Integer Program Reformulation for Robust Branch-and-Cut-and-Price Algorithms

Marcus Poggi de Aragão [1], Eduardo Uchoa [2]

[1] Departamento de Informática, PUC-Rio, poggi@inf.puc-rio.br

[2] Dep. de Engenharia de Produção, Universidade Federal Fluminense, uchoa@producao.uff.br

November 14th, 2003

Abstract

Since cut and column generation were established as two of the most important techniques in integer programming, researchers have looked for ways of combining them into a robust branch-and-cut-and-price algorithm. Here, "robust" means that neither branching nor the addition of cuts should change the structure of the pricing subproblems. In the last few years, several researchers independently noted that cuts expressed in terms of variables from a suitable original formulation could be added to the master problem without disturbing the pricing. This fact is still little known outside the "column generation community" and its consequences on integer programming are just beginning to be explored. This work intends to be an analysis of how to reformulate an integer program in order to build an efficient robust branch-and-cut-and-price. In particular, we propose an alternative master problem that can be quite advantageous in some situations. Another key issue addressed is how to avoid the pitfalls that arise from variable symmetries in the original formulations of many problems. We refer to extensive computational experiments on the capacitated vehicle routing, capacitated minimum spanning tree, and generalized assignment problems. Remarkable results on benchmark instances from the literature clearly attest the power of combining cut and column generation.

1 Introduction

Branch-and-Cut-and-Price (BCP) algorithms are branch-and-bound algorithms where the dual bounds are obtained by solving a linear program with exponentially many rows and columns. Therefore, cut generation (separation) and column generation (pricing) must be performed. We say that such algorithm is *robust* when the structures of the separation and pricing subproblems remain unchanged during its execution. The first BCP appears to be the one by Nemhauser and Park [9] on the edge-coloring problem. That algorithm was non-robust, the new cuts added to the master problem complicate the pricing subproblem, so it eventually has to be solved by costly IP techniques. Non-robust BCP algorithms are not likely to be effective on most cases.

In the late 90's, several researchers independently noted that cuts expressed in terms of variables from the original formulation could be dynamically separated, translated and added to the master problem. Those cuts do not change the structure of the pricing subproblem. Such observation allowed the construction of true robust BCP algorithms [12, 7, 8, 11, 3, 1].

This work gives two original contributions to the design of robust BCP algorithms.

1. We introduce an alternative to the traditional Dantzig-Wolfe master problem for column generation. We call it *Explicit Master*. A nice feature of the Explicit Master is that it makes the robust BCP mechanism quite evident. We prove that the Dantzig-Wolfe Master is equivalent to the Explicit Master plus the restriction that the reduced costs of all original variables must be zero. The algorithmic implication is that by working with the Explicit Master one may be able to fix variables cheaply. The Explicit Master is also a convenient device for performing what we call *multiple column generation*.

2. Symmetry on the variables from the original formulation is a serious drawback that prevents the construction of effective robust BCP algorithms on many important problems, such as bin-packing, cutting stock, graph coloring, several clustering and scheduling problems, among others. We propose a reformulation scheme to avoid those symmetries, extending the applicability of robust BCP algorithms to virtually any combinatorial optimization problem.

Perhaps a third important contribution is simply to point out the vast potential impact that BCP algorithms represent in the practice of integer programming. Up to now, BCP was only applied to (a few) problems where a pure branch-and-price (BP) already performed well. We argue that the applicability of this technique goes far beyond that class of problems. This claim is supported by computational experiments.

We take our results on the classical Capacitated Vehicle Routing Problem (CVRP) as a prototypical example. Pure BP algorithms are quite weak on the CVRP. The best results for this problem had been obtained by sophisticated branch-and-cut (BC) algorithms. But many instances from the literature in the range of 50–100 vertices, including some proposed more than 30 years ago, appeared to be far from being solved. A robust BCP for that problem led to a breakthrough, consistently solving instances with up to 100 vertices [5].

2 Dantzig-Wolfe Master × Explicit Master

Let (O) be the following integer program with n variables

$$(O) \qquad \begin{array}{llcl} Z_{IP} = \min & cx & & \\ s.t. & Ax & = & b \\ & Dx & \leq & d \\ & x & \in & Z_+^n \end{array}$$

We assume that $\mathcal{Q} = \{x \in Z_+^n | Dx \leq d\}$ is a *finite* set with elements $x^1, \ldots, x^p$. Let Q be a $n \times p$ matrix where each column corresponds to one element of $\mathcal{Q}$. There is a one-to-one correspondence between elements of $\mathcal{Q}$ and solutions of

$$\begin{array}{ll} x = & Q\lambda \\ s.t. & \mathbf{1}\lambda = 1 \\ & \lambda \in \{0,1\}^p \end{array}$$

