A Computational Investigation on the Strength of Dantzig-Wolfe Reformulations

Michael Bastubbe
Lehrstuhl für Operations Research, RWTH Aachen University,
Kackertstr. 7, D-52072 Aachen, Germany
michael.bastubbe@rwth-aachen.de

Marco E. Lübbecke
Lehrstuhl für Operations Research, RWTH Aachen University,
Kackertstr. 7, D-52072 Aachen, Germany
marco.luebbecke@rwth-aachen.de

Jonas T. Witt
Lehrstuhl für Operations Research, RWTH Aachen University,
Kackertstr. 7, D-52072 Aachen, Germany
jonas.witt@rwth-aachen.de

Abstract

In Dantzig-Wolfe reformulation of an integer program one convexifies a subset of the constraints, leading to potentially stronger dual bounds from the respective linear programming relaxation. As the subset can be chosen arbitrarily, this includes the trivial cases of convexifying no and all constraints, resulting in a weakest and strongest reformulation, respectively. Our computational study aims at better understanding of what happens in between these extremes. For a collection of integer programs with few constraints we compute, optimally solve, and evaluate the relaxations of all possible (exponentially many) Dantzig-Wolfe reformulations (with mild extensions to larger models from the MIPLIBs). We observe that only a tiny number of different dual bounds actually occur and that only a few inclusion-wise minimal representatives exist for each. This aligns with considerably different impacts of individual constraints on the strengthening the relaxation, some of which have almost no influence. In contrast, types of constraints that are convexified in textbook reformulations have a larger effect. We relate our experiments to what could be called a hierarchy of Dantzig-Wolfe reformulations.

2012 ACM Subject Classification Mathematics of computing → Integer programming

Keywords and phrases Dantzig-Wolfe reformulation, strength of reformulations, Lagrangean relaxation, partial convexification, column generation, hierarchy of relaxations

Digital Object Identifier 10.4230/LIPIcs.SEA.2018.11

1 Motivation

The strength of formulations is a central topic in integer programming, and expressed via the quality of the dual bound obtained from the respective linear programming relaxation. It is well-known that a Dantzig-Wolfe (DW) reformulation of an integer program, the convexification of a subset of the constraints, may yield strong dual bounds. Therefore, such reformulations have been proposed in the literature for many models stemming from various applications. Even though a DW reformulation follows a certain mechanics that exploits the structure of the model, this is by far not unique. Technically, *every subset* of constraints gives rise to its own DW reformulation, including the two trivial cases: convexifying no or all constraints, implying the weakest and strongest possible dual bounds, respectively.

17th International Symposium on Experimental Algorithms (SEA 2018).
Editor: Gianlorenzo D'Angelo; Article No. 11; pp. 11:1–11:12
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Because of this freedom, speaking of "the" strength of DW reformulations in general is not useful. Instead, a differentiation is necessary, but theoretical results are very scarce (we mention two exceptions later). In order to make progress on this topic our study provides some general computational intuition. To the best of our knowledge, it is the first of its kind.

Brief Background on Dantzig-Wolfe Reformulation

We sketch the basics of DW reformulating an integer linear program (including the mixed-integer case), mainly to introduce our notation and conventions. We consider

$$\begin{aligned} z_{IP} = \min \quad & c^T x \\ \text{s.t.} \quad & a_i^T x \geq b_i \quad \forall i \in I \\ & x \in \mathbb{Z}^n \ , \end{aligned} \tag{1}$$

where I denotes a finite index set, $n \in \mathbb{Z}_{>0}$, $c, a_i \in \mathbb{Q}^n$, and $b_i \in \mathbb{Q}$ for $i \in I$. We identify constraints with their respective index $i \in I$. When we speak of relaxations, we always refer to the linear programming relaxation, obtained by dropping the integrality requirement on the variables. We denote the optimum of the relaxation of (1) by z_{LP}. The DW reformulation ("convexification") of a subset $I' \subseteq I$ of constraints amounts to (implicitly) additionally require in (1) that $x \in \operatorname{conv}\left\{\tilde{x} \in \mathbb{Z}^n : a_i^T \tilde{x} \geq b_i \ \forall i \in I'\right\}$. It is irrelevant here (but very well understood) how this is technically achieved (see Vanderbeck [13] for details on Dantzig-Wolfe reformulation and the relation to Lagrangean relaxation). This reformulation has the same integer feasible solutions as (1), but the relaxation

$$\begin{aligned} z_{DW}(I') = \min \quad & c^T x \\ \text{s.t.} \quad & a_i^T x \geq b_i \quad \forall i \in I \setminus I' \\ & x \in \operatorname{conv}\left\{\tilde{x} \in \mathbb{Z}^n : a_i^T \tilde{x} \geq b_i \ \forall i \in I'\right\} \\ & x \in \mathbb{R}^n \end{aligned} \tag{2}$$

is potentially stronger than that of (1), i.e.,

$$z_{IP} \geq z_{DW}(I') \geq z_{LP} \quad \forall I' \subseteq I \ . \tag{3}$$

This relation is a main reason for performing a DW reformulation in the first place. For convenience we identify a DW reformulation of constraints $I' \subseteq I$ with I' itself. We formally repeat that both extreme cases $z_{IP} = z_{DW}(I)$ and $z_{LP} = z_{DW}(\emptyset)$ are possible. Therefore, the notion of strength of a DW reformulation must necessarily relate to I' [14]. Geoffrion [7] gave as necessary condition for $z_{DW}(I') \gneq z_{LP}$ that $\operatorname{conv}\left\{\tilde{x} \in \mathbb{Z}^n : a_i^T \tilde{x} \geq b_i \ \forall i \in I'\right\} \subsetneq \left\{\tilde{x} \in \mathbb{R}^n : a_i^T \tilde{x} \geq b_i \ \forall i \in I'\right\}$. The condition is not sufficient; in particular, the actual strengthening may depend on the objective function. For the stable set (and related) problems we recently characterized DW reformulations $I' \subseteq I$ for which $z_{IP} = z_{DW}(I')$ or $z_{LP} = z_{DW}(I')$ independently of the objective function [14]. A generalization does not seem to be in reach and nothing is known about what happens "in between."

