

## 12d Integrating Cutting Planes into Solution Methods for Non-Linear Disjunctive Programming Problems

*Nicolas W. Sawaya and Ignacio E. Grossmann*

Mixed Integer Non-Linear Programming (MINLP) and Generalized Disjunctive Programming (GDP) are common frameworks used to model non-linear continuous/discrete optimization problems. While the MINLP model is based entirely on algebraic equations and inequalities, the GDP model, which has been proposed by Raman and Grossmann (1994), allows a combination of algebraic and logical equations through disjunctions and logic propositions in terms of Boolean and continuous variables to be used in order to model these problems. In this presentation, we discuss the theoretical underpinnings of the cutting planes developed by Sawaya and Grossmann (2005), compare them to the lift-and-project cuts developed by Balas, Ceria and Cornuejols (1993), and address their incorporation into various solution methods for GDP problems in order to strengthen their relaxation without greatly increasing the problem size.

Different methods currently exist to solve MINLP problems, amongst which non-linear Branch and Bound (B&B), Outer- Approximation (OA), Generalized Benders Decomposition (GBD) and the Extended Cutting Plane method (ECP) are the most popular. For purposes of time however, we choose to focus our attention in this presentation on the B&B and OA frameworks. These methods can be implemented in their traditional forms if the GDP problem is converted into an MINLP problem through a traditional big-M transformation or by using the transformation developed by Lee and Grossmann (2001) that is based on determining the convex hull of each disjunction, or in their logical-equivalent forms if the original structure of the GDP problem is maintained.

The traditional B&B algorithm proceeds by relaxing the MINLP and subsequently branching on the binary variables until an optimal integer solution is obtained. In contrast, the disjunctive B&B algorithm (its logical equivalent) as promulgated by Beaumont (1990) and Lee and Grossmann (2001) proceeds by branching directly on the disjunctions in the GDP by fixing the Boolean variables in exactly one term of every disjunction to a value of True. The remaining non-selected terms in every disjunction are converted into a relaxed MINLP by using either the big-M relaxation (in Beaumont) or the convex hull relaxation (in Lee and Grossmann). The OA algorithm is an iterative decomposition scheme that cycles between a set of NLP problems and a linear master MIP problem in its traditional form, or a linear master GDP problem in its logical-equivalent form as delineated by Turkay and Grossmann (1996). The master problem is solved using traditional B&B in the former case, and disjunctive B&B in the latter case.

In all of the above methods, a full or partial conversion from GDP into MIP format using the aforementioned big-M or convex hull transformations is required at some point in the algorithm. While the convex hull reformulation will often produce a significantly tighter lower bound when compared with the traditional big-M reformulation, the limitation of this method is that it requires the introduction of new disaggregated variables and additional constraints that can greatly increase the size of the problem. In order to circumvent the issues of the large problem size in the convex hull reformulation and the looseness of the relaxation in the big-M reformulation, we propose the strengthening of the latter through cutting planes that are generated from the continuous convex hull relaxation. To accomplish this objective we generalize the cutting plane methodology developed by Sawaya and Grossmann (2005) to the non-linear case and apply it to the root node of both the traditional and disjunctive B&B algorithms, as well as at the level of the master problem in both the traditional and logic OA.

The proposed cutting plane method relies on converting the (full or partial) GDP problem into an equivalent big-M reformulation that is successively strengthened by convex hull cuts generated from a

separation problem whose feasible region corresponds to that of the convex hull reformulation. The separation problem yields a point within the convex hull relaxation that lies at minimum distance from the solution point of the big-M problem, which is then used to generate a cut that is added to the big-M problem. This sequence of separation problems is repeatedly solved, either until the optimal feasible integer solution of the original problem in the case of a B&B schema or of the specific master problem in the case of an OA schema is found, or else until there is no improvement within a specified tolerance, in which case one switches to the B&B algorithm or back to the OA algorithm, respectively.

It is interesting to note that the proposed cuts, though related to a certain extent to some of the work done on lift-and-project cutting planes originally developed by Balas, Ceria and Cornuejols (1993), are different in important ways from the latter. The major distinction lies in the derivation and generation of our cuts, which are crucially based on a GDP formulation, as opposed to a MIP formulation, as in the case of lift-and-project cuts. Furthermore, if we were to contextualize their work within a GDP framework, lift-and-project cuts would be generated through a sequential convexification procedure that would obtain by taking the convex hull of one disjunction at a time, as opposed to our proposed method which generates cuts based on a formulation that considers the intersection of the convex hulls of every disjunction.

We discuss the above differences between the cuts and illustrate the application and performance of the proposed schemes for incorporating cutting plane techniques in all aforementioned algorithms with several test problems that include process networks, safety and layout problems. Comparisons with existing methods are also given to identify the cases for which the incorporation of cutting planes is the most effective. Finally, we discuss the possible use of the cutting planes as a way for deciding whether to reformulate disjunctions as big-M or convex hull constraints.