters then calculate sum-square-error [1]
- **Engineering design:** Engineering tasks naturally lead to optimization problems

1 Stuber, M. et al. Optimization Methods and Software, 2015, 30, 424-460

Factorable Programming

- Nonconvex problems are typically solved via variations on branch-and -bound algorithms [2].
- Generally we need to solve the relaxed problem to construct the lower bound

$$f^{LBD} = \min_{\mathbf{p}\in P} f^{cv}(\hat{\mathbf{x}}(\mathbf{p}), \mathbf{p})$$
$$\text{s.t. } \mathbf{g}^{cv}(\hat{\mathbf{x}}(\mathbf{p}), \mathbf{p}) \leq \mathbf{0}$$

Example:

$$f^{LBD} = \min_{\mathbf{p}\in P} (p \times \exp(-p))^3 + \exp(-p)$$

Factorable decomposition:

$$f^{LBD} = \min_{\mathbf{z}\in Z} z_6$$

$$\begin{aligned} z_1 &= p & z_4 &= z_1 z_3 \\ z_2 &= -z_1 & z_5 &= z_4^3 \\ z_3 &= \exp(z_2) & z_6 &= z_5 + z_3 \end{aligned}$$

Each constraint in the model is then relaxed using function in library.

[2] Horst, R. & Toy. H. Global Optimization: Deterministic Approaches, 2013

Relaxations using Set-Valued Mappings

Process Systems and Operations Research Laboratory

- Auxiliary variable methods (AVM) introduce a significant number of variables and constraints not present in original problem [3].
- Set-valued mappings can be used to progressively build valid bounds as an alternative without introducing additional variables [4].

Example:

$$f^{LBD} = \min_{\mathbf{p}\in P} (p \times \exp(-p))^3 + \exp(-p)$$

Factorable decomposition:

$$f^{LBD} = \min_{\mathbf{p}\in P} z_6(\mathbf{p})^{cv}$$

$$z_1^{cv/cc} \leftarrow p^{cc/cv} \qquad z_4 \leftarrow (z_1 z_3)^{cv/cc}$$

$$z_2^{cv/cc} \leftarrow (-z_1)^{cv/cc} \qquad z_5 \leftarrow (z_4^3)^{cv/cc}$$

$$z_3^{cv/cc} \leftarrow \exp(z_2)^{cv/cc} \qquad z_6 \leftarrow (z_5 + z_3)^{cv/cc}$$

3 Horst, R. & Toy. H. Global Optimization: Deterministic Approaches, 2013

4 Mitsos, A. et al. SIAM Journal of Optimization, 2009, 20, 573-601

McCormick Composition Rule [5]:

Let $Z \subset \mathbb{R}^n$, $X \subset \mathbb{R}$ be nonempty convex. The composite function $g = \phi \circ f$ s.t. $f : Z \to \mathbb{R}$ is continuous, $F : X \to \mathbb{R}$, $f(Z) \subset X$. Let $f^{cv}, f^{cc} : Z \to \mathbb{R}$ be relaxations of f on Z. Let $\phi^{cv}, \phi^{cc} : X \to \mathbb{R}$ be relaxations of ϕ on X. Let $\xi^*_{\text{min}}/\xi^*_{\text{max}}$ be a min/max of ϕ^{cv}/ϕ^{cc} on X.

$$g^{cv} : Z \to \mathbb{R} : z \mapsto \phi^{cv}(\text{mid}(f^{cv}, f^{cc}, \xi^*_{\text{min}}))$$
$$g^{cc} : Z \to \mathbb{R} : z \mapsto \phi^{cc}(\text{mid}(f^{cv}, f^{cc}, \xi^*_{\text{max}}))$$

- McCormick relaxation approach for algorithm introduced in [5] provides a series of composition rules expressing inequalities in form of bounding functions.
- Usually second-order convergent and tighter than interval bounds [5,6].
- Desirable to minimize clustering in branch-and-bound algorithm [7].