Our Approach

For each instance, taken from a set of small models of different problem classes, we compute the dual bounds $z_{DW}(I')$ from all $2^{|I|}$ DW reformulations $I' \subseteq I$ and collect some statistics about the respective DW reformulations. We then report these statistics for each problem class. When we plot figures, this is usually only for one representative of each class, because they look similar for the other problems of this class.

2 Instances and Experimental Setup

In order to keep the task of evaluating *all* (exponentially many) DW reformulations of an integer program manageable, we only consider very small models with up to 18 constraints. We first consider instances of problems where DW reformulation classically applies well (we call these models *structured*): bin packing (bpp), vertex coloring, capacitated p-median (cpmp), single-source capacitated facility location (cflp), generalized assigment (gap), and capacitated vehicle routing problems with time windows (vrp) problems. We used an instance generator to create small bin packing problems [12] and created 3 vertex coloring instances on connected graphs with 2 or 3 vertices (yielding integer programs with up to 15 constraints). Furthermore, we created 2 instances for the capacitated vehicle routing problem with time windows consisting of a single depot, 2 customers, and 2 vehicles (yielding mixed integer programs with up to 18 constraints). For all other problems, we took instances from the literature [4, 5, 8, 10, 11] and modified them to reduce the number of constraints. We chose standard textbook formulations for all problems with as few constraints as possible (e.g., by using formulations with few but relatively weak coupling constraints). Additionally, we consider very small and easy-to-solve instances from MIPLIB 3 [3], namely flugpl, mod008, and p0033. Although the instances markshare1, markshare2, mas74, and mas76 have only a small number of constraints, they turned out to be too hard to solve in preliminary experiments. All our instances have a positive integrality gap, i.e., $z_{LP} < z_{IP}$.

For the experiments, we use a development version of the generic branch-price-and-cut solver GCG [6], see `www.or.rwth-aachen.de/gcg` for the current released version 2.1.4. We implemented a so-called detector that creates, for each instance, all possible DW reformulations $I' \subseteq I$ and solve their relaxations optimally by column generation to obtain $z_{DW}(I')$. We turned off separation of cutting planes as well as the internal handling of problems with integral optimum (this would lead to only integer dual bounds $\lceil z_{DW}(I') \rceil$ for $I' \subseteq I$). For the structured models we also disabled presolving. The MIPLIB instances were presolved, i.e., the number of constraints and variables can differ from the original instance.

The total computation time spend for optimally solving over one million relaxations by column generation was approximately 100 hours. These numbers do not include the time spent for creating the decompositions and evaluating the computations.

3 Number and Frequency of Distinct Dual Bounds

This section is motivated by hierarchies of relaxations which are, roughly speaking, finite chains of (nested) stronger and stronger relaxations (of the same type), starting from the linear programming relaxation and arriving at the convex hull of integer feasible solutions. Several of these hierarchies are known, e.g., the Chvátal-Gomory procedure produces one.

Let $\mathcal{I} = 2^I$ denote the powerset of I. Consider the partially ordered set $P = (\mathcal{I}, \subseteq)$ consisting of all $2^{|I|}$ DW reformulations and their partial order induced by set inclusion. The empty set $\emptyset$ is the unique minimal element and I is the unique maximal element of P. Every chain $\emptyset = I_0 \subseteq I_1 \subseteq \cdots \subseteq I_m = I$ in P obviously defines a hierarchy of relaxations with $z_{LP} = z_{DW}(I_0) \leq z_{DW}(I_1) \leq \cdots \leq z_{DW}(I_m) = z_{IP}$, where the inequalities need not be (all) strict. That is, a chain may induce fewer than $|I| + 1$ dual bounds. We call a DW reformulation I' *minimal* if there does not exist any $I'' \subsetneq I'$ with $z_{DW}(I'') = z_{DW}(I')$.

Our first experiment reveals for every given instance how many distinct dual bounds actually occur and in what frequencies. In particular, what is the distribution of dual bounds in $[z_{LP}, z_{IP}]$ and what can we learn about minimal DW reformulations?

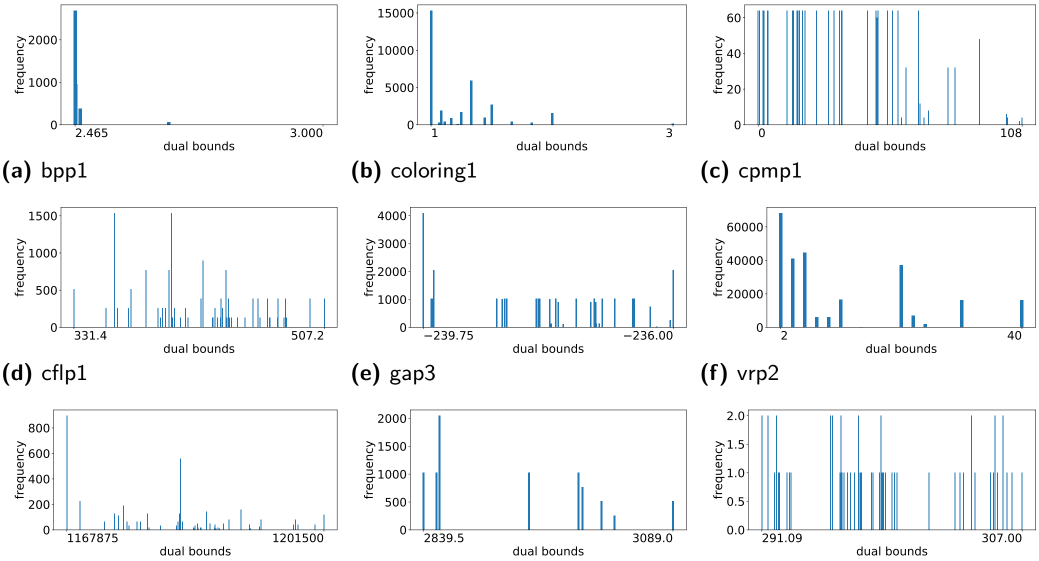
Table 1 For each (toy) instance, we state the type (binary, general integer, or mixed integer variables); the number of constraints (nconss); the number of variables (nvars); the number of DW reformulations (nrefs); the number of distinct dual bounds (ndbs); the number of minimal DW reformulations (nmin); as well as the average (cavg), the minimum (cmin), and the maximum (cmax) number of distinct dual bounds in a chain.

