

Converging upon Basic Feasible Solutions through Dantzig–Wolfe Decomposition

Joseph L. Rios · Kevin Ross

the date of receipt and acceptance should be inserted later

Abstract We derive an important property for solving large-scale integer programs by examining the master problem in Dantzig–Wolfe decomposition. In particular, we prove that if a linear program can be divided into subproblems with affinely independent corner points, then there is a direct mapping between basic feasible solutions in the master and original problems. This has implications for integer programs where the feasible region has integer corner points, ensuring that integer solutions to the original problem will be found even through the decomposition approach. An application to air traffic flow scheduling, which motivated this result, is highlighted.

Keywords Integer Programming, Parallel Algorithms

1 Introduction

Large-scale integer programs arise in several contexts, including transportation scheduling. Solving even the linear programming (LP) relaxation of these problems may be computationally expensive and require decomposition to solve in reasonable time, but such approaches have few guarantees with respect to integrality. In particular Dantzig–Wolfe (DW) decomposition benefits from both improved memory utilization and inherent parallelization for linear programs of the correct form, but the optimal solution found is not necessarily basic. An optimal, basic solution is important for integer programs whose formulation is such that basic solutions imply integer solutions. When that is the case, a basic solution would obviate the need for any further integer optimization.

In Goncalves (1968), it was shown that a basic feasible solution found by DW can be trivially mapped to a basic feasible solution of the original problem if it satisfies certain properties related to its basic variables and the degeneracy

of the subproblems. Here we show that affine independence of the columns is sufficient to satisfy this criterion, and that Binary Integer Programs (BIP) can always be decomposed to satisfy this.

Others have studied how to find basic solutions from interior optimal solutions in Megiddo (1991) and Bixby and Saltzman (1994), but to our knowledge this is the first application of its kind to DW decomposition. One application of DW decomposition to integer programming is Vanderbeck (2010), which integrates a branching scheme based on a pricing oracle in order to find integer solutions, in contrast to our approach which looks to identify when DW can find integer solutions directly. A more general application of column generation for integer programming is provided by Barnhart et al (1998).

A formulation of the Traffic Flow Management Problem in air traffic control provided by Bertsimas and Patterson (1998) is such an example of a strongly-formulated model that benefits from the results presented here. In fact, the strong performance with respect to both solution quality and run time of the application of DW decomposition to this problem was highlighted in Rios and Ross (2010), which motivated the result presented here.

2 Dantzig-Wolfe Decomposition

DW decomposition is a method for solving LPs of block-angular form:

$$\text{Minimize} \quad \mathbf{c}'_1 \mathbf{x}_1 + \mathbf{c}'_2 \mathbf{x}_2 + \dots + \mathbf{c}'_l \mathbf{x}_l \quad (1)$$

$$\text{subject to} \quad D_1 \mathbf{x}_1 + D_2 \mathbf{x}_2 + \dots + D_l \mathbf{x}_l = \mathbf{b}_0 \quad (2)$$

$$F_i \mathbf{x}_i = \mathbf{b}_i, \mathbf{x}_i \geq 0, i \in \{1, 2, \dots, l\} \quad (3)$$

where the \mathbf{x} , \mathbf{b} , and \mathbf{c} vectors are of appropriate dimension.

Constraints (2) are called *connecting constraints* since they involve variables from the independent sets given by (3). Denoting P_i to be the solutions to (3), having extreme points $\{\mathbf{x}_i^j\}_{j \in J_i}$, each feasible \mathbf{x}_i in P_i can be represented as a convex combination of $\{\mathbf{x}_i^j\}_{j \in J_i}$.

By substitution we can then rewrite the above LP as the *full master program*:

$$\begin{aligned} \text{Minimize} \quad & \sum_{i=1}^l \sum_{j \in J_i} \lambda_i^j \mathbf{c}'_i \mathbf{x}_i^j \\ \text{subject to} \quad & \sum_{i=1}^l \sum_{j \in J_i} \lambda_i^j D_i \mathbf{x}_i^j = \mathbf{b}_0 \\ & \sum_{j \in J_i} \lambda_i^j = 1, \lambda_i^j \geq 0, i \in \{1, 2, \dots, l\} \end{aligned}$$

where $\{\lambda_i^j\}_{j \in J_i, i \in [1, \dots, l]}$ are now the decision variables.