5 Mitsos, A. et al. SIAM Journal of Optimization, 2009, 20, 573-601

6 Bompadre, A. et al. Journal of Global Optimization, 2012, 52, 1-28

7 Kannan, R. et al. Journal of Global Optimization, 2017, 69, 629-676

Prior Work - Global Optimization

Process Systems and Operations Research Laboratory

McCormick Operator Methods:

- Multiplication, Addition, and Maximization [5,8]
- Concavoconvex and Convexoconcave Relaxations [5]
- Reverse Operator Propagation [9]

Auxiliary Variable Methods:

- Incorporation of Convexity Detection (DCP) [10]
- Convex Transformable Intermediate Expressions [11]
- Multilinear and Constrained McCormick Relaxations [12]

5 Mitsos, A. et al. SIAM Journal of Optimization, 2009, 20, 573-601

8 Khan, K. et al. J Glob Optim, 2017, 67, 687-729

9 Wechsung, A. J Glob Optim, 2015, 63 (1), 1-36

10 Khajavirad, A. et al. Math Prog Computation, 2018, 10 (3), 383-421

11 Khajavirad, A. et al. Math Prog, 2014, 1-2, 107-140

12 Bao, X. et al. Math Prog Computation, 2015, 7(1), 1-37

DAG Manipulation in Global Optimization

Process Systems and Operations Research Laboratory

- One of the most common uses for directed graph manipulation is during presolve for convexity detection and cut generation [13].
- These techniques commonly require bounds on variables and the first four rearrangements reduce the directed graph to a single operator which envelopes may be known.

$$\log(a^x) = x\log(a) \tag{2}$$

$$a^{\log(x)} = x^{\log(a)} \tag{3}$$

$$(a^x)^b = (a^b)^x \tag{4}$$

$$(x^a)^b = x^{ab} \tag{5}$$

$$(x_1,\ldots,x_n)^a = x_1^a,\ldots,x_n^a \tag{6}$$

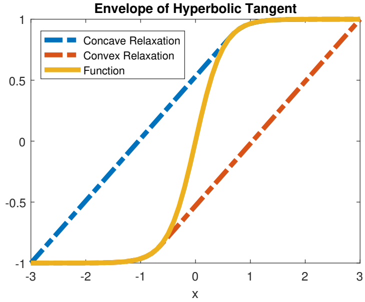
13 Khajavirad, A. et al. Math Prog Comp, 2018, 10, 383-421

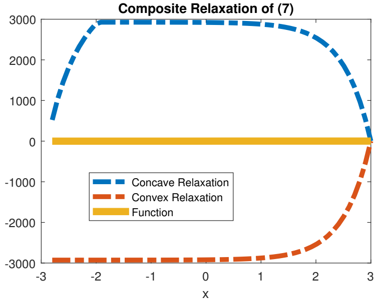
Outline

Subexpression Tightening

- Common univariate subexpressions that appear in modeling: Polynomials, $x\log(x)$, sigmoid functions, solutions to ODEs, series approximations
- Known envelopes are often tighter than composite relaxations. Consider:

$$f(x) = \frac{(\exp(3x) - \exp(-3x))}{(\exp(3x) + \exp(-3x))} \tag{7}$$





14 Schweidtmann, AM., Mitsos Journal of Optimization Theory and Applications (2018) 1-24.

Extending Set-Valued Mapping with DAG Approaches

Process Systems and Operations Research Laboratory

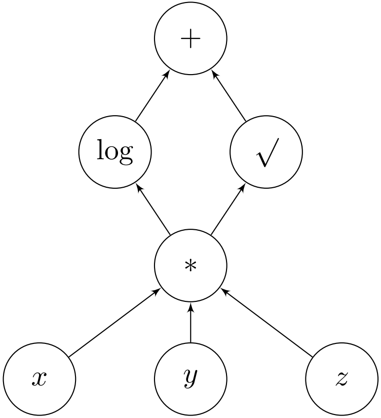
- McCormick operators embed many disciplined convex programming rules (DCP) while calculating relaxations[15].
- Currently, no simple logical rules exist for identifying concavoconvex and convexoconcave subexpressions.
- The principle purpose of this work is to provide a method for identifying these subexpressions and minimizing the number of composition steps needed in these calculations.