Instance	type	nconss	nvars	nrefs	ndbs	nmin	cavg	cmin	cmax
bpp1	BP	12	42	4096	5	27	3.318	3.0	4.0
bpp2	BP	10	30	1024	5	18	3.375	3.0	4.0
bpp3	BP	14	56	16384	6	38	3.765	3.0	5.0
cflp1	BP	14	45	16384	49	66	5.283	4.0	7.0
cflp2	BP	14	44	16384	77	90	6.400	5.0	7.0
cflp3	BP	14	44	16384	83	86	6.367	5.0	7.0
coloring1	BP	15	12	32768	16	835	4.172	2.0	8.0
coloring2	BP	12	12	4096	7	75	2.399	2.0	4.0
coloring3	BP	6	6	64	4	8	1.800	1.0	3.0
cpmp1	BP	11	30	2048	42	46	5.733	5.0	9.0
cpmp2	BP	11	30	2048	46	52	5.597	5.0	9.0
cpmp3	BP	11	30	2048	31	32	4.481	4.0	6.0
gap1	BP	15	50	32768	26	44	3.122	2.0	6.0
gap2	BP	15	50	32768	28	67	3.016	3.0	7.0
gap3	BP	15	50	32768	40	56	4.441	3.0	9.0
vrp1	MIP	18	18	262144	20	3008	3.828	2.0	8.0
vrp2	MIP	18	18	262144	12	208	3.753	2.0	7.0
flugpl	IP	12	14	4096	48	65	4.050	3.0	7.0
mod008	BP	6	319	64	51	51	5.517	4.0	6.0
p0033	BP	13	28	8192	9	9	3.183	2.0	4.0

3.1 Toy Instances

In Table 1, we list some statistics on the instances and their DW reformulations. The number of distinct dual bounds (ndbs) is usually *much* smaller than the number of all DW reformulations, i.e., many $I' \subseteq I$ yield the same $z_{DW}(I')$. The only exception is MIPLIB instance mod008 which has a much higher number of variables relative to $|I|$. Even more interestingly, also the number of minimal DW reformulations is *very* small, often not much larger than ndbs. This could hint at many chains in P with only a small number of distinct dual bounds, i.e., with many non-minimal DW reformulations. The numbers cmin, cavg, cmax give some distribution information about the number of distinct dual bounds in a chain.

Figure 1 shows histograms of the number of distinct dual bounds which often occur with the same frequencies (and often enough, these are powers of 2). This also supports the existence of constraints that do not have an influence on the dual bound in most DW reformulations. For example in the capacitated p-median instance cpmp1, there exist 5 set partitioning constraints (forcing each location to be assigned to exactly one median) and a cardinality constraint (forcing to choose exactly p locations as medians). In Section 4 we will see that often these constraints do not improve the dual bound when added to the set of convexified constraints, which explains why the frequency $2^6 = 64$ occurs multiple times in the histogram of the instance cpmp1 in Figure 1(c).



(a) bpp1 **(b)** coloring1 **(c)** cpmp1

(d) cflp1 **(e)** gap3 **(f)** vrp2

(g) flugpl **(h)** p0033 **(i)** mod008

Figure 1 Histogram for the number of DW reformulations with different dual bounds. On the x-axis we see the (discrete) spectrum of potential dual bounds from weakest z_{LP} to strongest z_{IP} and the y-axis displays the respective frequencies of dual bounds.

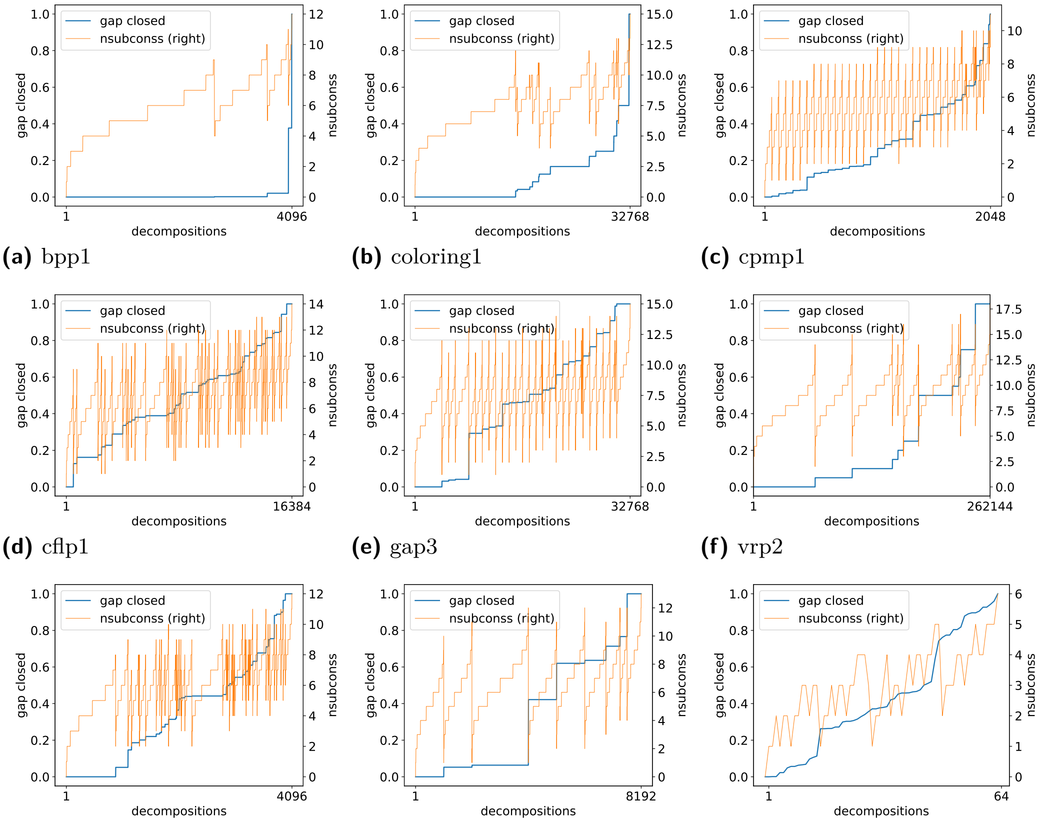
The histograms give the impression that weak(er) dual bounds are more frequent than strong(er) ones. Therefore, we analyze how many "good" DW reformulations exist. We normalize the dual bounds, which helps us to compare them across different models: We define the integrality gap that was closed by DW reformulation $I' \subseteq I$ as

