The Convex Hull of Function Vectors

M. Ballerstein[1]**, D. Michaels**[2]**, and R. Weismantel**[3]

Abstract: A challenging task in designing global optimization algorithms is to construct tight convex relaxations that provide reasonably globally valid bounds on a mixed-integer nonlinear program (MINLP). For a general MINLP, convex relaxations are usually obtained by replacing each non-linearity occuring in the model description by a convex under- and a concave overestimator. The mathematical object studied to derive such estimators is given by the convex hull of the graph of a function $f : \mathbb{R}^n \to \mathbb{R}$ restricted to a relevant domain $D \subseteq \mathbb{R}^n$, i. e. the convex set $\mathrm{conv}\big(\{(x, f(x)) \in \mathbb{R}^{n+1} \mid x \in D\}\big)$.

The separate consideration of the non-linearities ignores all interactions between different non-linearities through the convexification step. To derive improved relaxations, we investigate, for a set of functions $f_i : D \subseteq \mathbb{R}^n \to \mathbb{R}$, $i \in I$, the convex hull of the graph of the vector-valued function $f : D \subseteq \mathbb{R}^n \to \mathbb{R}^{|I|}$ given by the function vector $f(x) := \big(f_1(x), ..., f_{|I|}(x)\big)$, called the convex hull of f. An important example is given by the convex hull of the function vector consisting of all quadratic monomials in a set of variables. In this work, we discuss useful properties of convex hulls of function vectors in a more general setting. In particular, we establish a link to a set of convex hulls associated with a certain family of real-valued functions. This link is used to define improved relaxations. We especially focus on some well-structured classes of function vectors of low dimension. Numerical examples are presented demonstrating the impact of this concept.

[1,2,3] Eidgenössische Technische Hochschule Zürich
Institut für Operations Research
Rämistrasse 101, 8092 Zürich, Switzerland
{*martin.ballerstein, dennis.michaels, robert.weismantel*}*@ifor.math.ethz.ch*