Row-Reduced Column Generation for Degenerate Master Problems

Jacques Desrosiers and Jean Bertrand Gauthier
GERAD & HEC Montréal, Canada

Marco E. Lübbecke
RWTH Aachen University, Germany

November 9, 2011; revised: February 5, 2013

Abstract

Column generation for solving linear programs with a huge number of variables alternates between solving a master problem and a pricing subproblem to add variables to the master problem as needed. The method is known to suffer from degeneracy of the master problem, exposing what is called the tailing-off effect. Inspired by recent advances in coping with degeneracy in the primal simplex method, we propose a *row-reduced column generation* method that may take advantage of degenerate solutions. The idea is to reduce the number of constraints to the number of strictly positive basic variables in the current master problem solution. The advantage of this row-reduction is a smaller working basis, and thus a faster re-optimization of the master problem. This comes at the expense of a more involved pricing subproblem that needs to generate weighted subsets of variables that are compatible with the row-reduction, if possible. Such a compatible subset of variables gives rise to a strict improvement in the objective function value if the weighted combination of the reduced costs is negative. We thus state, as a by-product, a necessary and sufficient optimality condition for linear programming.

This methodological paper generalizes the *improved primal simplex* and *dynamic constraints aggregation* methods. On highly degenerate linear programs, recent computational experiments with these two algorithms show that the row-reduction of a problem might have a large impact on the solution time. We conclude with a few algorithmic and implementation issues.

Key Words: Column generation, degeneracy, dynamic row-reduction.

1 Introduction

Column generation, invented to solve large-scale linear programs (LPs), is particularly successful in the context of branch-and-price (Barnhart et al. 1998, Lübbecke and Desrosiers 2005) for solving well-structured integer programs. Column generation is used to solve the LP relaxations at each node of a search tree, and often produces strong dual bounds. It alternates between solving a restricted master problem (an LP) and one or several subproblems (usually integer programs) in order to dynamically add new variables to the model. However, column generation has a bad reputation for its slow convergence, known as the *tailing-off effect*, a drawback mainly attributed to the degeneracy of the restricted master LP solutions. This defect is particularly visible when solving LP relaxations of combinatorial optimization problems—a main application area of branch-and-price.

In this methodological paper, we present a *row-reduced column generation* (RrCG) method which turns degeneracy into a potential advantage. We dynamically partition the restricted master problem constraints based on the numerical values of the current basic variables. The idea is to keep only those constraints in the restricted master problem that correspond to strictly positive basic variables. This leads to a *row-reduced* restricted master problem which does not only discard most variables from consideration in column generation, but also reduces the number of constraints, and in particular the size of the current working basis. In linear algebra terms, we work with a projection into the subspace spanned by the column-vectors of the non-degenerate variables. This is similar to the idea of a deficient basis in the simplex method (Pan 1998).

Our work generalizes the *improved primal simplex* method (IPS) (Elhallaoui et al. 2010a, Raymond et al. 2010) for solving degenerate linear programs, and the *dynamic constraints aggregation* method (Elhallaoui et al. 2005, 2008, 2010b) for solving LP relaxations of set partitioning problems (by column generation) stemming from vehicle routing and crew scheduling applications. The referenced papers suggest that, on highly degenerate linear programs, a row-reduction of a problem shows great promise in reducing overall solution times.

The paper is organized as follows. Section 2 recalls the column generation method with the definitions of the master problem MP, its variable-restricted version RMP and its pricing subproblem SP. Section 3 presents the RrCG approach. It essentially defines the row and column partitions of the master problem based on a current degenerate solution, introduces the *row-reduced restricted master problem* RrRMP and its associated pricing subproblem rSP, and finally brings in a specialized column generator cSP for single columns compatible with the row-reduced master problem. Section 3.4 discusses the case of inequality constraints followed by an algorithm. Section 3.6 provides a necessary and sufficient optimality condition for linear programs. Finally Section 4 discusses some properties followed by implementation issues. Our conclusions complete the paper.

2 Column Generation and Notation

Let us briefly recall the mechanism of standard column generation, see Lübbecke and Desrosiers (2005) for a general introduction. We would like to solve the following linear program, called the *master problem* (MP), with a prohibitively large number of variables $\boldsymbol{\lambda} \in \mathbb{R}^n_+$

$$
\begin{array}{rlrcll}
z^\star_{MP} := & \min & \mathbf{c}^\intercal \boldsymbol{\lambda} & & & \\
& \text{s.t.} & A\boldsymbol{\lambda} & = & \mathbf{b} & [\boldsymbol{\pi}] \\
& & \boldsymbol{\lambda} & \geq & \mathbf{0} &
\end{array}
\tag{1}
$$

with $A \in \mathbb{R}^{m \times n}$, $\mathbf{c} \in \mathbb{R}^n$, and $\mathbf{b} \in \mathbb{R}^m$. The corresponding dual variables $\boldsymbol{\pi} \in \mathbb{R}^m$ are listed in brackets. We assume that $\boldsymbol{\lambda}$ includes m non-negative artificial variables, hence A is of full row rank, and MP is feasible if $\mathbf{b} \geq \mathbf{0}$. In applications, every coefficient column $\mathbf{a}$ of A encodes a combinatorial object $\mathbf{x} \in X$ like a path, permutation, set, or multi-set. To stress this fact, we write $\mathbf{a} = \mathbf{a}(\mathbf{x})$ and $c = c(\mathbf{x})$ for its cost coefficient. Column generation works with a *restricted master problem* (RMP) which involves a small subset of variables only. At each iteration, RMP is solved to optimality first. Then, like in the primal simplex algorithm, we look for a non-basic variable to price out and enter the current basis. That is, we either find a column $\mathbf{a}(\mathbf{x})$ of cost $c(\mathbf{x})$ with a negative reduced cost $\bar{c}(\mathbf{x})$ or need to prove that no such variable exists. This is accomplished by solving the *pricing subproblem* (SP)

$$
\bar{c}^\star_{SP} := \quad \min_{\mathbf{x} \in X} \{c(\mathbf{x}) - \boldsymbol{\pi}^\intercal \mathbf{a}(\mathbf{x})\}. \tag{2}
$$

If $\bar{c}^\star_{SP} \geq 0$, no negative reduced cost columns exist and the current solution $\boldsymbol{\lambda}$ of RMP (embedded into $\mathbb{R}^n_+$) optimally solves MP (1) as well. Otherwise, a minimizer of SP (2) gives rise to a variable to be added to RMP, and we iterate.

Functions $c(\mathbf{x})$ and $\mathbf{a}(\mathbf{x})$ may be linear functions, as in a Dantzig-Wolfe reformulation of a linear program (Dantzig and Wolfe 1960), but $c(\mathbf{x})$ is typically non-linear in many practical applications such as in rich vehicle routing and crew scheduling (Desaulniers et al. 1998). Non-linearities may increase the difficulty in solving SP, but it always ends up in a scalar cost c_j and a vector $\mathbf{a}_j$ of scalar coefficients for each variable λ_j in MP, $j \in \{1, \ldots, n\}$.

Notation. Vectors are written in bold face. We denote by I_k the $k \times k$ identity matrix and by $\mathbf{0}$ (resp. $\mathbf{1}$) a vector/matrix with all zero (resp. one) entries of appropriate contextual dimensions. For subsets $I \subseteq \{1, \ldots, m\}$ of row-indices and subsets $J \subseteq \{1, \ldots, n\}$ of column-indices we denote by A_{IJ} the sub-matrix of A containing the rows and columns indexed by I and J, respectively. We will further use standard linear programming notation like $A_J \boldsymbol{\lambda}_J$, the subset of columns of A which are indexed by J multiplied by the corresponding sub-vector of variables $\boldsymbol{\lambda}_J$. There is one notable exception: The set N will *not* denote the non-basis (but usually a superset). Even though one never actually computes the inverse of a basis matrix, our exposition will sometimes rely on "tableau data," when it is conceptually more convenient.

3 Row-Reduced Column Generation

RMP is a *column-reduced* MP and its variables are generated as needed by solving SP. The *row-reduced column generation* comes into play when the current solution of RMP is degenerate with $p < m$ positive variables. In what follows, we define a *row-reduced* RMP, denoted RrRMP, which decreases the number of rows to only p.