$$\text{gap\_cl}(I') := \frac{z_{DW}(I') - z_{LP}}{z_{IP} - z_{LP}} \ . \tag{4}$$

Note that $\text{gap\_cl}(I') = 0 \iff z_{DW}(I') = z_{LP}$ and $\text{gap\_cl}(I') = 1 \iff z_{DW}(I') = z_{IP}$.

In Figure 2, we depict the gaps that were closed for all DW reformulations of a particular instance. The plots for instances of different problems have some common features. There are (many) more DW reformulations $I' \subseteq I$ that close only a small amount of the gap than there are DW reformulations with $\text{gap\_cl}(I') \approx 1$. This is particularly pronounced for bin packing, vertex coloring, and vehicle routing instances for which a huge portion of DW reformulations are weakest possible (for the considered objective function). Note that these instances are highly symmetric; the experiments suggest that this results in many symmetric DW reformulations yielding the same dual bound. This statement is endorsed by the fact that instances of the capacitated p-median and the capacitated facility location problem, which are more general, less symmetric variants of the bin packing problem, induce more distinct dual bounds than the bin packing instances.

Nevertheless, we notice that most DW reformulations are "in between" the weakest and strongest possible DW reformulations, i.e., for most subsets $I' \subseteq I$ it holds that $z_{LP} < z_{DW}(I') < z_{IP}$. Finally, we look at the number of convexified constraints in Figure 2. The plots illustrate that the pure number $|I'|$ is *not* a good indicator for the strength of the DW reformulation: in the majority of the instances there exist weak DW reformulations with a relatively large number of convexified constraints. Moreover, as the number of minimal DW reformulations is usually not much larger than the number of distinct dual bounds, for each "plateau" of the blue curve there are only few minimal DW reformulations, probably with relatively few convexified constraints (see lowest brown line for each plateau).



(a) bpp1 **(b)** coloring1 **(c)** cpmp1 **(d)** cflp1 **(e)** gap3 **(f)** vrp2 **(g)** flugpl **(h)** p0033 **(i)** mod008

Figure 2 The x-axis shows for each instance all DW reformulations $I' \subseteq I$, sorted by dual bound (in case of ties sorted by $|I'|$). The gap closed (4) by each DW reformulation is shown on the y-axis. The secondary y-axis displays the number of convexified constraints of each DW reformulation.

3.2 Extensions to larger Instances?

Our conjecture that $|\{z_{DW}(I') : I' \subseteq I\}| \ll 2^{|I|}$ is impractical to verify on instances with larger $|I|$. Unfortunately, there is little hope that we even obtain a statistical statement from a random sampling of DW reformulations unless we have further information about the structure of the model: the sample size would need to be too large.

The underlying statistics relates to the *distinct elements problem*, which in our context reads as: Given ℓ randomly drawn DW reformulations (and the corresponding dual bounds) from the set of all $2^{|I|}$ DW reformulations, estimate the total number of distinct dual bounds occurring among all DW reformulations. The minimum sample size ℓ needed to estimate this number (with high probability) within a given additive error tolerance $\Delta = c2^{|I|}$ for any small constant c is in $\Theta(\frac{2^{|I|}}{|I|})$ [15] which is not much smaller than $\Theta(2^{|I|})$.

4 The Influence of Individual Constraints on Dual Bounds

We have seen that the number of minimal DW reformulations is very small and the frequency of distinct dual bounds is often (close to) a power of 2. This hints at constraints that do not (or rarely) improve the dual bound when additionally convexified in particular DW reformulations. This individual impact is analyzed next.

We first introduce some notation. For each constraint $i \in I$, we investigate how the dual bound $z_{DW}(I')$ changes for subsets $I' \subseteq I$ with $i \notin I'$ if we add the constraint i to the set I' of convexified constraints, i.e., we compare $z_{DW}(I')$ and $z_{DW}(I' \cup \{i\})$. We define the *gain* of a constraint $i \in I$ when added to the set of convexified constraints $I' \subseteq I$ with $i \notin I'$ as

$$\text{gain}(i, I') := \frac{z_{DW}(I' \cup \{i\}) - z_{DW}(I')}{z_{IP} - z_{LP}} \quad . \tag{5}$$

The normalization to the integrality gap helps again to compare the gains of constraints from different instances. Correspondingly, we define the *average gain* of constraint $i \in I$ as

$$\text{gain}(i) := \frac{\sum_{I' \subseteq I : i \notin I'} \text{gain}(i, I')}{|\{I' \subseteq I : i \notin I'\}|} \quad . \tag{6}$$

