Comput Optim Appl (2017) 68:473–478
DOI 10.1007/s10589-017-9959-9

COAP 2016 Best Paper prize

© Springer Science+Business Media, LLC 2017

Each year, the editorial board of Computational Optimization and Applications (COAP) selects a paper from the preceding year's publications for the Best Paper Award. In 2016, 97 papers were published by COAP. The recipients of the 2016 Best Paper Award are Pietro Belotti (FICO Xpress), Pierre Bonami (IBM-CPLEX), Matteo Fischetti (University of Padova), Andrea Lodi (École Polytechnique de Montréal), Michele Monaci (University of Bologna), Amaya Nogales-Gómez (Huawei Paris), and Domenico Salvagnin (University of Padova) for their paper "On handling indicator constraints in mixed integer programming" published in volume 65, pages 545–566. This article highlights the research related to the award winning paper.

The paper [2] addresses possible ways to handle disjunctive constraints, i.e., constraints that either hold or are relaxed depending on the value of a binary variable. Typically, these conditions are modelled by using indicator constraints, i.e., by associating a valid large coefficient, ubiquitously called *big-M*, for each binary variable that controls the disjunction, and by deriving the associated indicator constraint. This method allows one to deal with nonlinearities and logical implications staying within the Mixed Integer Linear Programming (MILP) framework, and can be used for Mixed Integer Quadratic Programming (MIQP) as well. Actually, this formulation is the easiest (and commonly used) disjunctive programming approach (see, e.g., [1,6] and [7]) to deal with indicator constraints and disjunctions in general, and has been widely studied in the literature, both from theoretical and from computational viewpoints (see, e.g., [4]). Unfortunately, it is well known that the Linear Programming relaxation of this formulation may be extremely weak, as the binary variable that controls the constraint can take a very small fractional value and still make the constraint satisfied. This makes branch-and-cut algorithms extremely inefficient, and the solution of the models extremely hard in practice.

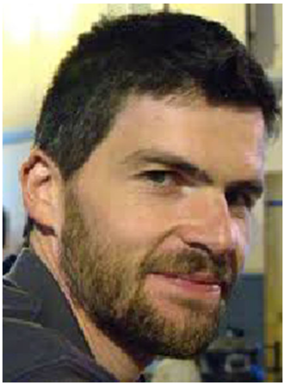
An alternative way to handle disjunctive constraints is to resort to Mixed Integer Nonlinear Programming (MINLP), replacing indicator constraints with suitable nonconvex constraints. The presence of this nonconvexity might not significantly increase the complexity of the resulting model (see, e.g., some applications in Chemical Engineering [7]). However, in the cases where those logical constraints were the only sources of nonconvexity, the common approach has always been that of using indicator constraints and MILP techniques.

In this paper, the authors consider a relevant class of convex MIQP problems arising in *supervised classification* (see, e.g., [9]). In particular, *Support Vector Machines* (SVM) methods have proven to be one of the state-of-the-art methods for supervised classification, calling for a hyperplane that partitions a given set of objects in the d-dimensional space. The objective function, to be minimized, is typically defined by the so-called *ramp loss* function (see [8]). These problems have a great relevance in real-world applications and represent a pivotal application in the area of Machine Learning. A preliminary set of computational experiments showed that, even using state-of-the-art commercial MILP solvers, most of the 2-dimensional instances proposed in the literature [5] were unsolvable in practice. On the contrary, a Global Optimization (GO) solver was able to solve all instances but one within a reasonable computing time. This bad behavior of MILP solvers was also experienced after changing some internal parameters (including the algorithm to solve the relaxation at each branch-and-bound node), and showed that the GO solver is intrinsically more suitable for solving this class of problems.

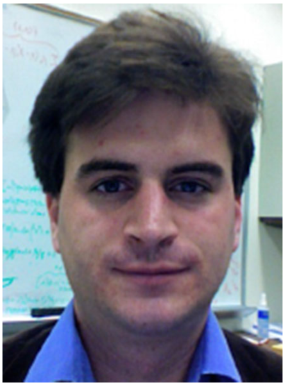
Analyzing the main differences between the two solvers, it turned out that bound reduction had a considerable impact on the performances of the GO solver. As a matter of fact, any GO solver adopts, at each branch-and-bound node, a fast bound reduction procedure to eliminate portions of the feasible region, while guaranteeing that at least one optimal solution is retained. The authors showed that adding a similar ingredient to MILP-based solvers could improve the performances of the resulting algorithms. Indeed, applying an aggressive tightening of the bounds of some variables as a preprocessing step produced an effective MILP-based solver that was able to solve all instances to proven optimality within at most 90 s. Inspired by these results, the bound tightening procedure has also been adopted in one commercial MILP solver (namely, IBM-ILOG Cplex since version 12.6.1). IBM-ILOG Cplex implements this procedure for general indicator constraints, adding the so-called *local implied bound cuts* at each branch-and-bound node, thus going towards a real integration of aggressive bound tightening in MILP. An additional set of computational experiments compared the performances of IBM-ILOG Cplex with and without these cuts, showing that a significant improvement in the performances can be obtained.