15 Shen, X. et al. IEEE 55th Conference on Decision and Control (CDC), 2016, 1009-1014

Directed Acyclic Graph of Functions

Consider the function below

$f(x, y, z) = \log(xyz) + \sqrt{xyz}$



Factorable Function [16]

A function F is *factorable* if it can be expressed in terms of a finite number of factors $v_1, \ldots, v_m$, s.t. given $\imath \in S$, $v_i = p_i$ for $i = 1, \ldots, n_p$, and v_k is defined for $n_p \leq k \leq m$ as either

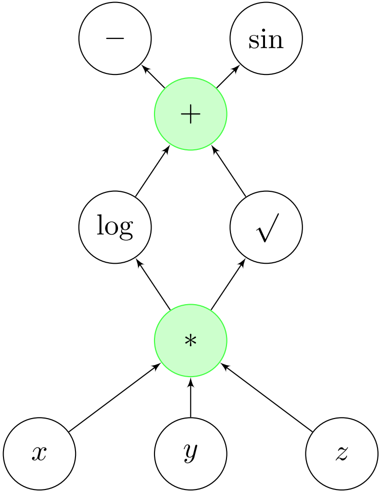
1. $v_k = v_i + v_j$, with, $i, j < k$
2. $v_k = v_i v_j$, with, $i, j < k$
3. $v_k = U_k(v_i)$, with, $i < k$, where $U_k : X \rightarrow R$ is a univariate intrinsic function

and $F(\mathbf{p}) = v_m$.

16 Scott, J. et al. J. Global Optim, 2011, 51, p569-606

Directed Acyclic Graph of Functions

Process Systems and Operations Research Laboratory



DAG of a Function [16]

The graph has the representation, G = (V,E,f), were $v \in V$, $e \in E$ is a two-tuple of (v_1, v_2), mappings $f_s, f_t : E \to V \times V$ such that $\forall e \in E$, we have $f_s(e) \neq f_t(e)$.

Articulation Point or Bicut of a Graph[17]

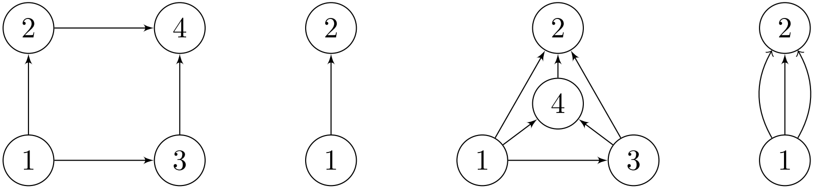
A pair $v_1, v_2 \in V \times V$ is a bicut of a directed graph G = (V,E,F) iff $G - v_1, v_2$ has more components than G, $G - v_1$, or $G - v_2$.

16 Vu, X. et al., J. Global Optim, 2005, 33(4), p541-562

17 Tarjan, R. et al., SIAM J. Comput. 1985, 14(4): p862-874

SQPR Tree Representation [18]

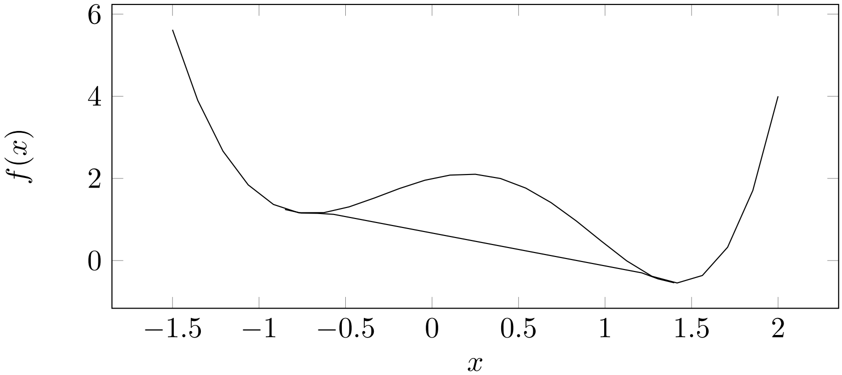
- All bicuts can be computed in linear time using well-known algorithms.
- The DAG is represented by a tree that linear subgraphs containing: (S) three or greater cycle graphs, (Q) dipole graphs, (P) single edge graphs, or (R) any other triconnected graph.
- Enumerating all tricuts or higher cuts poses significantly more difficulty.



18 Tarjan, R. et al., SIAM J. Comput. 1985, 14(4): p862-874

Classification via Root Finding

- Identify envelopes by finding the number of roots of the second derivative to identify convex and concave regions.
- We sequential then sequentially find points connecting supporting line segments and the original function.