Since the vast majority of λ_i^j variables are valued zero at any given iteration, most *columns* $D\mathbf{x}_i^j$ are irrelevant to the master, which motivates *column generation*. For each i , an independent LP is created, known as a *subproblem*:

$$\begin{array}{ll} \text{Minimize} & (\mathbf{c}'_i - \mathbf{q}'D_i)\mathbf{x}_i \\ \text{subject to} & F_i\mathbf{x}_i = \mathbf{b}_i, x_i \geq 0 \end{array}$$

where \mathbf{q} is the dual variable vector associated with the connecting constraints, made available from the master problem. The master problem, in essence, *asks* the subproblem for a variable (column) that will improve the master’s objective function. The subproblem provides a variable \mathbf{x}_i with negative reduced cost whenever that is possible. The master iterates over all subproblems until its objective can no longer be improved, implying an optimal solution has been found.

A full description of DW can be found in Dantzig and Wolfe (1960) and Bertsimas and Tsitsiklis (1997), with computational issues highlighted in Ho (1987); Ho and Louie (1981) and Tebboth (2001).

3 Mapping Basic Feasible Solutions

When an optimal solution to the original LP exists, DW will always converge to an optimal solution (as shown by Dantzig and Wolfe (1960)), but it may not be a basic feasible solution to the original problem. The only criterion of which we are aware that relates the basic feasible solutions of the master and original problems was presented by Goncalves (1968), a result which has not been frequently utilized. We state this criterion here with more commonly accepted notation.

Let T be a solution to the full master problem with N non-zero λ_i^j values. For each subproblem i , define $P_i^0 = \sum_{j \in J_i} \lambda_i^j x_i^j$ to be the component of T that corresponds to a feasible solution of subproblem i . Let $F_i^{d_i}$ be the face of smallest dimension (d_i) that contains P_i^0 in the respective problem’s feasible space. Define the *degree of degeneracy* of a basic solution β to be the number of zero-valued basic variables representing that solution, denoted $\mathcal{D}(\beta)$. A basic feasible solution to the full master program with l subproblems maps to a basic feasible solution of the original program if and only if:

$$\mathcal{D}(T) + N - l + \sum_{i=1}^l \mathcal{D}(F_i^{d_i}) - \sum_{i=1}^l d_i \geq 0. \quad (4)$$

This was discovered and proven by Goncalves (1968). We will refer to this inequality as the *Goncalves Criterion*. Assuming that all variables from the original problem are present in exactly one subproblem (always enforceable), then $\mathcal{D}(T) + N - l$ is the number of connecting constraints in the master problem and $\mathcal{D}(T) + N$ is the number of basic variables at any iteration of the DW algorithm. A full description is found in Goncalves (1968), and the

correspondence between Appa (1969) and Goncalves (1969) corrects a small error and argues the theoretical utility of this result.

4 An Example of the Goncalves Criterion

The intuition behind the Goncalves Criterion is potentially muddled by the necessary notation. For that reason, we provide an example exercising many aspects of it using a trivial linear program. Consider the following LP:

$$\text{Minimize } -x_1 - x_2 \quad (5)$$

$$\text{subject to } x_1 + x_2 = 1 \quad (6)$$

$$x_1 \leq 1 \quad (7)$$

$$x_2 \leq 1 \quad (8)$$

$$x_i \geq 0 \quad \forall i \in \{1, 2\}.$$

Note that the optimal objective value is -1 and is achieved by an infinite number of solutions satisfying constraints 6-8. We will examine two paths the DW algorithm might take through a decomposition of this problem and show their respective relationships to the Goncalves Criterion.