The traditional IP reformulation of (O) consists in replacing x with the expression above [6]. Relaxing the integrality constraints, we get the *Dantzig-Wolfe Master* LP.

$$(DWM) \qquad \begin{array}{llcl} Z_{DWM} = \min & (cQ)\lambda & & \\ s.t. & (AQ)\lambda & = & b \\ & \mathbf{1}\lambda & = & 1 \\ & \lambda & \geq & 0 \end{array}$$

The lower bound Z_{DWM} may improve over the one obtained with the linear relaxation of (O), for $Z_{DWM} = \min cx \; s.t. \; Ax = b, \, x \in \text{Conv}(Dx \leq d, \, x \in Z_+^n)$. Because p is very large, *(DWM)* has to be solved by column generation. The pricing subproblem is the following IP.

$$(PS) \qquad \begin{array}{lrcl} v(\mu,\nu) = \min & (c - \mu A)x & - & \nu \\ s.t. & Dx & \leq & d \\ & x & \in & Z_+^n, \end{array}$$

where μ and ν are the dual variables associated with the rows of *(DWM)*. The column generation scheme is only practical when constraints $Dx \leq d$ have a "nice structure", which allows many pricing subproblems to be solved in reasonable time. It is essential to keep the pricing subproblem tractable after cutting (or branching).

Let $\overline{\lambda}$ be a fractional solution of *(DWM)*. Define $\overline{x} = Q\overline{\lambda}$. Separate a valid cut $a^i x \leq b_i$ such that $a^i \overline{x} > b_i$. The constraint $(a^i Q)\lambda \leq b_i$ can be added to *(DWM)*, just as if $a^i x \leq b_i$ belonged to the original $Ax = b$ constraints in (O). The pricing subproblem will continue to be (PS), the only difference being the extra row of A and the extra element of μ that have to be considered in the calculation of $(c - \mu A)$.

Consider again the original problem (O). Introduce additional variables x', defined to be equal to x, as follows.

$$(O') \qquad \begin{array}{lrrcl} Z_{IP} = \min & & cx & & \\ s.t. & x' & -x & = & 0 \\ & & Ax & = & b \\ & Dx' & & \leq & d \\ & x' & & \in & Z_+^n \\ & & x & \in & Z_+^n \end{array}$$

Replace x' by its equivalent expression in terms of Q. Dropping the integrality we get the *Explicit Master*

LP.

$$
(EM) \qquad \begin{array}{llrrcl}
Z_{EM} = \min & & & cx & & \\
s.t. & & Q\lambda & -x & = & 0 \\
 & & \mathbf{1}\lambda & & = & 1 \\
 & & & Ax & = & b \\
 & & \lambda, & x & \geq & 0
\end{array}
$$

Let π, ν and μ be the dual variables associated to the three sets of constraints in (EM). The columns corresponding to λ variables are priced by solving the following IP.

$$
(PS') \qquad \begin{array}{lrcl}
v(\pi,\nu) = \min & -\pi x & - & \nu \\
s.t. & Dx & \leq & d \\
 & x & \in & Z_+^n
\end{array}
$$

Variables μ do not appear in the subproblem. It is clear that separating cuts over x and adding them to $Ax = b$ does not change the pricing structure.

As expected, $Z_{DWM} = Z_{EM}$. The size of *(DWM)* is smaller than the size of (EM). What are the potential advantages of constructing a robust BCP over (EM) ?

2.1 Fixing by Reduced Costs

After solving *(DWM)*, one can try to fix x variables by the so-called Lagrangean Probing. Each attempt to fix a single variable requires the solution of an IP similar to (PS). This can be much time consuming on many cases. On the other hand, after solving (EM), x variables can be fixed by reduced costs for free. The following lemma explains that difference between *(DWM)* and (EM).

Lemma 1 *Solving (DWM) is equivalent to solving (EM) with the dual space restricted by constraints $\bar{c} = c - \mu A + \pi = 0$.*

The fixing of variables by reduced costs in (EM) can be much improved in practice by a dual perturbation that favors dual solutions leading to large reduced costs.

2.2 Multiple Column Generation

Suppose we have the following IP, where $Dx \leq d$ and $Fx \leq f$ are both "nicely structured" sets of constraints, but $(Dx \leq d, Fx \leq f)$ is not.

$$(O) \qquad \begin{array}{llcl} \min & cx & & \\ s.t. & Ax & = & b \\ & Dx & \leq & d \\ & Fx & \leq & f \\ & x & \in & Z_+^n \end{array}$$

Let Q_1 and Q_2 be the matrices where columns correspond to integer solutions in $\{x \in Z_+^n | Dx \leq d\}$ and $\{x \in Z_+^n | Fx \leq f\}$ respectively. The Explicit Master is a convenient device to perform multiple column generation on Q_1 and Q_2, as shown bellow.