Let DW reformulation I' belong to level $\ell = 0, \ldots, |I|$ if $|I'| = \ell$. We want to analyze the gain of a constraint when added to the set of convexified constraints of a DW reformulation in a given level. We define the (average) gain of constraint $i \in I$ in level $\ell = 0, \ldots, |I| - 1$ as

$$\text{gain}_\ell(i) := \frac{\sum_{I' \subseteq I : |I'| = \ell, i \notin I'} \text{gain}(i, I')}{|\{I' \subseteq I : |I'| = \ell, i \notin I'\}|} \quad . \tag{7}$$

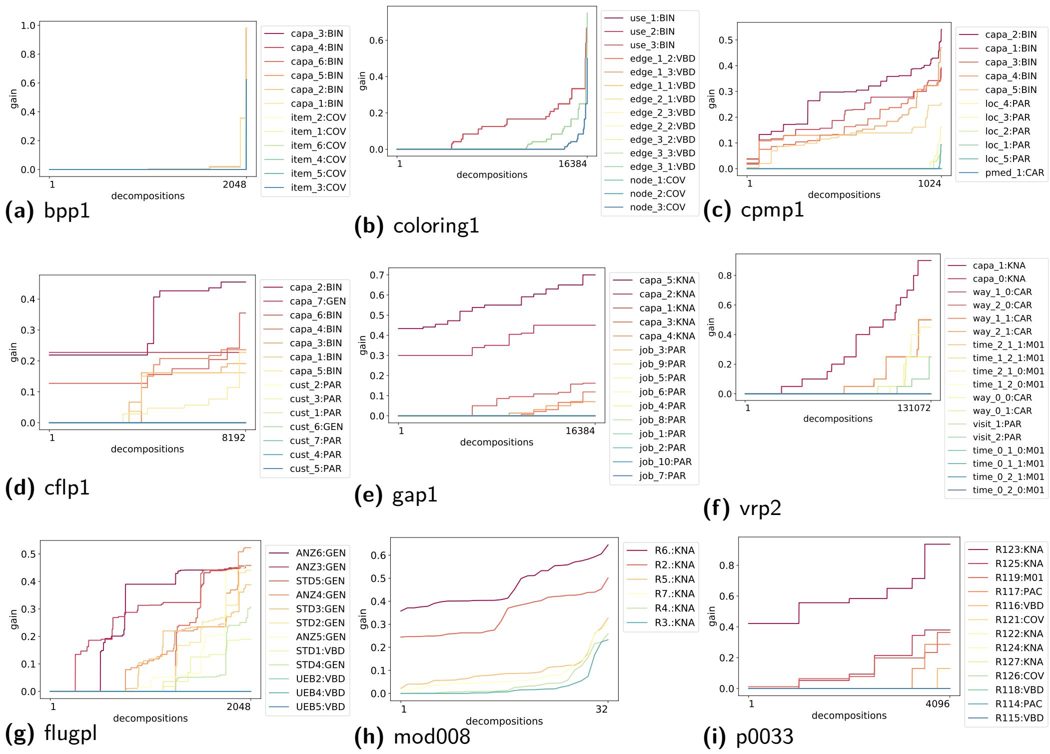
Similarly, the (average) gain of constraint $i \in I$ up to level $\ell = 0, \ldots, |I| - 1$ is defined as

$$\text{gain}_{\leq \ell}(i) := \frac{\sum_{I' \subseteq I : |I'| \leq \ell, i \notin I'} \text{gain}(i, I')}{|\{I' \subseteq I : |I'| \leq \ell, i \notin I'\}|} \quad . \tag{8}$$

Obviously, all gains defined in (5)–(8) range in $[0, 1]$. We depict the distribution of gains (defined in Equation (5)) for all constraints in Figure 3. As additional information, we include the constraint types as defined in the MIPLIB [9]. Notice that the lines corresponding to different constraints cross only occasionally. This can be interpreted as follows: Whenever the highest gain of constraint i is higher than the highest gain of constraint i', the overall gain of constraint i is higher than the overall gain of constraint i'. Hence, the average gain should give a sufficiently accurate view on which constraints have a significant effect on the dual bounds when considering all DW reformulations. Moreover, we observed that for each constraint $i \in I$ the sum of (average) gains across all levels $\sum_{\ell=0}^{m-1} \text{gain}_\ell(i)$ behaves similarly as the average gain $\text{gain}(i)$, which is why we only depict the (average) gain per level including their sum in Figure 4.

The difference in gain of constraints corresponding to different types of constraints is remarkably huge, as can be seen in Figure 4. In particular, bin packing (BIN) and knapsack (KNA) constraints, which are convexified in textbook DW reformulations, have a much larger gain than the other types. This holds not only for the sum of (average) gains in each level, but also for the individual gain in each level as well as the overall gain distribution.

Additionally, we observe that a large gain in low levels is a good indicator for a large gain in higher levels, and hence, a good indicator for a large average gain as well. This correlation becomes visible in Figure 5. The scale of the x-axis in Figure 5 is not uniform since we are only interested in whether there is a correlation between the gain in level ℓ and the average gain independently for each level ℓ at all. The correlation in the bin packing and vertex coloring instances (not shown in Figure 5) is not as high as in the other instances: In several low levels (the exact number depends on the instance) no constraint has positive gain. We assume that this is due to the high symmetry in these instances and that in an optimal LP solution not all bins/colors are used. Apart from this, it seems that the magnitude of the gain in low levels predicts the average gain quite well, which is rather remarkable.



(a) bpp1 **(b)** coloring1 **(c)** cpmp1 **(d)** cflp1 **(e)** gap1 **(f)** vrp2 **(g)** flugpl **(h)** mod008 **(i)** p0033

Figure 3 For each constraint $i \in I$ (a line of a particular color) the DW reformulations $I' \subseteq I$ (on the x-axis) are sorted by non-decreasing gain(i, I'), which is shown on the y-axis. The legend lists the constraint's name and type according to the MIPLIB constraint types [9], separated by a colon. The cardinality (CAR) constraints in vehicle routing problems are actually flow conservation constraints (this is called "upgrading" in SCIP/GCG by negating variables).