First, consider a single subproblem for DW decomposition created by using both constraints 7 and 8. The feasible region of that subproblem would be the unit square in the positive quadrant. The key matrices (using the notation from Section 2) are $D_1 = \begin{bmatrix} 1 & 1 \end{bmatrix}$ and $F_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$. Given this F_1 , the four corresponding \mathbf{x}_1^j values can be labeled as follows:

$$\mathbf{x}_1^1 = (0, 0)$$

$$\mathbf{x}_1^2 = (0, 1)$$

$$\mathbf{x}_1^3 = (1, 0)$$

$$\mathbf{x}_1^4 = (1, 1).$$

Assume that through a ‘‘Phase 1’’ procedure, the first two solutions provided to the reduced master problem to form a basis are \mathbf{x}_1 and \mathbf{x}_4 (we need two since there are two constraints in the reduced master: one for the connecting constraint and one of the subproblem convexity constraint). Forming the products \mathbf{x}_1^1 and \mathbf{x}_1^4 with D_1 provides the single-valued columns (recall because we only have one connecting constraint 6) $D_1\mathbf{x}_1^1 = 0$ and $D_1\mathbf{x}_1^4 = 2$, respectively. The reduced master will assign weights for λ_1^1 and λ_1^4 of 0.5 and 0.5, respectively, to obtain an optimal solution (i.e., $(x_1, x_2) = (0.5, 0.5)$). Given all of these values, we can calculate the various GC parameters:

$$\mathcal{D}(T) = 0 \quad (\text{reduced master solution is non-degenerate})$$

$$N = 2 \quad (\text{number of non-zero columns in reduced master})$$

$$l = 1 \quad (\text{number of subproblems})$$

$$d_1 = 2 \quad (\text{solution maps to center of } F_1, \text{ a 2-D space})$$

$$\mathcal{D}(F_1^{d_1}) = 0 \quad (\text{no constraints in } F_1 \text{ are at equality}).$$

Now checking the GC with these parameters we see that

$$\mathcal{D}(T) + N - l + \sum_{i=1}^l \mathcal{D}(F_i^{d_i}) - \sum_{i=1}^l d_i = 0 + 2 - 1 + 0 - 2 \not\geq 0$$

and, therefore, this solution does *not* map to a basic solution of the original problem. Note that this is despite the fact that the solution is optimal in the original problem. This is also an example of how DW can converge to non-basic (though optimal) solutions of the original problem and might complicate integer optimization problems. For instance, if we added an integrality constraint to x_1 we would still have work to do if provided this solution from DW.

Now consider the same decomposition as described above, however for an initial basis was are provided with \mathbf{x}_1 and \mathbf{x}_2 . The columns (λ_1^1 and λ_1^2) of the reduced master will then be $D_1\mathbf{x}_1^1 = 0$ and $D_1\mathbf{x}_1^2 = 1$, respectively. That will force values of $\lambda_1^1 = 0$ and $\lambda_1^2 = 1$ for and \mathbf{x} value of $(0, 1)$, which is an optimal solution to the original problem. We can again calculate the GC parameters:

$$\mathcal{D}(T) = 1 \quad (\text{reduced master is degenerate: one zero-valued basic variable})$$

$$N = 1 \quad (\text{number of non-zero columns in reduced master})$$

$$l = 1 \quad (\text{number of subproblems})$$

$$d_1 = 0 \quad (\text{solution maps to a corner point (a 0-D space) of } F_1)$$

$$\mathcal{D}(F_1^{d_1}) = 1 \quad (\text{one constraint in } F_1 \text{ is at equality}).$$

Again checking the GC with these new parameters we see

$$\mathcal{D}(T) + N - l + \sum_{i=1}^l \mathcal{D}(F_i^{d_i}) - \sum_{i=1}^l d_i = 1 + 1 - 1 + 1 - 0 \geq 0$$

and, therefore, this solution maps to a basic solution of the original problem. This is easily seen since this solution in the original problem maps to a corner point of its feasible space.

5 Guaranteeing Satisfaction of the Goncalves Criterion

We now state the main results of this paper, which build upon the Goncalves Criterion.