$$(EM) \qquad \begin{array}{llllcl} \min & & & cx & & \\ s.t. & Q_1\lambda_1 & & -x & = & 0 \\ & \mathbf{1}\lambda_1 & & & = & 1 \\ & & Q_2\lambda_2 & -x & = & 0 \\ & & \mathbf{1}\lambda_2 & & = & 1 \\ & & & Ax & = & b \\ & \lambda_1, & \lambda_2, & x & \geq & 0 \end{array}$$

3 Reformulating Symmetric Problems

On many important problems where BP is usually applied, the corresponding original formulations are highly symmetric. This happens on problems like bin-packing, cutting stock, graph coloring, several clustering and scheduling problems, and many others. Most of those problems share the same characteristic: one searches for an optimal feasible assignment of elements to sets, and some of those sets are indistinguishable. The original formulations corresponding to their usual column generation may have variables like x_i^l, meaning that element i is assigned to the set numbered l. Cutting (or branching) over those symmetric variables can be completely ineffective.

We propose a remedy to allow effective robust BCP algorithms on those problems. An automatic reformulation translates an original formulation over x_i^l variables to a new one over variables x_{ij} meaning that items i and j are assigned to the same set and that items k, $i < k < j$, are not assigned to that set. Cuts over the x_{ij} variables can be very effective. The side effect of the reformulation is that the pricing subproblems get a little more difficult, but remains unchanged after cutting or branching, as required.

We exemplify on the Bin-Packing Problem (BPP). Given a set of items numbered as $\{1, \ldots, n\}$, positive integer weights $d(1), \ldots, d(n)$ and a bin capacity C; the BPP is to determine how to pack the items into the minimum number of bins. Introduce a dummy item 0. Define the set V as $\{0, \ldots, n\}$ and V_+ as $\{1, \ldots, n\}$ Given a set $S \subseteq V_+$, let $d(S)$ be the sum of the weights of all items in S. Define a directed graph $G = (V, A)$, where $A = \{(i, j) \,|\, i \in V, j \in V, i < j\}$. For each arc in A define a binary variable x_{ij}. The reformulation on x_{ij} variables follows:

$$
\begin{array}{llllll}
\min & \sum_{(0,j)\in\delta^+(0)} x_{0j} & & & & (1)\\
& \sum_{(i,j)\in\delta^-(j)} x_{ij} & = & 1 & \forall j \in V_+ & (2)\\
& \sum_{(i,j)\in\delta^+(i)} x_{ij} & \leq & 1 & \forall i \in V_+ & (3)\\
& \sum_{(i,j)\in\delta(S)} x_{ij} & \geq & \lceil d(S)/C \rceil & \forall S \subseteq V_+ & (4)\\
& x_{ij} & \in & \{0,1\} & \forall (i,j) \in A & (5)
\end{array}
$$

Constraints (2-3) and (5) together requires x to define in G a collection of cycles and directed paths from vertex 0. Constraints (4) forbids cycles and directed paths where total vertex weights exceeds C. Note that this formulation is completely asymmetric, each solution corresponds to a different packing.

Define Q as the set of 0-1 vectors that define directed paths from vertex 0 not exceeding total vertex weight C. Define $Ax = b$ as the set of constraints (2). It can be shown that

$$
\begin{array}{llrcl}
 & Z_{DWM} = \min & \mathbf{1}\lambda & & \\
(DWM) & s.t. & (AQ)\lambda & = & b \\
 & & \lambda & \geq & 0
\end{array}
,
$$

yields exactly the same bound obtainable by solving the traditional column generation for the BPP. But now, branching or cutting over the x variables (using inequalities (4), for instance) is effective and robust. The pricing problem is not a knapsack problem anymore, but a capacitated shortest-path problem over an acyclic graph. It can still be solved by efficient pseudo-polynomial algorithms.

The same reformulation can be applied to the graph coloring problem. In that case the set Q corresponds to directed paths in G from vertex 0 and only traversing vertices corresponding to independent sets in the graph that is being colored. The resulting pricing is strongly NP-hard, but tractable for graphs with up to a few hundreds vertices. Cuts corresponding to odd holes, anti-odd holes and other structures can be added to obtain high quality bounds.

The automatic reformulation can be extended to more complex symmetric problems, where the original integer variables are like x_i^l, but represent the number of copies of item i assigned to the set numbered l. This is also true for variables x_i^{ol}, the number of copies of item i assigned to the set of type o numbered l. This last case covers several important problems, like cutting stock with different stock sizes, scheduling on heterogeneous machines, etc. Again, the resulting reformulations can be proved to be completely asymmetric.

4 Computational Results

We are currently implementing robust BCP algorithms for many classical problems. We briefly cite some results already published.