3.1 Row and Column Partitions

Let $\boldsymbol{\lambda}$ be a feasible solution to MP, with the index set $F \subset \{1, \ldots, n\}$ of variables at strictly positive value, that is, $\boldsymbol{\lambda}_F > \mathbf{0}$. These variables are *free* to increase or decrease relatively to their current values. All other possibly present variables assume a *null* value, that is, $\boldsymbol{\lambda}_N = \mathbf{0}$ for $N := \{1, \ldots, n\} \setminus F$. We assume that $\boldsymbol{\lambda}$ is degenerate in the sense that the number of positive variables is less than the number of rows of MP, i.e., $|F| = p < m$. The columns of A_F are required to be linearly independent, which is no restriction when $\boldsymbol{\lambda}$ is computed with a simplex algorithm. This assumption allows us to construct a basis matrix A_B for MP representing the solution $\boldsymbol{\lambda}$ in the following way. Identify a subset $P \subset \{1, \ldots, m\}$ of p linearly independent rows of A_F and "fill up" with $m - p$ unit columns to provide for artificial basic variables in the rows indexed by $Z := \{1, \ldots, m\} \setminus P$. More precisely, this yields the following form

$$A_B = \begin{bmatrix} A_{PF} & \mathbf{0} \\ A_{ZF} & I_{m-p} \end{bmatrix}. \tag{3}$$

One way to accomplish this form is to initialize the RMP with columns A_F and m artificial variables and apply a *phase I* of the primal simplex algorithm. The above construction induces row and column partitions, and MP (and the corresponding vector of dual variables $\boldsymbol{\pi}$) reads as

$$\begin{array}{rlcccccl} z^\star_{MP} := & \min & \mathbf{c}_F^\intercal \boldsymbol{\lambda}_F & + & \mathbf{c}_N^\intercal \boldsymbol{\lambda}_N & & & \\ & \text{s.t.} & A_{PF}\boldsymbol{\lambda}_F & + & A_{PN}\boldsymbol{\lambda}_N & = & \mathbf{b}_P & [\boldsymbol{\pi}_P] \\ & & A_{ZF}\boldsymbol{\lambda}_F & + & A_{ZN}\boldsymbol{\lambda}_N & = & \mathbf{b}_Z & [\boldsymbol{\pi}_Z] \\ & & \boldsymbol{\lambda}_F & , & \boldsymbol{\lambda}_N & \geq & \mathbf{0}. & \end{array} \tag{4}$$

The inverse of the above basis matrix (3) has a particularly easy form,

$$A_B^{-1} = \begin{bmatrix} A_{PF}^{-1} & \mathbf{0} \\ -A_{ZF}A_{PF}^{-1} & I_{m-p} \end{bmatrix}. \tag{5}$$

If we left-multiply (4) by A_B^{-1} we obtain the equivalent "tableau data" formulation

$$\begin{array}{rlcccccl} z^\star_{MP} := & \min & \mathbf{c}_F^\intercal \boldsymbol{\lambda}_F & + & \mathbf{c}_N^\intercal \boldsymbol{\lambda}_N & & & \\ & \text{s.t.} & \boldsymbol{\lambda}_F & + & \bar{A}_{PN}\boldsymbol{\lambda}_N & = & \bar{\mathbf{b}}_P & [\bar{\boldsymbol{\pi}}_P] \\ & & & & \bar{A}_{ZN}\boldsymbol{\lambda}_N & = & \mathbf{0} & [\bar{\boldsymbol{\pi}}_Z] \\ & & \boldsymbol{\lambda}_F & , & \boldsymbol{\lambda}_N & \geq & \mathbf{0}, & \end{array}$$

where $\bar{A}_{PN} = A_{PF}^{-1}A_{PN}$, $\bar{A}_{ZN} = A_{ZN} - A_{ZF}A_{PF}^{-1}A_{PN}$, and $[\bar{\boldsymbol{\pi}}_P, \bar{\boldsymbol{\pi}}_Z]^\intercal = [\boldsymbol{\pi}_P, \boldsymbol{\pi}_Z]^\intercal A_B$. We can see that our choice of basis reveals that our row partition reflects the degenerate solution in the tableau form, where $P = \{i \mid \bar{b}_i > 0\}$ and $Z = \{i \mid \bar{b}_i = 0\}$. Indeed, $(\boldsymbol{\lambda}_F, \boldsymbol{\lambda}_N)$ is a degenerate basic solution to MP.

According to the basis inverse A_B^{-1} described above, observe that $\bar{\boldsymbol{\pi}}_Z = \boldsymbol{\pi}_Z$, a dual vector that is equal to the costs of the artificial variables selected as basic. The current solution $\boldsymbol{\lambda}_F = \bar{\mathbf{b}}_P = A_{PF}^{-1}\mathbf{b}_P$ is computed using the smaller *working basis* A_{PF}^{-1}, a $p \times p$ matrix. Finally, observe that there is no need to left-multiply the system of constraints in row-set P by A_{PF}^{-1}. Hence, MP can be expressed as follows:

$$\begin{array}{rllcllll} z^\star_{MP} := & \min & \mathbf{c}_F^\intercal \boldsymbol{\lambda}_F & + & \mathbf{c}_N^\intercal \boldsymbol{\lambda}_N & & & \\ & \text{s.t.} & A_{PF}\boldsymbol{\lambda}_F & + & A_{PN}\boldsymbol{\lambda}_N & = & \mathbf{b}_P & [\boldsymbol{\pi}_P] \\ & & & & \bar{A}_{ZN}\boldsymbol{\lambda}_N & = & \mathbf{0} & [\boldsymbol{\pi}_Z] \\ & & \boldsymbol{\lambda}_F & , & \boldsymbol{\lambda}_N & \geq & \mathbf{0}, & \end{array} \tag{6}$$

and the pricing subproblem SP now equivalently writes as

$$\bar{c}^\star_{SP} := \min_{\mathbf{x}\in X} \left\{ c(\mathbf{x}) - \boldsymbol{\pi}_P^\intercal \mathbf{a}_P(\mathbf{x}) - \boldsymbol{\pi}_Z^\intercal \bar{\mathbf{a}}_Z(\mathbf{x}) \right\}. \tag{7}$$

We emphasize that the basis we constructed for MP is not unique. The form we propose has convenient properties, but even there it is our choice which rows appear in P.

3.2 The Row-Reduced Master and the Pricing Subproblem

To exploit the degeneracy of the solution exhibited in (6), we *row-reduce* the RMP, denoted by RrRMP. To this end, we discard the rows in set Z:

$$\begin{array}{rllcllll} z^\star_{RrRMP} := & \min & \mathbf{c}_F^\intercal \boldsymbol{\lambda}_F & + & \mathbf{c}_N^\intercal \boldsymbol{\lambda}_N & & & \\ & \text{s.t.} & A_{PF}\boldsymbol{\lambda}_F & + & A_{PN}\boldsymbol{\lambda}_N & = & \mathbf{b}_P & [\boldsymbol{\pi}_P] \\ & & \boldsymbol{\lambda}_F & , & \boldsymbol{\lambda}_N & \geq & \mathbf{0}. & \end{array} \tag{8}$$

Solution $(\boldsymbol{\lambda}_F, \boldsymbol{\lambda}_N)$ is optimal for MP (6) if no negative reduced cost columns exist, that is, if $\bar{c}^\star_{SP} \geq 0$, or equivalently, if $\bar{\mathbf{c}}_N \geq \mathbf{0}$. From (8), it holds that $\boldsymbol{\pi}_P^\intercal = \mathbf{c}_F^\intercal A_{PF}^{-1}$. However, the value of dual vector $\boldsymbol{\pi}_Z$ is not known. Consequently, we cannot solve the pricing subproblem SP as expressed in (7), and we need to come up with an alternative.

