Analysis of reduced costs filtering for alldifferent and minimum weight alldifferent global constraints

Guillaume Claus, Hadrien Cambazard and Vincent Jost

Abstract. An incomplete filtering technique known as variable fixing has been used in integer programming for a long time. It relies on the reduced costs of the variables given by an optimal dual solution of the linear relaxation. Reduced-costs are used to detect some of the 0/1 variables that must be fixed to either 0 or 1 in any solution improving the best known. Reduced cost based filtering was introduced in CP for a global constraint referred to as MINIMUM WEIGHT ALLDIFFERENT and to the best of our knowledge, no analysis of this filtering technique has ever been performed. We therefore propose an analysis of reduced costs filtering for this constraint, showing that arc-consistency can be achieved with reduced-costs of n dual solutions and that this bound is tight. For ALLDIFFERENT, a single dual solution is enough. From a practical side, our end goal is the design of incomplete but anytime primal-dual filtering approaches. We illustrate this idea on the MINIMUM WEIGHT ALLDIFFERENT where a near-complete filtering can be done in shorter times.

1 Introduction

Mixed integer programming (MIP) and Constraint Programming (CP) have often been combined in the past to take advantage of the complementary strengths of the two frameworks. Many approaches have been proposed to benefit from their modeling and solving capabilities [6, 21, 19, 3, 2]. A typical integration of the two approaches is to use the linear relaxation of the entire problem in addition to the local consistencies enforced by the CP solver. The relaxation can detect infeasibility and is often added to provide a bound on the objective. A number of previous works have also proposed to use the linear relaxation for filtering the domains in a constraint programming framework [18, 19, 3, 2, 10]. Based on the relaxation, filtering can be performed using a technique referred to as reduced cost based filtering [10, 13]. It is a specific case of cost-based filtering that aims at filtering out values leading to non-improving solutions. It originates from variable fixing [16] which is performed in MIP to detect some 0/1 variables that must be fixed to either 0 or 1 in any solution improving the best known. Variable fixing relies on the reduced costs of the variables given by an optimal dual solution of the linear relaxation. It is known to be incomplete because it strongly depends on the specific dual solution used. Alternatively, it was recently shown in [11] that a complete filtering, namely arc-consistency, can be achieved with a linear relaxation when the problem considered is a satisfaction problem with an ideal integer programming formulation. Such formulations can be found for a number of common global constraints such as ELEMENT, ALLDIFFERENT, GLOBAL-CARDINALITY or GEN-SEQUENCE [19, 11]. The approach does not apply to global constraints involving a cost variable such as MINIMUMWEIGHTALLDIFFERENT [7, 10] even though it has an ideal LP formulation. A natural extension to the work [11] is to handle an objective function i.e a cost variable from the constraint point of view.

We are therefore interested in the design of filtering algorithms based on linear programming for polynomial global constraints with a cost variable. Note that when an ideal LP formulation is available for the constraint, a naive approach, typically used in practice when checking or designing propagators is to solve one LP for each variable-value pair.

Since the approach of [11] does not easily extend, we go back to reduced cost based filtering and investigate the specific case of the MINIMUMWEIGHTALLDIFFERENT global constraint (referred to as MINWALLDIFF for short in the rest of the paper). This constraint enforces n distinct values to be assigned to n variables so that the cost of the assignment is no more than a given upper-bound. Assigning to a variable $X_i$ a value $j$ of its domain incurs a cost $c_{ij} \in \mathbb{N}$ and the overall cost is the sum of all individual assignment costs. This constraint is related to the assignment problem for which a well-known LP ideal formulation is available. Interestingly, cost-based filtering was introduced in CP with the MinWALLDIFF [10] and reduced costs of the linear relaxation were used to perform filtering. An arc-consistency algorithm is first given in [20] for the more general case of the GLOBALCARDINALITY constraint with costs. Later on, [22] focuses on MinWALLDIFF and achieves arc-consistency in $O(n(d + mlog(m)))$, where $n$ denotes the number of variables, $m$ is the cardinality of the union of all variable domains, and $d$ denotes the sum of the cardinalities of the variable domains. Let’s give some details about reduced costs filtering to properly state the results of the present paper. In general, the consistency of a given value $j$ of a variable $X_i$ is established by computing the minimum increase of the optimal objective due to the assignment $X_i = j$. The optimal value of the problem restricted with $X_i = j$ is referred to as the $(i, j)$-optimal value and denoted $z^*_{ij}$. Value $j$ of $X_i$ is inconsistent if $z^*_{ij}$ is greater than the maximum cost allowed denoted $\pi$. A typical lower bound of $z^*_{ij}$ is the LP reduced cost, $r^*_{uj}$, available from an optimal dual solution $u^*$ of the linear relaxation of the assignment problem (namely $z^* + r^*_{uj} \leq z^*_{uj}$). It was used to perform an incomplete filtering in [10]. However, the value of $r^*_{uj}$ depends on the dual solution $u$ found and greatly varies in practice from one solution to another. We are now ready to state the results presented in this paper.