4.1 Extensions to larger Instances from the MIPLIBs

Figure 5 suggests a correlation between the gain in low levels and the average gain. Since it is intractable to compute the gains in larger instances even for levels 1 or 2, we focus on the gain in level 0, i.e., on larger instances we compute gain$_0(i)$ for each constraint $i \in I$.

We investigate instances from MIPLIB 2003 and 2010 [1, 9] for which relaxations of DW reformulations were already computed by column generation in [2]. On 12 out of these 38 instances (we excluded mine-166-5 because some DW reformulations failed to solve) there exist constraints having positive gain$_0(i)$; Figure 6 depicts the number of constraints with positive gain$_0(i)$ as well as the average gain$_0(i)$ for each constraint type on these 12 instances.

First of all, we note that set partitioning/packing/covering (PAR/PAC/COV), cardinality (CAR), and invariant knapsack (IVK) constraints cannot have positive gain in level 0 due to Geoffrion's result [7], which can also be seen in Figure 6. For all other types that occur on the MIPLIB instances there exist some constraints having positive gain in level 0. Furthermore, bin packing (BIN) and knapsack (KNA) constraints often have positive, relatively large gain in level 0 compared to other constraints. As in the toy instances, this suggests that convexifying these constraints might give strong dual bounds.

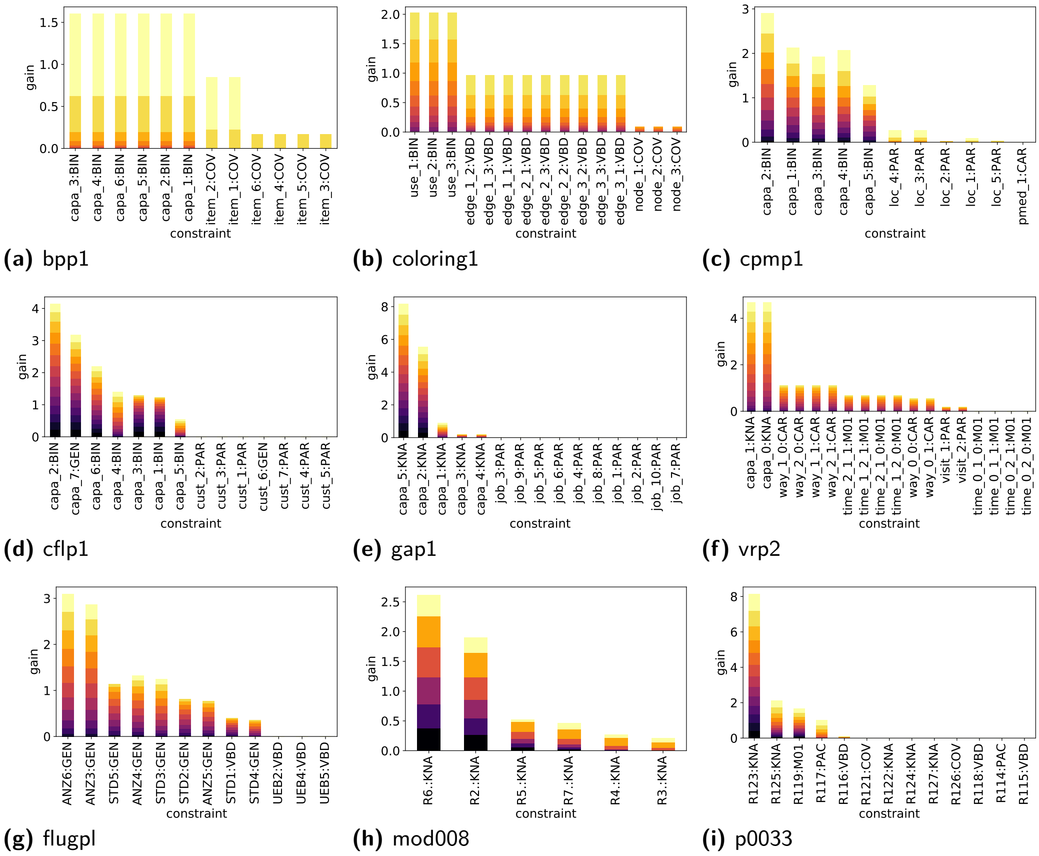


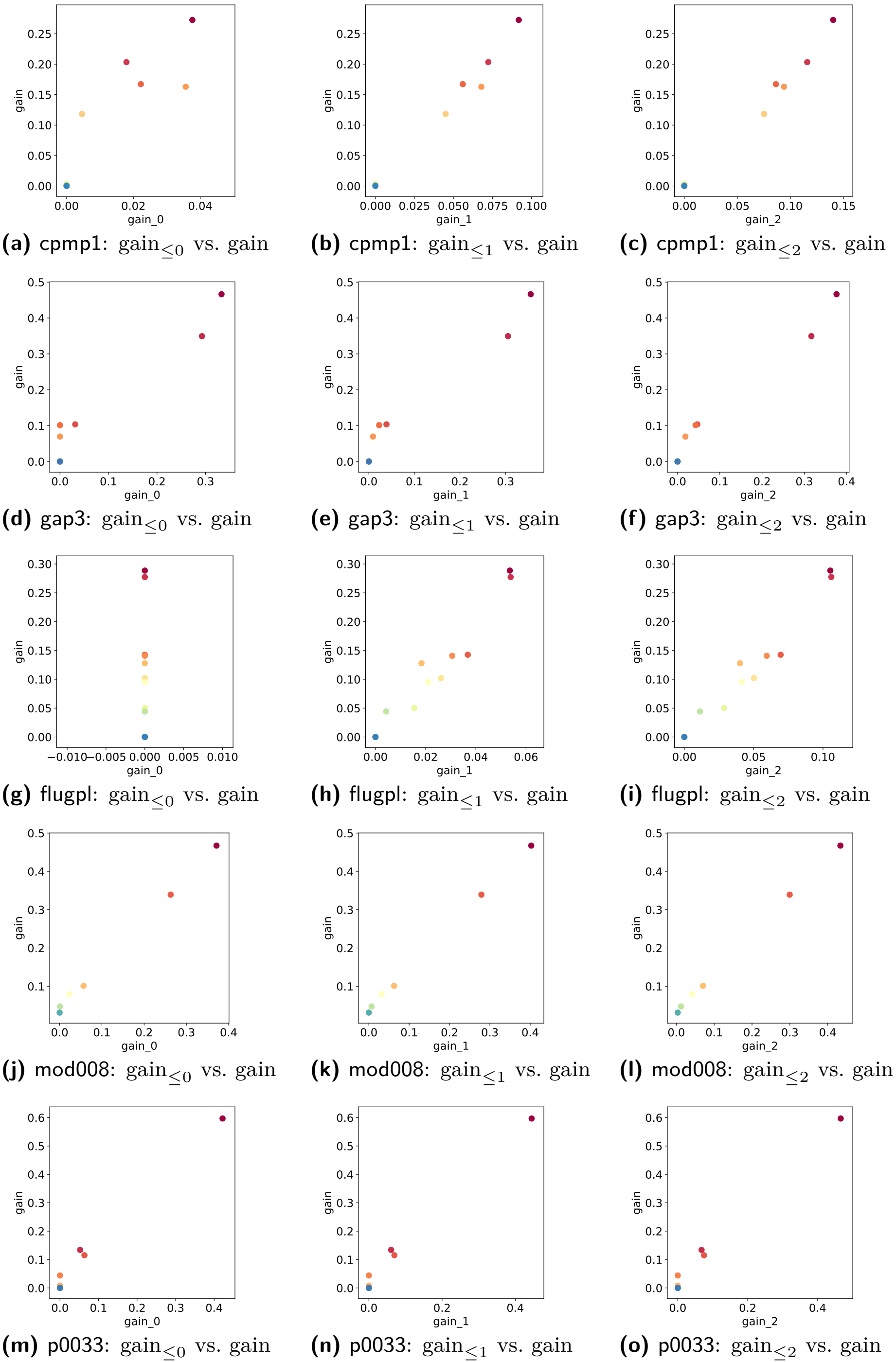
Figure 4 For each constraint (x-axis) the gain summed over all levels is displayed (y-axis). More precisely, the gains of different levels are stacked by increasing level and depicted in different colors.

5 What we have learned and what lies ahead

Because of Geoffrion's result [7] one formulates carefully that a DW reformulation *potentially* gives a stronger dual bound than z_{LP}. Figure 2 suggests that for most instances a random DW reformulation *will* give *some* improvement over z_{LP}. As the actual improvement depends on the objective function, repeating our experiments with several (randomly drawn) objective functions per instance should be interesting.