Compute the *partial reduced cost* vector $\tilde{\mathbf{c}}_N^\intercal := \mathbf{c}_N^\intercal - \boldsymbol{\pi}_P^\intercal \bar{A}_{PN} = \mathbf{c}_N^\intercal - \mathbf{c}_F^\intercal A_{PF}^{-1} A_{PN}$ and write the current reduced cost vector $\bar{\mathbf{c}}_N$ in terms of the unknown vector $\boldsymbol{\pi}_Z$ of dual variables: $\bar{\mathbf{c}}_N^\intercal = \tilde{\mathbf{c}}_N^\intercal - \boldsymbol{\pi}_Z^\intercal \bar{A}_{ZN}$. To verify the non-negativity of $\bar{\mathbf{c}}_N$, one can find the minimum value of its components by solving $\max\{\gamma \mid \gamma \leq \bar{c}_j,\ \forall j \in N\}$, i.e., by solving the following pricing subproblem rSP over γ and the unknown vector $\boldsymbol{\pi}_Z$

$$\begin{array}{rlrcll} \bar{c}^\star_{rSP} := & \max & \gamma & & & \\ & \text{s.t.} & \mathbf{1}\gamma + \bar{A}_{ZN}^\intercal \boldsymbol{\pi}_Z & \leq & \tilde{\mathbf{c}}_N & [\boldsymbol{\lambda}_N], \end{array} \tag{9}$$

where $\boldsymbol{\lambda}_N \geq \mathbf{0}$ acts as the dual variable vector. In other words, given *only* the non-degenerate variables $\boldsymbol{\lambda}_F > \mathbf{0}$ of cost $\mathbf{c}_F$ and the associated columns in A_F from which we derive the row-partition, we compute the dual vector $\boldsymbol{\pi}_P$ and check whether any vector $\boldsymbol{\pi}_Z$ exists such that $\gamma < 0$

(to generate a column with negative reduced cost to be added to problem RrRMP) or to otherwise prove the optimality of $\boldsymbol{\lambda}_F$ for RrRMP (and hence for MP). The dual of (9) defines the pricing problem rSP in terms of $\boldsymbol{\lambda}_N$, the current vector of null variables:

$$\begin{array}{rlrcll} \bar{c}^\star_{rSP} := & \min & \tilde{\mathbf{c}}_N^\intercal \boldsymbol{\lambda}_N & & & \\ & \text{s.t.} & \mathbf{1}^\intercal \boldsymbol{\lambda}_N & = & 1 & [\gamma\,] \\ & & \bar{A}_{ZN}\boldsymbol{\lambda}_N & = & \mathbf{0} & [\boldsymbol{\pi}_Z] \\ & & \boldsymbol{\lambda}_N & \geq & \mathbf{0}\ . & \end{array} \tag{10}$$

3.3 Solving the Pricing Step

An optimal solution $\boldsymbol{\lambda}_N^\star$ to (10) may contain a single variable $\lambda_j^\star = 1,\ j \in N$, for which $\bar{\mathbf{a}}_{Zj} = \mathbf{0}$, or more generally, a convex combination of several positive variables such that $\bar{A}_{ZN}\boldsymbol{\lambda}_N^\star = \mathbf{0}$. In the pricing step, we consider two subproblems: rSP defined by (10) and a specialized one denoted cSP exploiting the following property:

Definition 1. *Given the solution vector $\boldsymbol{\lambda}_F > \mathbf{0}$ of positive variables, vector $\mathbf{a} = \begin{bmatrix} \mathbf{a}_P \\ \mathbf{a}_Z \end{bmatrix}$ is called* compatible with row-set P *if and only if $\bar{\mathbf{a}}_Z := \mathbf{a}_Z - A_{ZF}A_{PF}^{-1}\mathbf{a}_P = \mathbf{0}$.*

Vector $\mathbf{b}$ is compatible since $\bar{\mathbf{b}}_Z = \mathbf{0}$ and so are the column-vectors of A_F. When appropriate, we also say that a variable associated with a compatible column is compatible. The artificial basic variables we selected are incompatible. More generally, degenerate basic variables are incompatible. The interest in compatibility comes from the fact that a compatible column $\mathbf{a}_j$ with negative reduced cost $\bar{c}_j$ yields a non-degenerate pivot. Indeed, the step size given by the *ratio-test* is computed only on the row-set P, that is, $\rho_j = \min_{i\in P}\left\{\frac{\bar{b}_i}{\bar{a}_{ij}} \mid \bar{a}_{ij} > 0\right\}$. Because $\bar{b}_i > 0,\ \forall i \in P$, then $\rho_j > 0$ and the objective of MP strictly improves by $\rho_j\bar{c}_j < 0$, unless $\bar{a}_{ij} \leq 0,\ \forall i \in P$, in which case MP is unbounded.

On the one hand, if we restrict our attention to only generating compatible columns (which is a natural idea in our context), matters can sometimes simplify considerably as there is no need to know $\boldsymbol{\pi}_Z$. Given the dual vector $\boldsymbol{\pi}_P^\intercal = \mathbf{c}_F^\intercal A_{PF}^{-1}$ retrieved from the solution of RrRMP (8), we define a specialized subproblem cSP which is the pricing subproblem SP (7) augmented with a set of linear constraints imposing compatibility with the row-reduced master problem for solution-column $\mathbf{a}(\mathbf{x})$, that is,

$$\bar{c}^\star_{cSP} \quad := \quad \min_{\mathbf{x}\in X}\left\{c(\mathbf{x}) - \mathbf{c}_F^\intercal A_{PF}^{-1}\mathbf{a}_P(\mathbf{x}) \ \mid\ \mathbf{a}_Z(\mathbf{x}) - A_{ZF}A_{PF}^{-1}\mathbf{a}_P(\mathbf{x}) = \mathbf{0}\right\}. \tag{11}$$

Adding the set of constraints $\mathbf{a}_Z(\mathbf{x}) - A_{ZF}A_{PF}^{-1}\mathbf{a}_P(\mathbf{x}) = \mathbf{0}$ may destroy the structure of the subproblem and makes it more difficult to solve in some cases. Anyhow, if a non-basic compatible column $\mathbf{a}_j, j \in N$, with $\bar{c}_j = \bar{c}^\star_{cSP} < 0$ is generated from the solution of (11), one updates the current solution as:

$$\begin{array}{lcll} \lambda_j & = & \rho_j & \\ \lambda_k & = & 0 & \forall k \in N \setminus \{j\} \\ \boldsymbol{\lambda}_F & = & \bar{\mathbf{b}}_P - \rho_j\bar{\mathbf{a}}_{Pj} & \\ z_{RrRMP} & = & \mathbf{c}_F\bar{\mathbf{b}}_P + \rho_j\bar{c}^\star_{cSP}\ . & \end{array} \tag{12}$$

The number of positive variables in the new solution is at most p, that is, it can be more degenerate.

On the other hand, we can solve rSP (10) to look for a convex combination of columns to improve the objective value of MP. Subproblem rSP is a linear program solved by column generation over X. In its pricing step, we look for a column $\begin{bmatrix} 1 \\ \bar{\mathbf{a}}_Z(\mathbf{x}) \end{bmatrix}$ with a reduced cost $\bar{c}(\mathbf{x}) = \tilde{c}(\mathbf{x}) - \gamma - \boldsymbol{\pi}_Z^\intercal \bar{\mathbf{a}}_Z(\mathbf{x})$ of negative value, where $\tilde{c}(\mathbf{x}) := c(\mathbf{x}) - \boldsymbol{\pi}_P^\intercal \mathbf{a}_P(\mathbf{x})$ and $\bar{\mathbf{a}}_Z(\mathbf{x}) := \mathbf{a}_Z(\mathbf{x}) - A_{ZF} A_{PF}^{-1} \mathbf{a}_P(\mathbf{x})$. Given $\boldsymbol{\pi}_P^\intercal = \mathbf{c}_F^\intercal A_{PF}^{-1}$ obtained from the solution of RrRMP (8) together with γ and $\boldsymbol{\pi}_Z$ retrieved from the current solution of (10), the pricing subproblem for generating variables as needed for solving rSP is given by

$$\begin{aligned} \bar{c}^\star &:= -\gamma + \min_{\mathbf{x} \in X} \left\{ c(\mathbf{x}) - \mathbf{c}_F^\intercal A_{PF}^{-1} \mathbf{a}_P(\mathbf{x}) - \boldsymbol{\pi}_Z^\intercal \bar{\mathbf{a}}_Z(\mathbf{x}) \right\} \\ &= -\gamma + \min_{\mathbf{x} \in X} \left\{ c(\mathbf{x}) - (\mathbf{c}_F^\intercal A_{PF}^{-1} - \boldsymbol{\pi}_Z^\intercal A_{ZF} A_{PF}^{-1}) \mathbf{a}_P(\mathbf{x}) - \boldsymbol{\pi}_Z^\intercal \mathbf{a}_Z(\mathbf{x}) \right\} . \end{aligned} \tag{13}$$

Apart from the constant term $-\gamma$, (13) is the usual SP (2) with dual vector

$$(\boldsymbol{\pi}_P^\intercal,\ \boldsymbol{\pi}_Z^\intercal) = (\mathbf{c}_F^\intercal A_{PF}^{-1} - \boldsymbol{\pi}_Z^\intercal A_{ZF} A_{PF}^{-1},\ \boldsymbol{\pi}_Z^\intercal) \ .$$

Subproblem rSP (10) combines variables in N such that $\bar{A}_{ZN} \boldsymbol{\lambda}_N^\star = \mathbf{0}$. Hence, vector $A_N \boldsymbol{\lambda}_N^\star$ (of reduced cost value $\bar{c}_{rSP}^\star$) is compatible with row-set P. Given $\boldsymbol{\lambda}_N^\star$, the updated values are computed according to RrRMP (8) as

$$\begin{aligned} \boldsymbol{\lambda}_N &= \rho \boldsymbol{\lambda}_N^\star \\ \boldsymbol{\lambda}_F &= \bar{\mathbf{b}}_P - \rho \bar{A}_{PN} \boldsymbol{\lambda}_N^\star \\ z_{RrRMP} &= \mathbf{c}_F \bar{\mathbf{b}}_P + \rho \bar{c}_{rSP}^\star \ . \end{aligned} \tag{14}$$

The number of positive variables is at most $p + (m - p + 1) - 1 = m$ and the new solution could be less as well as more degenerate.