Theorem 1 *Any decomposition of a linear program in block-angular form such that each subproblem has affinely independent corner points will satisfy the Goncalves Criterion at every iteration of the Dantzig–Wolfe decomposition algorithm.*

Proof Let N_i represent the number of non-zero λ_i^j in T that correspond to columns from subproblem i . Since the corner points of each P_i are affinely independent, $P_i^0 = \sum_{j \in J_i} \lambda_i^j x_i^j$ must be found on a face of dimension $d_i =$

$N_i - 1$ (see discussion of affine frames in Gallier (1999), for example). Since $\mathcal{D}(T)$ and $\mathcal{D}(F_i^{d_i})$ will always be non-negative by definition,

$$\mathcal{D}(T) + N - l + \sum_{i=1}^l \mathcal{D}(F_i^{d_i}) - \sum_{i=1}^l d_i \geq \sum_{i=1}^l (N_i - 1 - d_i) = 0$$

and the Goncalves Criterion is satisfied. \square

Theorem 2 *The linear relaxation of any Binary Integer Program (BIP) can be decomposed into subproblems with affinely independent corner points.*

Proof The linear relaxation of any BIP includes the constraints $0 \leq x_i \leq 1$. Therefore one (trivial) decomposition is constructed by placing all constraints into the set of connecting constraints except for the $0 \leq x_i \leq 1$ constraints. By placing exactly one of these into each separate subproblem, the conditions of Theorem 1 are satisfied since the only feasible corner points to any subproblem are $x_i \in \{0, 1\}$, which are affinely independent. \square

Through Theorems 1 and 2 it is clear that there is at least a trivial decomposition to any BIP that will allow for satisfaction of the Goncalves Criterion at every DW iteration. Of course by itself this is not particularly helpful, as the full decomposition is equivalent to solving the problem without any decomposition at all. However, we also observe the following:

Observation 1 *If a problem is formulated for DW decomposition and any one of the subproblems is further subdivided into two or more subproblems, then the left-hand side of Inequality (4) will often increase, and almost never decrease.*

This is true because the combination of the first three terms of Inequality (4), $\mathcal{D}(T) + N - l$, gives the number of connecting constraints, which is nondecreasing as further subdivisions are completed. Each subdivision requires some constraints which were previously in a single subproblem to be moved into the connecting constraints, which increases this value. It cannot be strictly guaranteed that the remaining terms are nondecreasing, but the tighter constraints of each subproblem coupled with their decreased dimensionality generally lead to lower values for d_i .

Combining Theorems 1 and 2 with Observation 1, the following method will guarantee to eventually find an optimal basic feasible solution of any BIP, and often find it without needing to trivially decompose the problem.

1. *Decompose the LP relaxation using DW decomposition and solve.*
2. *If the DW solution is basic in the original problem, then STOP.*
3. *Otherwise, select a subproblem with non-basic P_i^0 , subdivide that problem further, and repeat.*

Because of Theorem 2, this will eventually find a basic solution for a BIP. If the problem is totally unimodular, then this will also be an integer solution. This was observed in particular for the air traffic scheduling problem described

next which, while not totally unimodular, is known to have mostly integer basic feasible solutions (see Bertsimas and Patterson (1998)). In Rios and Ross (2010) it was seen that the procedure described above did in practice find an optimal solution without needing to fully decompose the problem. This resulted in a superior solution quality and run-time compared to any other known method of solution.

6 An Example from Air Traffic Management

We now provide an example from a transportation sub-domain, but we will re-iterate that the application of DW using the insights described in Section 5 will apply to other applications just as well. This is especially true for models like the one we are about to present that have binary decision variables and are strongly formulated such that most corner points are integer-valued. These properties help leverage the results of Theorems 1 and 2.

The key criteria In Air Traffic Management, the demand and capacity balance of airports and en route airspace must be carefully managed. Bertsimas and Patterson (1998) introduced the following BIP to describe the core scheduling problem:

$$\begin{aligned} \text{Minimize} \quad & \sum_f \left[(c_f^g - c_f^a) \sum_{t,k=P(f,1)} t(w_{f,t}^k - w_{f,t-1}^k) \right. \\ & \left. + c_f^a \sum_{t,k=P(f,\text{last})} t(w_{f,t}^k - w_{f,t-1}^k) + (c_f^a - c_f^g)d_f - c_f^a r_f \right] \end{aligned} \quad (9)$$