This is not to say that picking a DW reformulation at random will give one with a strong dual bound; actually, Figure 2 shows that this is rather unlikely. This is particularly visible for bin packing, but also for coloring and vehicle routing. It is fair to say that for bin packing using "the correct" DW reformulation is crucial for obtaining a strong dual bound – and this is precisely a reformulation that we would find in a textbook.

We started this experiment with the sense that there should be some sort of hierarchy of DW reformulations (other than the powerset of I). This is strongly supported (much stronger than we expected) by two experimental results we got: First, only a tiny fraction of the possible $2^{|I|}$ distinct dual bounds actually occurs, which is also true for the number of minimal DW reformulation which is very small (we would love to see geometric/polyhedral explanations for this). Second, individual (types of) constraints have considerably different impacts on strengthening an existing DW reformulation. Some of them seem to have almost no influence at all.



(a) cpmp1: $\text{gain}_{\leq 0}$ vs. gain **(b)** cpmp1: $\text{gain}_{\leq 1}$ vs. gain **(c)** cpmp1: $\text{gain}_{\leq 2}$ vs. gain

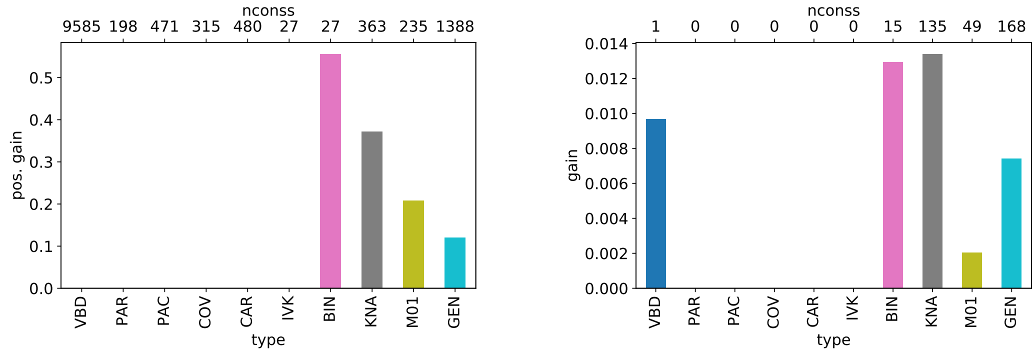
(d) gap3: $\text{gain}_{\leq 0}$ vs. gain **(e)** gap3: $\text{gain}_{\leq 1}$ vs. gain **(f)** gap3: $\text{gain}_{\leq 2}$ vs. gain

(g) flugpl: $\text{gain}_{\leq 0}$ vs. gain **(h)** flugpl: $\text{gain}_{\leq 1}$ vs. gain **(i)** flugpl: $\text{gain}_{\leq 2}$ vs. gain

(j) mod008: $\text{gain}_{\leq 0}$ vs. gain **(k)** mod008: $\text{gain}_{\leq 1}$ vs. gain **(l)** mod008: $\text{gain}_{\leq 2}$ vs. gain

(m) p0033: $\text{gain}_{\leq 0}$ vs. gain **(n)** p0033: $\text{gain}_{\leq 1}$ vs. gain **(o)** p0033: $\text{gain}_{\leq 2}$ vs. gain

Figure 5 For each constraint $i \in I$ (colored dots), the $\text{gain}_{\leq \ell}(i)$ in levels up to $\ell = 0, 1, 2$ (x-axis) is plotted against the average $\text{gain}(i)$ (y-axis), suggesting a correlation. The colors corresponding to the constraints are identical to the ones in Figure 3.