Finally, several compatible columns can be retrieved from the pricing problem cSP and added simultaneously to (8). Moreover, the positive variables of $\boldsymbol{\lambda}_N^\star$ solution of rSP can be entered one by one in RMP (1), in any order, see Elhallaoui et al. (2010a): the last variable entered ensures a non-degenerate pivot because the convex combination $A_N \boldsymbol{\lambda}_N^\star$ is compatible.

3.4 Inequality Constraints

Consider the following linear master problem MP with greater-or-equal inequality constraints:

$$z_{MP}^\star := \min \quad \mathbf{c}^\intercal \boldsymbol{\lambda} \quad \text{s.t.} \quad A\boldsymbol{\lambda} \geq \mathbf{b}, \qquad \boldsymbol{\lambda} \geq \mathbf{0} \ . \tag{15}$$

Introducing a vector of surplus variables $\boldsymbol{\delta} \in \mathbb{R}_+^m$, one obtains MP in standard form:

$$z_{MP}^\star := \min \quad \mathbf{c}^\intercal \boldsymbol{\lambda} \quad \text{s.t.} \quad A\boldsymbol{\lambda} - \ \boldsymbol{\delta} = \mathbf{b}, \qquad \boldsymbol{\lambda},\ \boldsymbol{\delta} \geq \mathbf{0} \ . \tag{16}$$

The case with less-or-equal inequality constraints $A\boldsymbol{\lambda} \leq \mathbf{b}$ can be treated in a similar way by considering the transformation $-A\boldsymbol{\lambda} \geq -\mathbf{b}$. When the current basic solution to (15) with vector of positive variables $\boldsymbol{\lambda}_F > \mathbf{0}$ is such that $A_F\boldsymbol{\lambda}_F \neq \mathbf{b}$, the basis also contains some vector of surplus variables $\boldsymbol{\delta}_S > \mathbf{0}$, for $S \subset \{1, \ldots, m\}$. Let $|S| = s$ and denote by $Z := \{1, \ldots, m\} \setminus (F \cup S)$ the set of remaining rows. The basis can be written in terms of $\boldsymbol{\lambda}_F > \mathbf{0}$, the slack variables $\boldsymbol{\delta}_S > \mathbf{0}$, and again artificial variables for the $m - p - s$ remaining constraints. Basis A_B and its inverse write as

$$A_B = \left[\begin{array}{cc|c} A_{PF} & \mathbf{0} & \mathbf{0} \\ A_{SF} & -I_s & \mathbf{0} \\ \hline A_{ZF} & \mathbf{0} & I_{m-p-s} \end{array}\right], \qquad A_B^{-1} = \left[\begin{array}{cc|c} A_{PF}^{-1} & \mathbf{0} & \mathbf{0} \\ A_{SF}A_{PF}^{-1} & -I_s & \mathbf{0} \\ \hline A_{ZF}A_{PF}^{-1} & \mathbf{0} & I_{m-p-s} \end{array}\right].$$

Upon left-multiplication by the inverse, the transformed master problem becomes:

$$\begin{array}{rlllllllll} z^\star_{MP} := \min & \mathbf{c}_F^\intercal\boldsymbol{\lambda}_F & & + & \mathbf{c}_N^\intercal\boldsymbol{\lambda}_N & & & & & \\ \text{s.t.} & \boldsymbol{\lambda}_F & & + & \bar{A}_{PN}\boldsymbol{\lambda}_N & - & A_{PF}^{-1}\boldsymbol{\delta}_P & & = & \bar{\mathbf{b}}_P \\ & & \boldsymbol{\delta}_S & + & \bar{A}_{SN}\boldsymbol{\lambda}_N & - & A_{SF}A_{PF}^{-1}\boldsymbol{\delta}_P & & = & \bar{\mathbf{b}}_S \\ & & & & \bar{A}_{ZN}\boldsymbol{\lambda}_N & - & A_{ZF}A_{PF}^{-1}\boldsymbol{\delta}_P & -\boldsymbol{\delta}_Z & = & \mathbf{0} \\ & \boldsymbol{\lambda}_F, & \boldsymbol{\delta}_S, & & \boldsymbol{\lambda}_N, & & \boldsymbol{\delta}_P, & \boldsymbol{\delta}_Z & \geq & \mathbf{0}. \end{array} \tag{17}$$

From (17), we can derive RrRMP, cSP, and rSP. The row-reduced master problem with $p + s$ constraints is obtained by discarding row-set Z from the formulation while keeping the original data matrices and inequality constraints:

$$\begin{array}{rlllllll} z^\star_{RrRMP} := \min & \mathbf{c}_F^\intercal\boldsymbol{\lambda}_F & + & \mathbf{c}_N^\intercal\boldsymbol{\lambda}_N & & & \\ \text{s.t.} & A_{PF}\boldsymbol{\lambda}_F & + & A_{PN}\boldsymbol{\lambda}_N & \geq & \mathbf{b}_P & [\boldsymbol{\pi}_P] \\ & A_{SF}\boldsymbol{\lambda}_F & + & A_{SN}\boldsymbol{\lambda}_N & \geq & \mathbf{b}_S & [\boldsymbol{\pi}_S] \\ & \boldsymbol{\lambda}_F \geq \mathbf{0}, & & \boldsymbol{\lambda}_N \geq \mathbf{0} & , & & \end{array} \tag{18}$$

where $\boldsymbol{\pi}_P, \boldsymbol{\pi}_S \geq \mathbf{0}$. Current (known) dual values are $\boldsymbol{\pi}_P^\intercal = \mathbf{c}_F^\intercal A_{PF}^{-1}$ and $\boldsymbol{\pi}_S^\intercal = \mathbf{0}$. Therefore, subproblem cSP for generating compatible variables is given by (11) and it only depends on the column-vectors A_F of the positive variables $\boldsymbol{\lambda}_F > \mathbf{0}$. Subproblem rSP, again written in terms of the current null variable vectors $\boldsymbol{\lambda}_N$, $\boldsymbol{\delta}_P$, and $\boldsymbol{\delta}_Z$, becomes:

$$\begin{array}{rllllllll} z^\star_{rSP} := \min & \tilde{\mathbf{c}}_N^\intercal\boldsymbol{\lambda}_N & + & \mathbf{c}_F^\intercal A_{PF}^{-1}\boldsymbol{\delta}_P & & & & & \\ \text{s.t.} & \mathbf{1}^\intercal\boldsymbol{\lambda}_N & + & \mathbf{1}^\intercal\boldsymbol{\delta}_P & + & \mathbf{1}^\intercal\boldsymbol{\delta}_Z & = & 1 & [\gamma] \\ & \bar{A}_{ZN}\boldsymbol{\lambda}_N & - & A_{ZF}A_{PF}^{-1}\boldsymbol{\delta}_P & - & \boldsymbol{\delta}_Z & = & \mathbf{0} & [\boldsymbol{\pi}_Z] \\ & \boldsymbol{\lambda}_N, & & \boldsymbol{\delta}_P, & & \boldsymbol{\delta}_Z & \geq & \mathbf{0}. & \end{array} \tag{19}$$

It is solved by column generation except that the $\boldsymbol{\delta}_P$ and $\boldsymbol{\delta}_Z$ variables need not be generated. Therefore, (13) can be used to price out the valuable columns of $\boldsymbol{\lambda}_N$. Its solution is a convex combination of the variables in $\boldsymbol{\lambda}_N$, $\boldsymbol{\delta}_P$ and $\boldsymbol{\delta}_Z$ such that $\bar{A}_{ZN}\boldsymbol{\lambda}_N^\star - A_{ZF}A_{PF}^{-1}\boldsymbol{\delta}_P^\star - \boldsymbol{\delta}_Z^\star = \mathbf{0}$.

3.5 An RrCG Algorithm

We summarize our discussion with a pseudo-code of our *row-reduced column generation* algorithm for degenerate master problems. As long as the solution is non-degenerate, we have the classical alternation between RMP and SP (until line 9). When a degenerate solution is identified in line 10 ($p < m$), the row-reduced RrRMP benefits from this (lines 11 to 19).