subject to

$$\sum_{f:P(f,1)=k} (w_{f,t}^k - w_{f,t-1}^k) \leq D_k(t), \quad \forall k \in \text{Airports}, t \in \text{Time} \quad (10)$$

$$\sum_{f:P(f,\text{last})=k} (w_{f,t}^k - w_{f,t-1}^k) \leq A_k(t), \quad \forall k \in \text{Airports}, t \in \text{Time} \quad (11)$$

$$\sum_{f:P(f,i)=j,P(f,i+1)=j'} (w_{f,t}^j - w_{f,t}^{j'}) \leq S_j(t), \quad \forall j \in \text{Sectors}, t \in \text{Time} \quad (12)$$

$$w_{f,t+\mu(f,j)}^{j'} - w_{f,t}^j \leq 0, \quad \forall f \in \mathcal{F}, j = P(f,i), j' = P(f,i+1) \quad (13)$$

$$w_{f,t}^j - w_{f,t-1}^j \geq 0, \quad \forall f \in \mathcal{F}, j \in (f\text{'s flight path}) \quad (14)$$

$$w_{f,t}^j \in \{0, 1\}, \quad \forall f \in \mathcal{F}, j \in (f\text{'s flight path}), t \in \text{Time} \quad (15)$$

The decision variables $w_{f,t}^j$ are valued 1 if and only if a flight f has entered sector j by time t (14). The objective (9) is to minimize the weighted sum of ground and air delay costs, subject to airport departure (10), airport arrival (11), and sector capacity constraints (12). Each flight is described as an ordered list of distinct sectors and airports constrained by earliest and latest feasible time steps as well as physical and temporal limitations of the flights in (13) and (14), respectively.

Real-world instances (thousands of flights, hundreds of sectors, airports, and time steps) of this model are computationally intense. Fortunately, the LP relaxation is in block diagonal form, which suggests the application of DW decomposition, first presented in Rios and Ross (2010). It is also known to have mostly integer corner points (see Bertsimas and Patterson (1998)), which makes it a good candidate for application of the approach described in this paper.

It can be shown that the solutions satisfying (13) and (14) have affinely independent corner points for each flight, f . By theorem 1, this means that this particular BIP will find basic solutions through DW decomposition without needing to fully decompose the problem to the trivial case described in the proof of that theorem. Following the algorithm described in section 3 and to demonstrate the utility of the approach, the problem was decomposed into successively finer subproblems. At the end of each decomposition, a good integerization approach (described as the *choose-one* heuristic in Rios and Ross (2010)) was used to find the nearest integer solution to the optimal solution found through DW decomposition, with the *optimality gap* corresponding to the difference in objective value between the optimal and corresponding integer solutions.

Figure 1 shows the results in terms of this optimality gap (on the left) and the run time (on the right). Both measures improve as the number of subproblems increases, until the optimality gap reaches zero. The maximal decomposition required was 5920 subproblems, which was the number of flights in the scenario, at which stage constraints (13) and (14) relating to an individual flight were placed into individual subproblems. Note that this is significantly less than the 654,300 variables (all binary), which shows that the trivial decomposition described in theorem 2 is not necessary to reach the optimal integer solution. This result is especially significant because it provides a method for minimizing delays in the United States national airspace which can be operated in real-time. This is in contrast to previous methods which either could not guarantee optimality or could not solve within the window of time being scheduled (e.g., a scenario with a two-hour planning horizon could take over two hours to solve).

The reasons for the improvements in runtime and solution quality presented in Figure 1 are not completely obvious, but are owed to several factors. One factor includes the properties described in this paper for obtaining basic solutions through properly decomposed subproblems. Another important factor for the improved runtime is the fact that the decreasing size of the subproblems make them increasingly trivial for modern linear programming software