- Capacitated Vehicle Routing Problem [5]. As already mentioned in the introduction, a robust BCP over the Dantizig-Wolfe Master yielded excellent results on the CVRP.
- Capacitated Spanning Tree Problem [4]. Since the pricing subproblem here is expensive, we implemented the robust BCP over the Explicit Master to allow fixing by reduced costs. The resulting algorithm obtained a breakthrough, being the only one able to solve the instances with non-unitary demands from the OR-Library with 50 and 100 vertices.
- Generalized Assignment Problem [10]. A robust BCP over the Dantzig-Wolfe Master could solve 3 out of the 5 remaining open instances from the OR-Library.

5 Perspectives

BCP algorithms are still rare in the literature of today. The new theoretical insights about robust BCP and the good computational results that are being obtained by such algorithms on important problems indicate that this situation may soon change. This certainly raises many interesting questions and research topics.

- Obtaining high-performance from a BC or from a BP requires experience to choose among several possible strategies (and often requires some computational tricks too). BCP multiplies the number of possibilities. For example, should one price to optimality before cutting ? Or cutting is better done along with (or even before) pricing ?
- For obvious reasons, polyhedral research concentrates on polytopes associated to problems where pure BC works. Robust BCP motivates new polyhedral investigations on other problems. For example, what are the families of facets of the BPP polytope over the x_{ij} variables ?
- Which families of cuts yield effective cuts in a BCP context ? The point is that even families of facets can be already implicitly given by the column generation. A result of this kind was recently obtained by A. Letchford[1], proving that generalized multistar inequalities for the CVRP are useless in a BCP similar to that described in [5].
- In a similar way, robust BCP motivates devising new combinatorial relaxations/decompositions to problems, even if they do not yield a good pure BP algorithm. Take the Travelling Salesperson Problem (TSP) as example. The 1-tree relaxation is not interesting on a BCP because it can be shown that fractional solutions satisfying the degree and subtour elimination constraints can be represented as convex combinations of 1-trees and those subtour cuts can be separated in polynomial time. But what about state-space relaxations like those proposed in [2] ? And the closed walk of size n avoiding subcycles of size less than k and visiting exactly once the vertices of a given set S relaxation ? It is possible in a BCP to restrict x to be a convex combination of such walks by polynomial-time pricing (for fixed k and $|S|$). Can we do the same using known TSP cuts separated in polynomial time ?

References

[1] C. Barnhart, C. Hane, and P. Vance. Using branch-and-price-and-cut to solve origin-destination integer multicommodity flow problems. *Operations Research*, 40:318–326, 2000.

[1]Personal communication.

[2] N. Christofides, A. Mingozzi, and P. Toth. State-space relaxation procedures for the computation of bounds to routing problems. *Networks*, 11:145–164, 1981.

[3] G. Felici, C. Gentile, and G. Rinaldi. Solving large MIP models in supply chain management by branch & cut. Technical report, Istituto di Analisi dei Sistemi ed Informatica del CNR, Italy, 2000.

[4] R. Fukasawa, M. Poggi de Aragão, O. Porto, and E. Uchoa. Robust branch-and-cut-and-price for the capacitated minimum spanning tree problem. In *Proceedings of the International Network Optimization Conference*, pages 231–236, Evry, France, 2003.

[5] R. Fukasawa, M. Poggi de Aragão, M. Reis, and E. Uchoa. Robust branch-and-cut-and-price for the capacitated vehicle routing problem. Technical Report RPEP Vol.3 no.8, Universidade Federal Fluminense, Engenharia de Produção, Niterói, Brazil, 2003. Available at *http://www.producao.uff.br/rpep/revista32003.htm.*

[6] P. Gilmore and R. Gomory. A linear programming approach to the cutting-stock problem. *Operations Research*, 9:849–859, 1961.

[7] D. Kim, C. Barnhart, K. Ware, and G. Reinhardt. Multimodal express package delivery: A service network design application. *Transportation Science*, 33:391–407, 1999.

[8] N. Kohl, J. Desrosiers, O. Madsen, M. Solomon, and F. Soumis. 2-path cuts for the vehicle routing with time windows. *Transportation Science*, 33:101–116, 1999.

[9] G. Nemhauser and S. Park. A polyhedral approach to edge coloring. *Operations Research Letters*, 10:315–322, 1991.

[10] A. Pigatti. Modelos e algoritmos para o problema de alocação generalizada e aplicações. Master's thesis, Pontifícia Universidade Católica do Rio de Janeiro, Brazil, July 2003.

[11] J. Van den Akker, C. Hurkens, and M. Savelsbergh. Time-indexed formulation for machine scheduling problems: column generation. *INFORMS Journal on Computing*, 12:111–124, 2000.

[12] F. Vanderbeck. Lot-sizing with start-up times. *Management Science*, 44:1409–1425, 1998.