After solving RMP (line 4) or RrRMP (line 13), the optimality test for MP is via the solution of a pricing subproblem, either the classical SP (line 8) if the current solution is non-degenerate or rSP which is solved by column generation (line 17). In both situations, new columns are added to RMP if $\bar{c}^{\star}_{SP} < 0$ or $\bar{c}^{\star}_{rSP} < 0$, otherwise MP is optimal.

If the current RMP solution is degenerate in line 10, row-index sets P and Z are defined/updated (line 11), and the RrRMP is built and solved (line 13). In that case, priority can be given to the specialized pricing subproblem cSP (line 14) and, if $\bar{c}^{\star}_{cSP} < 0$, compatible columns are added to RrRMP (line 15). Otherwise, the pricing subproblem rSP needs to be solved by column generation (line 17). If $\bar{c}^{\star}_{rSP} < 0$ in line 18, we add subsets of incompatible columns to RMP and iterate (again from line 4). The algorithm stops when RMP and hence MP are optimal (line 19).

1 **algorithm** row-reduced column generation RrCG;
2 initialize RMP (1);
3 **repeat**
4   solve RMP (1);
5   identify matrix A_F with $p \leq m$ positive variables;
6   construct basis A_B containing $m - p$ artificial variables;
7   **if** $p = m$ **then** `// non-degenerate solution`
8   solve SP (2);
9   **if** $\bar{c}^{\star}_{SP} < 0$ **then** add column(s) to RMP (1);
10   **else** `//` $p < m$`, degenerate solution`
11   if necessary, update $P := \{i \mid \bar{b}_i > 0\}$, $Z := \{i \mid \bar{b}_i = 0\}$;
12   **repeat**
13   solve RrRMP (8), the *row-reduced* RMP;
14   solve cSP (11);
15   **if** $\bar{c}^{\star}_{cSP} < 0$ **then** add compatible column(s) to RrRMP (8);
16   **until** *no more compatible columns added*;
17   solve rSP (10) by column generation;
18   **if** $\bar{c}^{\star}_{rSP} < 0$ **then** add subsets of incompatible columns to RMP (1);
19 **until** *no more columns added*;

3.6 A Characterization of LP Optimality

Column generation generalizes the primal simplex algorithm. In the same spirit, the *improved primal simplex* method (IPS) used for solving degenerate linear programs (Elhallaoui et al. 2010a, Raymond et al. 2010) can be seen as a special case of the *row-reduced column generation* RrCG. The main difference is that rSP (10) itself needs to be solved by column generation, whereas in IPS all columns are explicitly given in advance. Moreover, given basis A_B, all variables can be characterized *a priori* as either compatible or incompatible. Hence, in IPS, the row-reduced master problem is defined on the compatible variables and a *complementary pricing subproblem* is solved over the incompatible variables only. Although not explicitly stated in previous IPS papers, the following result is a direct consequence of the pricing subproblem structure.

Theorem 1. *A feasible solution* $(\boldsymbol{\lambda}_F > \mathbf{0}, \boldsymbol{\lambda}_N = \mathbf{0})$, $|F| \leq m$, *is optimal for the linear program* (1) *if and only if there exists some dual vector* $\boldsymbol{\pi}_Z$ *such that* $\bar{c}^\star_{rSP} \geq 0$, rSP *being defined by* (10).

Proof. For necessity, recall that MP (1) is equivalent to MP (6). Firstly, $\boldsymbol{\lambda}_F = A^{-1}_{PF}\mathbf{b}_P$ and $\boldsymbol{\lambda}_N = \mathbf{0}$ is primal feasible for MP (6). Secondly, $\boldsymbol{\lambda}_F$ being basic, $\bar{\mathbf{c}}^\intercal_F = \mathbf{0}$, i.e., $\bar{\mathbf{c}}^\intercal_F - \boldsymbol{\pi}^\intercal_P A_{PF} = \mathbf{0}$. If there exists a dual vector $\boldsymbol{\pi}_Z$ such that $\bar{\mathbf{c}}^\intercal_N = \mathbf{c}^\intercal_N - \mathbf{c}^\intercal_F \bar{A}_{PN} - \boldsymbol{\pi}^\intercal_Z \bar{A}_{ZN} \geq \mathbf{0}$, the reduced costs of all variables are non-negative and $(\mathbf{c}^\intercal_F A^{-1}_{PF}, \boldsymbol{\pi}^\intercal_Z)$ is dual feasible for MP (6). Thirdly, since $\boldsymbol{\lambda}_N = \mathbf{0}$ and $\bar{\mathbf{b}}_Z = \mathbf{0}$, primal objective function $\mathbf{c}^\intercal_F \boldsymbol{\lambda}_F$ is equal to the dual objective function $\boldsymbol{\pi}^\intercal_P \mathbf{b}_P$. Therefore $[\boldsymbol{\lambda}_F, \boldsymbol{\lambda}_N]$ is optimal.

To show sufficiency, let $[\boldsymbol{\lambda}_F, \boldsymbol{\lambda}_N]$ be an optimal solution and assume $\bar{c}^\star_{rSP} < 0$. With the row-partition induced by basis A_B, $A_N \boldsymbol{\lambda}^\star_N = \begin{bmatrix} A_{PN}\boldsymbol{\lambda}^\star_N \\ A_{ZN}\boldsymbol{\lambda}^\star_N \end{bmatrix}$. Because vector $\bar{A}_{ZN}\boldsymbol{\lambda}^\star_N = \mathbf{0}$ in (10), $A_N \boldsymbol{\lambda}^\star_N$ is compatible. If $\boldsymbol{\lambda}^\star_N$ contains a single variable $\lambda^\star_j = 1$, $j \in N$, then $\bar{\mathbf{a}}^Z_j = \mathbf{0}$, vector $\mathbf{a}_j$ is compatible, and $\bar{c}_j = \tilde{c}_j = \bar{c}^\star_{rSP} < 0$. The p-dimensional column $\mathbf{a}_{Pj}$, and the associated variable λ_j in MP enters the basis of RrRMP, a non-degenerate pivot occurs (unless $\bar{\mathbf{a}}_{Pj} \leq \mathbf{0}$ in which case MP is unbounded), and the objective function improves by $\rho_j \bar{c}^\star_{rSP} < 0$.

More generally, if $\boldsymbol{\lambda}^\star_N$ contains several positive variables, the p-dimensional vector $A_{PN}\boldsymbol{\lambda}^\star_N$ can enter the basis of RrRMP as a single column. Let $(\bar{a}_i)_{i \in P} := \bar{A}_{PN}\boldsymbol{\lambda}^\star_N$. Because $\bar{A}_{ZN}\boldsymbol{\lambda}^\star_N = \mathbf{0}$, its reduced cost is computed as $\bar{\mathbf{c}}^\intercal_N \boldsymbol{\lambda}^\star_N = \tilde{\mathbf{c}}^\intercal_N \boldsymbol{\lambda}^\star_N = \bar{c}^\star_{rSP} < 0$. Unless all components of vector $(\bar{a}_i)_{i \in P}$ are non-positive, it strictly improves the objective function value by $\rho \bar{c}^\star_{rSP} < 0$ when added to RrRMP, where $\rho = \min_{i \in P} \left\{ \frac{\bar{b}_i}{\bar{a}_i} \mid \bar{a}_i > 0 \right\}$. This contradicts the optimality of $(\boldsymbol{\lambda}_F, \boldsymbol{\lambda}_N)$ and completes the proof. □

4 Some Properties and Implementation Issues

4.1 Selection of a Working Basis A_{PF}

Given the current solution $(\boldsymbol{\lambda}_F, \boldsymbol{\lambda}_N)$ to RMP, the $p \times p$ working basis A_{PF} is a matrix containing p linearly independent rows of A_F, not uniquely defined if $p < m$. Matrix A_{PF} does not only characterize the row-partition of the constraints of RrRMP, rSP and cSP, but also, for $j \in N$, the partial reduced costs coefficients $\tilde{c}_j = c_j - \mathbf{c}^\intercal_F A^{-1}_{PF}\mathbf{a}_{Pj}$, and column components of $\bar{\mathbf{a}}_j$, that is, $\bar{\mathbf{a}}_{Pj} = A^{-1}_{PF}\mathbf{a}_{Pj}$ and $\bar{\mathbf{a}}_{Zj} = \mathbf{a}_{Zj} - A^{-1}_{PF}\mathbf{a}_{Pj}$. However, we show by a linear algebra argument that, for any set of p linearly independent rows of A_F, the pricing subproblem holds the same information based on A_F and thus provides the *same solution set.* That is, the latter is independent of the row-partition induced by the selected working basis A_{PF}. This comes from the fact that only compatible columns belong to the vector space spanned by A_F.