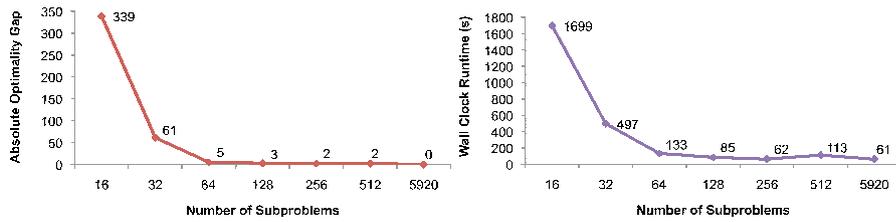


Fig. 1 Quality and speed over an increasing number of subproblems. Both the optimality gap (on the left) and the runtime (on the right) improve as the problem is subdivided into more subproblems. For this air traffic example, the optimality gap goes to zero with 5920 subproblems

to solve. Coupled with the availability of computer systems with multiple cores, the runtime decreases dramatically despite having to solve many more subproblems.

7 Discussion and Conclusion

We have presented a sufficient condition for consistently mapping basic solutions from a Dantzig–Wolfe decomposition back to basic solutions of the original linear program. By ensuring affine independence of the corner points in the subproblems, one can guarantee a direct correspondence. We have shown that any BIP can be decomposed to satisfy this condition, and computational results on a large-scale BIP indicate that integer solutions may be found even using a less refined decomposition than necessary. In fact, the computational results hint that, in practice, *near* affine independence will aid in ensuring a mapping exists between basic solutions in the master and original problems. When integer programs are strongly formulated, this result shows that the positive qualities of the original program in terms of integer corner points will not be lost when performing Dantzig–Wolfe decomposition when the subproblems are carefully constructed.

There are several fertile directions for future research in this area. A fundamental question to which a mathematical modeler might want a clear answer is when to employ a technique as described here versus a branch-and-price approach. Assuming there is no free lunch, the answer will depend greatly on the models being used. A large-scale computational study comparing implementations of DW and branch-and-price would likely shed light on scenarios more befitting the DW approach described here over branch-and-price, and vice-versa. Another interesting extension to this work would be to determine what level of decomposition to apply a priori to achieve results similar to those presented in Section 6. Currently that decision would be more art than science, relying on the experience and insight of the user.

References

- Appa G (1969) Dantzig-wolfe decomposition algorithm. *Operational Research Quarterly* 20(2):275
- Barnhart C, Johnson EL, Nemhauser GL, Savelsbergh MWP, Vance PH (1998) Branch-and-price: Column generation for solving huge integer programs. *Operations Research* 46(3):316–329
- Bertsimas D, Patterson SS (1998) The air traffic flow management problem with enroute capacities. *Operations Research* 46(3):406–422
- Bertsimas D, Tsitsiklis JN (1997) *Introduction to Linear Optimization*. Athena Scientific, Belmont, Massachusetts
- Bixby RE, Saltzman MJ (1994) Recovering an optimal basis from an interior point solution. *Operations Research Letters* 15:169–178
- Dantzig GB, Wolfe P (1960) Decomposition principle for linear programs. *Operations Research* 8(1):101–111
- Gallier JH (1999) *Curves and Surfaces in Geometric Modeling: Theory and Algorithms*. Morgan Kaufmann
- Goncalves A (1968) Basic feasible solutions and the dantzig-wolfe decomposition algorithm. *Operational Research Quarterly* 19(4):465–469
- Goncalves A (1969) Mr. goncalves replies. *Operational Research Quarterly* 20(2):276
- Ho JK (1987) Recent advances in the decomposition approach to linear programming. *Mathematical Programming Study* 31:119–127
- Ho JK, Loute E (1981) An advanced implementation of the dantzig-wolfe decomposition algorithm for linear programming. *Mathematical Programming* 20(1):303–326
- Megiddo N (1991) On finding primal- and dual-optimal bases. *ORSA Journal on Computing* 3(1):63–65
- Rios J, Ross K (2010) Massively parallel dantzig-wolfe decomposition applied to traffic flow scheduling. *Journal of Aerospace Computing, Information, and Communication* 7(1):32–45
- Tebboth JR (2001) A computational study of dantzig-wolfe decomposition. PhD thesis, University of Buckingham
- Vanderbeck F (2010) Branching in branch-and-price: a generic scheme. *Mathematical Programming* pp 1–46