(a) Relative number of constraints with positive $\text{gain}_0(i)$ (y-axis); the secondary x-axis shows the total number of each type.

(b) Average $\text{gain}_0(i)$ of constraints with positive $\text{gain}_0(i)$ (y-axis); the secondary x-axis shows the total number of each type with pos. $\text{gain}_0(i)$.

Figure 6 Constraint types (c.f. [9]) that occur in the 12 MIPLIB instances of our testset (from [2]) which contain constraints with positive $\text{gain}_0(i)$ (x-axis); in particular, BIN is bin packing, KNA is knapsack, M01 is mixed binary, and GEN is general.

We conjecture that the poset of DW reformulations defined in Section 3 contains a (very) sparse substructure that "represents" all DW reformulations for a given instance. A starting point could consist of the set $\mathcal{I}^* \subseteq 2^I$ of minimal DW reformulations partially ordered by set inclusion (remember that in our experiments $|\mathcal{I}^*| \ll 2^{|I|}$). This again gives a poset $P^* = (\mathcal{I}^*, \subseteq)$ of height at most $|I|$ (and by Dilworth's theorem of width at least $|\mathcal{I}^*|/|I|$), but our experiments show that the height can actually be smaller, c.f. numbers cmin, cavg, cmax in Table 1. Could studying the structure and properties of such a poset yield insights into DW reformulations and maybe explain the special behavior of bin packing and coloring instances? Even if we could characterize a meaningful substructure of P, would we be able to (efficiently) compute it? Are we able to (efficiently) recognize that a DW reformulation is minimal once we have optimally solved it? Even only answering this for particular problem classes would be valuable. It is interesting in this context whether there exist pathological instances (like the Klee-Minty cubes in linear programming) with $2^{|I|}$ DW reformulations. If so, one might seek output sensitive algorithms for computing e.g., P^* whose complexity depends on the size of the output (here $|P^*|$) to account for the cases (we observed) in which the number of minimal DW reformulations is small.

Equipped with such questions we are optimistic that our experimental work spawns mathematical, algorithmic, and computational questions that hopefully guide us to a better insight into the nature of DW reformulations in general.

References

1 T. Achterberg, T. Koch, and A. Martin. MIPLIB 2003. *Oper. Res. Lett.*, 34(4):361–372, 2006.

2 M. Bergner, A. Caprara, A. Ceselli, F. Furini, M.E. Lübbecke, E. Malaguti, and E. Traversi. Automatic Dantzig-Wolfe reformulation of mixed integer programs. *Math. Programming*, 149(1–2):391–424, 2015.

3 R Bixby, Sebastian Ceria, C McZeal, and M Savelsbergh. An updated mixed integer programming library: Miplib 3.0, 1996.

4 D.G. Cattrysse, M. Salomon, and L.N. Van Wassenhove. A set partitioning heuristic for the generalized assignment problem. *European J. Oper. Res.*, 72(1):167–174, 1994.

5 P.C. Chu and JE Beasley. A genetic algorithm for the generalised assignment problem. *Comput. Oper. Res.*, 24(1):17–23, 1997.

6 G. Gamrath and M.E. Lübbecke. Experiments with a generic Dantzig-Wolfe decomposition for integer programs. In P. Festa, editor, *Proceedings of the 9th International Symposium on Experimental Algorithms (SEA)*, volume 6049 of *Lect. Notes Comput. Sci.*, pages 239–252, Berlin, 2010. Springer-Verlag.

7 A.M. Geoffrion. Lagrangean relaxation for integer programming. *Math. Programming Stud.*, 2:82–114, 1974.

8 Kaj Holmberg, Mikael Rönnqvist, and Di Yuan. An exact algorithm for the capacitated facility location problems with single sourcing. *European J. Oper. Res.*, 113(3):544–559, 1999.

9 T. Koch, T. Achterberg, E. Andersen, O. Bastert, T. Berthold, R. E. Bixby, E. Danna, G. Gamrath, A. M. Gleixner, S. Heinz, A. Lodi, H. Mittelmann, T. Ralphs, D. Salvagnin, D. E. Steffy, and K. Wolter. MIPLIB 2010. *Math. Program. Comput.*, 3(2):103–163, 2011.

10 Anuj Mehrotra and Michael A Trick. A column generation approach for graph coloring. *INFORMS J. Comput.*, 8(4):344–354, 1996.

11 Rashed Sahraeian and Payman Kaveh. Solving capacitated p-median problem by hybrid k-means clustering and fixed neighborhood search algorithm. In *Proceedings of the 2010 International Conference on Industrial Engineering and Operation Management*, pages 1–6, 2010.

12 P. Schwerin and G. Wäscher. The bin-packing problem: A problem generator and some numerical experiments with ffd packing and mtp. *Internat. Trans. Oper. Res.*, 4(5-6):377–389, 1997.

13 F. Vanderbeck and M.W.P. Savelsbergh. A generic view of Dantzig-Wolfe decomposition in mixed integer programming. *Oper. Res. Lett.*, 34(3):296–306, 2006.

14 J.T. Witt and M.E. Lübbecke. Dantzig-Wolfe reformulations for the stable set problem. repORt 2015–029, Lehrstuhl für Operations Research, RWTH Aachen University, Nov 2015. URL: `http://www.or.rwth-aachen.de/research/publications/2015-DW-stable.pdf`.

15 Y. Wu and P. Yang. Sample complexity of the distinct elements problem. *ArXiv e-prints*, Dec 2016. `arXiv:1612.03375`.