Proposition 1. *Given the solution vector* $\boldsymbol{\lambda}_F > \mathbf{0}$ *of positive variables, vector* $\mathbf{a}$ *is* compatible with the row-set P *if and only if it belongs to the vector space spanned by* A_F.

Proof. Assume vector $\mathbf{a} = A_F\mathbf{y}$, that is, it can be written as a linear combination of the columns of A_F, or equivalently, $\begin{bmatrix} A_{PF} \\ A_{ZF} \end{bmatrix} \mathbf{y} = \begin{bmatrix} \mathbf{a}_P \\ \mathbf{a}_Z \end{bmatrix}$. Since A_{PF} is invertible, one obtains $\mathbf{y} = A_{PF}^{-1}\mathbf{a}_P$ from the first set of constraints. Substituting in the second set, we have $A_{ZF}A_{PF}^{-1}\mathbf{a}_P = \mathbf{a}_Z$ which means that $\mathbf{a}$ is compatible by Definition 1. To show the converse, let $\mathbf{a}$ be compatible with with the row-set P, that is, $\bar{\mathbf{a}}_Z := \mathbf{a}_Z - A_{ZF}A_{PF}^{-1}\mathbf{a}_P = \mathbf{0}$, where $\bar{\mathbf{a}}_P := A_{PF}^{-1}\mathbf{a}_P$. Hence, $\mathbf{a}_Z = A_{ZF}\bar{\mathbf{a}}_P$ and $\mathbf{a}_P = A_{PF}\bar{\mathbf{a}}_P$, or equivalently, $\mathbf{a} = A_F\bar{\mathbf{a}}_P$. Hence $\mathbf{a}$ belongs to the vector space spanned by A_F. □

This has a nice interpretation in set partitioning models that are common in vehicle routing, crew scheduling, and many other applications. Given an integer solution, A_F forms a set of p groups of rows. Hence, compatible columns are those combining these groups (and A_{PF} is the identity matrix I_p). For a fractional solution, the p independent rows of A_{PF} can be chosen as follows: keep one row from each group of identical rows of A_{PF}. Again this forms a set of p groups of rows, and the same compatibility interpretation applies. This is what is being used in the *dynamic constraints aggregation* method (Elhallaoui et al. 2005, 2008, 2010b, Benchimol et al. 2012). Furthermore, the convex combination of incompatible variables in (10) simply expresses the fundamental exchange mechanism of set partitioning solutions, that is, removing elements from some groups to insert them back in other groups.

Another interpretation can be given in the context of the minimum cost flow problem with non-negative flow variables, see Ahuja et al. (1993). Any feasible flow can be represented (in the residual network) as a collection of positive arcs (the non-degenerate arcs forming a forest) and all other arcs at zero (the degenerate ones). The column-vectors of the positive arcs form A_F. A degenerate arc is compatible if and only if it can be written in terms of the unique subset of positive arcs forming a cycle with it. Figure 1 represents the residual network in which the flow on the non-degenerate (positive) arcs is free to go in both directions while the degenerate arcs are at zero. Arc (8,9) can be written in terms of the free arcs (9,10), (11,10) and (11,8), hence the corresponding variable $x_{8,9}$ is compatible. An incompatible arc must link two trees of the forest. The combination of several incompatible arcs together with a selection of free arcs can form a cycle. In Figure 1, (6,9) and (5,11) are two incompatible arcs. However, combined with free arcs (9,10), (11,10), (5,1) and (1,6), they form a cycle, hence the sum $x_{6,9} + x_{5,11}$ of these two incompatible variables is compatible as well as their convex combination with an equal weight of one-half on each. Now observe the row partition derived from that network flow example. In the small tree composed of free arcs (1,5) and (1,6), two independent flow conservation constraints need to be selected to be part of set P, hence any one amongst nodes 1, 5 or 6 appears in set Z. This is the same mechanism for the larger tree composed of the five arcs (2,11), (2,7), (11,10), (11,8), and (9,10): any one amongst nodes 2, 7, 11, 10, 8 or 9 is selected to be part of set Z while the remaining four flow conservation constraints are in the set of independent rows of P. This leads us to the following two propositions regarding the choice of the working basis.

Consider two different working bases A_{PF} and A_{QF}, where Q provides an alternative set of p linearly independent rows of A_F. The reduced costs of the positive basic variables are zero, hence

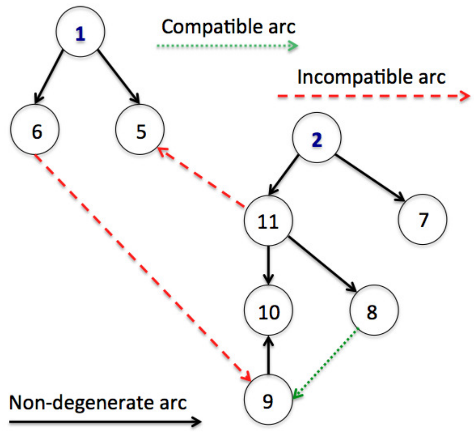

Figure 1: Compatible and incompatible arcs in a minimum cost flow problem

independent of the selected working basis. However, their column coefficients in RrRMP (8) are different since they are given by the column coefficients of the selected working basis. For cSP, Proposition 2 shows that the reduced costs of the compatible variables are indeed identical in the two subproblem versions, namely cSP_P and cSP_Q and their coefficient components are equivalent. Therefore cSP, the specialized pricing subproblem for generating compatible variables, is independent of the selected working basis. The proof is presented in the Appendix.

Proposition 2. *Given two working bases A_{PF} and A_{QF} of A_F, the corresponding pricing subproblems* cSP_P *and* cSP_Q *are equivalent programs.*

Although the reduced cost coefficients and column components of incompatible variables in rSP (10) clearly depend on the selected working basis, say either A_{PF} or A_{QF}, Proposition 3 shows that the solution set of the corresponding pricing subproblems rSP_P and rSP_Q is the same as well as the values of their optimal objective functions. Hence they are equivalent programs. Again the proof is postponed to the Appendix.

Proposition 3. *Given two working bases A_{PF} and A_{QF} of A_F, the corresponding pricing subproblems* rSP_P *and* rSP_Q *are equivalent programs.*

Regarding the interpretations in set partitioning models and minimum cost flow problems, we have the following. In the first case, for both integer or fractional solutions, each group is composed of identical rows in A_F, hence a single one per group is selected to appear in row-set P. The choice is therefore irrelevant. In the second case, each tree t of the forest contains n_t nodes and $n_t - 1$ free arcs. One (root) node per tree has to be removed and its flow conservation constraint appears in row-set Z. The choice of a root node per tree of the forest, currently nodes 1 and 2 in Figure 1, does not change the composition of any cycle nor its cost or reduced cost.

4.2 Flexibility in an Implementation

A consequence of the necessary and sufficient optimality condition of Theorem 1 is that the role of RrRMP could be relegated to only updating the current feasible solution, see (14). Indeed, there is no need to keep the generated columns, except those comprised in the current solution. Pricing problem rSP is sufficient to prove optimality of the solution or to provide a convex combination of columns for a strict improvement of the objective function. However in practice, several subsets of columns are selected at every iteration, either from cSP or from rSP, and sent to RrRMP or to RMP for an improvement of the current solution, by solving a linear program. This is possible because the compatibility notion is somewhat flexible. For example, one can include or not in the set of degenerate variables a variable that is at an implicit upper bound. In set partitioning models, this is the case for variables at value one.

Definition 2. *Given the solution vector* $\boldsymbol{\lambda}_F > \mathbf{0}$ *of non-degenerate variables, vector* $\mathbf{a}$ *is* compatible with row-set $Q \subseteq P$ *if and only if* $\bar{\mathbf{a}}_Z = \mathbf{0}$ *for* $Z := \{1, \ldots, m\} \setminus Q$.

This *row-compatibility* induces a two-step pivot procedure to enforce a strict improvement of the current solution. Firstly, a selected row-set $Q \subseteq P$ for RrRMP (still of row-size $p \leq m$) constrains the set of possible exiting variables from the current basis, the ratio-test being computed only for those $|Q| := q \leq p$ constraints. Secondly, a specialized pricing subproblem cSP selects an entering variable λ_j, $j \in N$, such that $\bar{c}_j < 0$ and $\bar{\mathbf{a}}_{Zj} = \mathbf{0}$, this zero vector being defined on $Z := \{1, \ldots, m\} \setminus Q$ according to the row-compatibility in use. This supports various implementation strategies. For example, one could temporarily restrain the search for entering variables to those for which Q is a strict subset of P and, for all $i \in Q$, $\bar{b}_i$ are relatively large. This can be considered as a partial pricing strategy and should accelerate the solution of RrRMP since only significant step sizes are expected.