In summary, the paper gives three main contributions:

1. It helps understanding the relationship between MILP and MINLP by focusing on the use of important ingredients from MINLP into MILP; this is partially in contrast with the general direction of using MILP ingredients into MINLP.
2. It provides a practical way of solving hard MILP formulations involving disjunctive constraints—a commonly used tool in Mathematical Optimization.
3. It provides one of the first references of successful use of sophisticated Discrete Optimization approaches for solving hard Machine Learning problems, a research area in huge expansion under the pioneering influence of Bertsimas (see, e.g., [3]).



Pietro Belotti earned a PhD in 2003 from the Technical University of Milan. He is a developer with the FICO Xpress Optimizer team since 2013. Prior to FICO, he worked at Carnegie Mellon University, Lehigh University, and Clemson University.



Pierre Bonami graduated in 2003 from Université Paris 6. He is working at IBM in the research and development team of the CPLEX Optimizer and on-leave from a permanent research position in CNRS (France). Previously, he has been a post-doctoral fellow at Carnegie Mellon University and then IBM research. He is the project manager of the open source solver Bonmin.

Matteo Fischetti graduated from University of Bologna in 1987. Since 1997 he is full professor of Operations Research at the University of Padua. He won, among others, the following international prizes: (1) Best Ph.D. Dissertation on Transportation, awarded by the Operations Research Society of America in 1987; (2) the INFORMS Edelman award in 2008; and (3) the Harold Larnder Prize awarded by the Canadian OR society in 2015.

Andrea Lodi is, since 2015, Canada Excellence Research Chair in "Data Science for Real-time Decision-Making" at the École Polytechnique de Montréal. He graduated from the University of Bologna in 2000, he has been Herman Goldstine Fellow at the IBM Mathematical Sciences Department, NY in 2005–2006 and full professor of Operations Research at DEI, University of Bologna in 2007–2015.

Michele Monaci is, since 2016, Associate Professor of Operations Research at the University of Bologna, where he graduated in 1998 and got a Ph.D. in System Engineering in 2002. He has been Assistant Professor at University of Padova from 2005 to 2016.

Amaya Nogales-Gómez received her B.Sc. in Mathematics in 2010 and an M.Sc. in Advanced Mathematics in 2012 from the University of Seville. She graduated with a Ph.D. in Mathematics from the University of Seville in 2015. She is currently working as a researcher in the Paris Research Center, Huawei France.

Domenico Salvagnin is Assistant Professor at the University of Padova since 2011. He also collaborates with the IBM CPLEX Optimizer research and development team since 2014. His research activity is in computational mixed integer linear programming.

References

1. Balas, E.: Disjunctive programming. Ann. Discrete Math. **5**, 3–51 (1979)
2. Belotti, P., Bonami, P., Fischetti, M., Lodi, A., Monaci, M., Nogales-Gómez, A., Salvagnin, D.: On handling indicator constraints in mixed integer programming. Comput. Optim. Appl. **65**(3), 545–566 (2016)
3. Bertsimas, D.: Statistics and machine learning via a modern optimization lens. INFORMS, The 2014–2015 Philip McCord Morse Lecture, pp. 1–42 (2014)
4. Bonami, P., Lodi, A., Tramontani, A., Wiese, S.: On mathematical programming with indicator constraints. Math. Program. **151**(1), 191–223 (2015)
5. Brooks, J.: Support vector machines with the ramp loss and the hard margin loss. Oper. Res. **59**(2), 467–479 (2011)
6. Ceria, S., Soares, J.: Convex programming for disjunctive convex optimization. Math. Program. **86**(3), 595–614 (1999)
7. Grossmann, I.E., Trespalacios, F.: Systematic modeling of discrete-continuous optimization models through generalized disjunctive programming. AIChE J. **59**(9), 3276–3295 (2013)
8. Shen, X., Tseng, G., Zhang, X., Wong, W.: On ψ-learning. J. Am. Stat. Assoc. **98**(463), 724–734 (2003)
9. Wu, X., Kumar, V., Ross Quinlan, J., Ghosh, J., Yang, Q., Motoda, H., McLachlan, G., Ng, A., Liu, B., Yu, P., Zhou, Z.-H., Steinbach, M., Hand, D., Steinberg, D., Steinberg, D.: Top 10 algorithms in data mining. Knowl. Inf. Syst. **14**(1), 1–37 (2007)