19 Maranas, C.D. et al., J Global Optim, 1995, 7, 143-182.

Interval Root-Finding [20]

Process Systems and Operations Research Laboratory

Interval Newton Iteration

Let $\tilde{x} \in X \in \mathbb{IR}$ the interval Newton operator is defined by $N(X,\tilde{x}) = \tilde{x} - F'(X)^{-1}F(\tilde{x})$. Then let $X' = X \bigcap N(X,\tilde{x})$.

Existence and Uniqueness of Roots

- Every zero $x^* \in X$ of F is in X'.
- If $X' = \emptyset$ then F contains no zero in X.
- If $\tilde{x} \in \text{int}(X)$ and $X' \subset X$ then F contains a unique zero.

Analogous methods exist for providing validated bounds on nonsmooth functions.

20 Moore, R.E. et al., Methods and Applications of Interval Analysis, 1979

Outline

Identifying Candidate Nodes

1. Set maximum number of roots to check for, k, and maximum number of child nodes to search.
2. Preprocess graph to eliminate tertiary and higher-arity functions.
3. Identify all bicut verticies.
4. For each parent bicut vertex, create a list A of all child bicut vertices in order of path length.
5. Starting with the parent vertex, check for a reformulation with fewer than k roots. If no reformulation exists, perform a binary search of all child nodes. Terminate with highest child found.

Benchmarking Setup

Process Systems and Operations Research Laboratory

- For all cases in the GLOBAL/BCP benchmarking library, bicut identification was performed in under 0.01 seconds (quadratic terms omitted).
- We selected ten problems from the GLOBAL/BCP and literature libraries to compare McCormick relaxation tightening method purposed with the branch-and-bound routines in EAGO.
- Only problems with at least one identified bicut were considered.
- An absolute tolerance of 10^{-4} was selected as the termination criteria. Ran single threaded on a 3.60GHz Intel Xeon E3-1270 v5 processor with 32GB in Ubuntu 16.04LTS and Julia v1.0. The affine lower bounding problem was solved using CPLEX 12.8.0. Ipopt v3.12 [21] was used to solve the NLP upper-bound problem.

21 Wächter, A. et al., Mathematical Programming, 2006, 106(1), p25-57

Benchmarking Results

Problem	Variables	Inequalities	Equalities	CPU[s] (EAGO)	CPU[s] (+Algo)
ex4_1_7	1	0	0	1.0	0.2
ex6_2_10	6	0	3	95.2	37.1
growthls	3	0	0	5.1	5.1
filter	2	0	1	0.6	0.6
hydro	30	0	25	0.9	0.9
hs62	3	0	1	4.5	4.5
st_ph1	6	5	0	0.1	0.07
tre	2	0	0	0.15	0.07
heat[5]	1	0	0	1.2	0.5
CS I [12]	2	0	9	0.7	0.3
CS II [12]	5	12	1	60.7	31.4

5 Mitsos, A. et al. SIAM Journal of Optimization, 2009, 20, 573-601

12 Bongartz, D. et al. Journal of Global Optimization, 2017, 20, 761-796

Future Work & Conclusions

- For the explored problems, an average speed up of approximately 40 percent was obtained.
- Future work, will focus on developing simple checks for implicitly-defined functions.
- Development of multivariate approaches presents some challenges:
  - Unknown if linear time classification of n-cut vertices with $n > 2$ exists.
  - Need to develop specialized approximations of envelopes described by convex programs [22,23].

22 Khajavirad, A. et al., Sahinidis, N.V. J Glob. Optim. 52, 391-409 (2012)

23 Khajavirad, A. et al., Sahinidis, N.V. Math. Program. 137, 371-408 (2013)

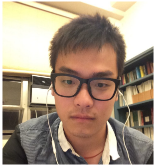
Acknowledgements

Process Systems and Operations Research Laboratory

UCONN
SCHOOL OF ENGINEERING

- Thanks to the members of the PSOR lab for discussion.
- Thanks to the University of Connecticut for providing funding for this research.

Prof. Matthew Stuber



Chenyu Wang

William Hale

Connor Dion

Jacob Chicano

Abiha Jafri

Questions?