Alternatively, when the entering variable is compatible with the solution $\boldsymbol{\lambda}_F$, the pivot is non-degenerate but the new basic solution may become degenerate, say $\boldsymbol{\lambda}_Q > \mathbf{0}$, $\boldsymbol{\lambda}_{N \setminus Q} = \mathbf{0}$, with $Q \subseteq P$. Then one can update or not the current row-partition of RrRMP, namely RrRMP$_P$. If it is updated, it becomes RrRMP$_Q$ with only q rows and $Z := \{1, \ldots, m\} \setminus Q$. If not, it should be pointed out that this also results in an exact algorithm as it still solves the transformed MP formulation in (6). In that case, the current compatibility rule with row-set P is simply maintained. No overhead computations are needed for an update of RrRMP$_P$ and the entering variables are still selected such that $\bar{\mathbf{a}}_{Zj} = \mathbf{0}$, where $Z := \{1, \ldots, m\} \setminus P$ is a strictly smaller set of rows than what would be required in an updated pricing subproblem. In that case, degenerate pivots may occur because of the degeneracy of some RrRMP$_P$ solutions.

5 Summary and Conclusions

Classical column generation works with a restricted master problem (RMP), that is, a subset of a model's variables that are dynamically added via a pricing subproblem SP. Like the simplex method, column generation is known to suffer from degeneracy. Inspired by recent successes in coping with degeneracy in the primal simplex method, we propose a *row-reduced column generation* method (RrCG). RrCG *exploits* degeneracy as the restricted master problem only has as many rows as there are positive basic variables.

Columns/variables are characterized as compatible or incompatible with respect to RrRMP. Compatible columns allow for a strict decrease of the objective function when entered into the basis, that is, a non-degenerate pivot. Two types of subproblems are proposed to generate variables: a specialized subproblem cSP for compatible variables, and rSP to price out any type of variables. The latter also needs to be solved by column generation. Pricing subproblem cSP is the original subproblem SP augmented with a set of linear constraints imposing compatibility requirements. It selects compatible columns as long as they are useful for non-degenerate pivots in RrRMP. When the reduced costs of all compatible columns are zero (or larger than a specified threshold), the pricing subproblem rSP is solved. It selects a convex combination of columns such that, again, the objective value strictly improves when they all enter into the current basis. In both cases, the row-size of RrRMP can be dynamically modified. The structure of the pricing problem rSP allows to derive a necessary and sufficient optimality condition for linear programs.

Decomposition in column generation for integer programs is based on the modeling structure of a compact formulation. It exploits the pricing subproblem for the selection of objects $\mathbf{x} \in X$. Row-partition in RrCG takes advantage of degenerate solutions, reduces the row-size of the master problem and its associated working basis, and thus, the computational effort for re-optimization. Combining column generation and dynamic row-partition during the solution process allows for exploiting both the modeling structure of its formulation and the algebraic structure of its solutions. Two special cases of RrCG are the *improved primal simplex* method and the *dynamic constraints aggregation* method for solving by column generation LP relaxations of set partitioning problems. On highly degenerate instances, recent computational experiments with these algorithms have shown a substantial impact on the solution time. This already opened the door to further research within that field, notably an integral simplex algorithm for the set partitioning problem (Zaghrouti et al. 2011) and a specialized version for the capacitated network flow problem which turns out to be strongly polynomial (Desrosiers et al. 2013).

As for any simplex or column generation algorithm, future work is needed on RrCG, mainly on implementation strategies. Amongst these are the moment for an update of the row-partition of RrRMP, efficient solvers for cSP and rSP, which pricing subproblem to call, whether to add many incompatible columns or their (single column) convex combination as returned from rSP, and much more on relaxed compatibility.

Acknowledgments. We would like to thank François Soumis from École Polytechnique de Montréal for giving us useful feedback on an earlier draft of this paper. This research has been supported by an NSERC (Canada) grant for the first author.

References

Ravindra K. Ahuja, Thomas L. Magnanti, and James B. Orlin. *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall, 1993.

Cynthia Barnhart, Ellis L. Johnson, George L. Nemhauser, Martin W. P. Savelsbergh, and Pamela H. Vance. Branch-and-price: Column generation for solving huge integer programs. *Operations Research*, 46(3): 316–329, 1998. doi: 10.1287/opre.46.3.316.

Pascal Benchimol, Guy Desaulniers, and Jacques Desrosiers. Stabilized dynamic constraint aggregation for solving set partitioning problems. *European Journal of Operational Research*, 223(2):360–371, 2012. doi: 10.1016/j.ejor.2012.07.004.

George B. Dantzig and Philip Wolfe. Decomposition principle for linear programs. *Operations Research*, 8 (1):101–111, 1960. doi: 10.1287/opre.8.1.101.

Guy Desaulniers, Jacques Desrosiers, Irina Ioachim, Marius M. Solomon, François Soumis, and Daniel Villeneuve. A unified framework for deterministic time constrained vehicle routing and crew scheduling problems. In T.G. Crainic and G. Laporte, editors, *Fleet Management and Logistics*, pages 57–93. Kluwer, 1998. doi: 10.1007/978-1-4615-5755-5_3.

Jacques Desrosiers, Jean Bertrand Gauthier, and Marco E. Lübbecke. A contraction-expansion algorithm for the capacitated minimum cost flow problem. Working paper, GERAD, Montréal, Canada, 2013.

Issmail Elhallaoui, Daniel Villeneuve, François Soumis, and Guy Desaulniers. Dynamic aggregation of set partitioning constraints in column generation. *Operations Research*, 53(4):632–645, 2005. doi: 10.1287/opre.1050.0222.

Issmail Elhallaoui, Guy Desaulniers, Abdelmoutalib Metrane, and François Soumis. Bi-dynamic constraint aggregation and subproblem reduction. *Computers & Operations Research*, 35(5):1713–1724, 2008. doi: 10.1016/j.cor.2006.10.007.

Issmail Elhallaoui, Abdelmoutalib Metrane, Guy Desaulniers, and François Soumis. An Improved Primal Simplex algorithm for degenerate linear programs. *INFORMS Journal on Computing*, 2010a. doi: 10.1287/ijoc.1100.0425.

Issmail Elhallaoui, Abdelmoutalib Metrane, François Soumis, and Guy Desaulniers. Multi-phase dynamic constraint aggregation for set partitioning type problems. *Mathematical Programming*, 123(2):345–370, 2010b. doi: 10.1007/s10107-008-0254-5.

Marco E. Lübbecke and Jacques Desrosiers. Selected topics in column generation. *Operations Research*, 53 (6):1007–1023, 2005. doi: 10.1287/opre.1050.0234.

P.-Q. Pan. A basis deficiency-allowing variation of the simplex method for linear programming. *Computers & Mathematics with Applications*, 36(3):33–53, 1998. doi: 10.1016/S0898-1221(98)00127-8.

Vincent Raymond, François Soumis, and Dominique Orban. A new version of the Improved Primal Simplex for degenerate linear programs. *Computers & Operations Research*, 37(1):91–98, 2010. doi: 10.1016/j.cor.2009.03.020.

Abdelouahab Zaghrouti, François Soumis, and Issmail El Hallaoui. Integral simplex using decomposition for the set partitioning problem. Technical report, GERAD, Montréal, Canada, 2011.

Appendix: Proofs

In this Appendix, we provide the proof of Proposition 3 followed by that of Proposition 2. Let us first express rSP (10) in terms of the variables $\boldsymbol{\lambda}_N$, and as a function of the working basis A_{PF} and the original cost and data coefficients:

$$\begin{array}{rrcl} \bar{c}^\star_{rSP} := \min & (\mathbf{c}^\intercal - \mathbf{c}_F^\intercal A_{PF}^{-1} A_P)\boldsymbol{\lambda}_N & & \\ \text{s.t.} & \mathbf{1}^\intercal \boldsymbol{\lambda}_N & = & 1 \\ & (A_Z - A_{ZF} A_{PF}^{-1} A_P)\boldsymbol{\lambda}_N & = & \mathbf{0} \\ & \boldsymbol{\lambda}_N & \geq & \mathbf{0}. \end{array}$$

Consider another working basis defined on row-set Q and denoted A_{QF}. It is again of dimension $p \times p$ and $\boldsymbol{\lambda}_F = A_{QF}^{-1}\mathbf{b}_Q = \bar{\mathbf{b}}_Q$. Duplicate rows in set Q such that MP becomes

$$\begin{array}{rrcrcl} z^\star_{MP} := \min & \mathbf{c}_F^\intercal \boldsymbol{\lambda}_F & + & \mathbf{c}_N^\intercal \boldsymbol{\lambda}_N & & \\ \text{s.t.} & A_{PF}\boldsymbol{\lambda}_F & + & A_{PN}\boldsymbol{\lambda}_N & = & \mathbf{b}_P \\ & A_{ZF}\boldsymbol{\lambda}_F & + & A_{ZN}\boldsymbol{\lambda}_N & = & \mathbf{b}_Z \\ & A_{QF}\boldsymbol{\lambda}_F & + & A_{QN}\boldsymbol{\lambda}_N & = & \mathbf{b}_Q \\ & \boldsymbol{\lambda}_F & , & \boldsymbol{\lambda}_N & \geq & \mathbf{0}. \end{array}$$

Given the vector of positive variables $\boldsymbol{\lambda}_F > \mathbf{0}$, consider the following two bases, the first being defined according to the working basis A_{PF} while the second is given according to the working basis A_{QF}:

$$\left[\begin{array}{c|cc} A_{PF} & \mathbf{0} & \mathbf{0} \\ \hline A_{ZF} & I_{m-p} & \mathbf{0} \\ A_{QF} & \mathbf{0} & I_p \end{array}\right] \quad \text{and} \quad \left[\begin{array}{c|cc} A_{QF} & \mathbf{0} & \mathbf{0} \\ \hline A_{ZF} & I_{m-p} & \mathbf{0} \\ A_{PF} & \mathbf{0} & I_p \end{array}\right].$$

The corresponding inverses are given as follows:

$$\left[\begin{array}{r|cc} A_{PF}^{-1} & \mathbf{0} & \mathbf{0} \\ \hline -A_{ZF}A_{PF}^{-1} & I_{m-p} & \mathbf{0} \\ -A_{QF}A_{PF}^{-1} & \mathbf{0} & I_p \end{array}\right] \quad \text{and} \quad \left[\begin{array}{r|cc} A_{QF}^{-1} & \mathbf{0} & \mathbf{0} \\ \hline -A_{ZF}A_{QF}^{-1} & I_{m-p} & \mathbf{0} \\ -A_{PF}A_{QF}^{-1} & \mathbf{0} & I_p \end{array}\right].$$

Pricing subproblem rSP can now be written according to the selected basis, namely rSP_P and rSP_Q:

$$\begin{array}{rrcl} \bar{c}^\star_{rSP_P} := \min & (\mathbf{c}^\intercal - \mathbf{c}_F^\intercal A_{PF}^{-1} A_{PN})\boldsymbol{\lambda}_N & & \\ \text{s.t.} & \mathbf{1}^\intercal \boldsymbol{\lambda}_N & = & 1 \\ & (A_{ZN} - A_{ZF} A_{PF}^{-1} A_{PN})\boldsymbol{\lambda}_N & = & \mathbf{0} \\ & (A_{QN} - A_{QF} A_{PF}^{-1} A_{PN})\boldsymbol{\lambda}_N & = & \mathbf{0} \\ & \boldsymbol{\lambda}_N & \geq & \mathbf{0} \end{array} \tag{20}$$

and

$$\begin{array}{rrcl} \bar{c}^\star_{rSP_Q} := \min & (\mathbf{c}^\intercal - \mathbf{c}_F^\intercal A_{QF}^{-1} A_{QN})\boldsymbol{\lambda}_N & & \\ \text{s.t.} & \mathbf{1}^\intercal \boldsymbol{\lambda}_N & = & 1 \\ & (A_{ZN} - A_{ZF} A_{QF}^{-1} A_{QN})\boldsymbol{\lambda}_N & = & \mathbf{0} \\ & (A_{PN} - A_{PF} A_{QF}^{-1} A_{QN})\boldsymbol{\lambda}_N & = & \mathbf{0} \\ & \boldsymbol{\lambda}_N & \geq & \mathbf{0}\ . \end{array} \tag{21}$$

Proposition 3. *Given two working bases A_{PF} and A_{QF} of A_F, the corresponding pricing subproblems* rSP_P *and* rSP_Q *are equivalent programs.*

Proof. We show that the solution set of rSP_P and rSP_Q is the same and the value of their objective functions is equal for any optimal convex combination $\boldsymbol{\lambda}_N^\star$. Left-multiplying the third set of constraints of (20) by $-A_{PF}A_{QF}^{-1}$, one obtains the third set of constraints of (21):

$$-A_{PF}A_{QF}^{-1}(A_{QN} - A_{QF}A_{PF}^{-1}A_{PN})\boldsymbol{\lambda}_N = (A_{PN} - A_{PF}A_{QF}^{-1}A_{QN})\boldsymbol{\lambda}_N = \mathbf{0}.$$

From the above equation, we have the identity $A_{PN}\boldsymbol{\lambda}_N = A_{PF}A_{QF}^{-1}A_{QN}\boldsymbol{\lambda}_N$. Substituting for $A_{PN}\boldsymbol{\lambda}_N$ in the objective function and in the second constraint set of (20), one completes the linear transformation of rSP_P into rSP_Q:

$$(\mathbf{c}^\intercal - \mathbf{c}_F^\intercal A_{PF}^{-1}A_{PN})\boldsymbol{\lambda}_N = \mathbf{c}^\intercal\boldsymbol{\lambda} - \mathbf{c}_F^\intercal A_{PF}^{-1}A_{PF}A_{QF}^{-1}A_{QN}\boldsymbol{\lambda}_N = (\mathbf{c}^\intercal - \mathbf{c}_F^\intercal A_{QF}^{-1}A_{QN})\boldsymbol{\lambda}_N$$

and

$$(A_{ZN} - A_{ZF}A_{PF}^{-1}A_{PN})\boldsymbol{\lambda}_N = A_{ZN}\boldsymbol{\lambda}_N - A_{ZF}A_{PF}^{-1}A_{PF}A_{QF}^{-1}A_{QN}\boldsymbol{\lambda}_N = (A_{ZN} - A_{ZF}A_{QF}^{-1}A_{QN})\boldsymbol{\lambda}_N. \quad \square$$

We next show that the reduced costs of the compatible variables are indeed identical in the two versions of cSP, namely cSP_P and cSP_Q and their coefficient components are equivalent. Therefore cSP, the specialized pricing subproblem for generating compatible variables, is independent of the selected working basis. We first provide these two versions of cSP, adapted from (11):

$$\begin{aligned} \bar{c}^\star_{cSP_P} := \min_{\mathbf{x}\in X} \quad & c(\mathbf{x}) - \mathbf{c}_F^\intercal A_{PF}^{-1}\mathbf{a}_P(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{a}_Z(\mathbf{x}) - A_{ZF}A_{PF}^{-1}\mathbf{a}_P(\mathbf{x}) = \mathbf{0} \\ & \mathbf{a}_Q(\mathbf{x}) - A_{QF}A_{PF}^{-1}\mathbf{a}_P(\mathbf{x}) = \mathbf{0} \end{aligned} \tag{22}$$

and

$$\begin{aligned} \bar{c}^\star_{cSP_Q} := \min_{\mathbf{x}\in X} \quad & c(\mathbf{x}) - \mathbf{c}_F^\intercal A_{QF}^{-1}\mathbf{a}_Q(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{a}_Z(\mathbf{x}) - A_{ZF}A_{QF}^{-1}\mathbf{a}_Q(\mathbf{x}) = \mathbf{0} \\ & \mathbf{a}_P(\mathbf{x}) - A_{PF}A_{PF}^{-1}\mathbf{a}_Q(\mathbf{x}) = \mathbf{0}. \end{aligned} \tag{23}$$

Proposition 2. *Given two working bases A_{PF} and A_{QF} of A_F, the corresponding pricing subproblems* cSP_P *and* cSP_Q *are equivalent programs.*

Proof. The proof is similar to that of Proposition 3 except that vector $\boldsymbol{\lambda}_N$ is not involved. Multiplying the last set of constraints of (22) by $-A_{PF}A_{QF}^{-1}$, one obtains the corresponding constraint set of (23). Since $\mathbf{a}_P(\mathbf{x}) = A_{PF}A_{PF}^{-1}\mathbf{a}_Q(\mathbf{x})$, it can be substituted in the objective function and the first set of constraints of (22). This shows that cSP_P and cSP_Q are equivalent programs. $\square$