We prove that there always exists an optimal dual solution $u^*$ such that the reduced cost $r^*_{u^*, ij}$ provides the $(i, j)$-optimal value (i.e. such that $z^* + r^*_{u^*, ij} = z^*_{uj}$). Moreover, we show that n dual solutions are sufficient to compute all $(i, j)$-optimal values and this bound is tight. These results also show that arc-consistency for

---

1 Univ. Grenoble Alpes, CNRS, Grenoble INP G-SCOP, 38000 Grenoble, France
Definition 1. MINWALLDIFF$(X_1, \ldots, X_n, Z, c)$ has a solution if and only if the following constraint network has a solution:

\[
\text{ALLDIFFERENT}(X_1, \ldots, X_n)
\sum_{i=1}^n c_{i,X_i} \leq Z
\]

For sake of simplicity, we consider the specific case of a permutation where $\bigcup_{i=1}^n D(X_i) = \{1, \ldots, n\}$. The results presented below, namely properties 1, 2, 3 and 4 hold if there are more values than variables (the proofs remain identical).

The minimum weight alldifferent is strongly related to the assignment problem or weighted bipartite perfect matching problem stated in the graph $G = (U, V, E, c)$ referred to as the weighted variable-value bipartite graph. The set $U = \{X_1, \ldots, X_n\}$ relates to the variables, $V = \{a_1, \ldots, a_m\}$ to the set of values and edge $(X_i, a_j)$ of cost $c_{ij}$ (also denoted as a triplet $(X_i, a_j, c_{ij})$) is in $E$ if and only if $a_j \in D(X_i)$. A perfect matching $M$ in $G$ is a set of $n$ vertex-disjoint edges that define a feasible assignment of distinct values to the variables. A minimum weighted perfect matching in $G$ is denoted $M^*$ and is a minimum cost assignment of the $X$ variables.

A useful graph representation associated to a matching $M$ of $G$ is the residual graph $G_M$ as introduced in [20]. $G_M = (U, V, A, c')$ is a directed bipartite graph with the same node sets as $G$ and with arcs defined as follows:

\[
A = \{(X_i, a_j, -c_{ij}) | (X_i, a_j) \in M\} \\
\cup \{(a_j, X_i, c_{ij}) | (X_i, a_j) \in E \setminus M\}
\]

In other words, the edges from $M$ are directed from $U$ to $V$ with a cost multiplied by $-1$ and the remaining edges are directed from $V$ to $U$ with their original cost. The total cost of a set of weighted edges or arcs $S$ is denoted $c_S$ and defined as $c_S = \sum_{(i,j) \in S} c_{ij}$. Figure 1 illustrates these notions with an example made of three variables that will serve later on. A non-optimal matching $M$ of cost $c_M = 1$ is shown on $G$ in bold with its residual graph. Assuming $Z = 2$, values 2 and 1 from $X_1$ and $X_2$ respectively are not consistent and the arc-consistent domains are shown on the right of the figure.

3 Filtering algorithms for Minimum Weight AllDifferent

We briefly review the filtering algorithm achieving arc-consistency that was initially given in [20] and detailed in [22]. A support for $Z$, the lower bound of $Z$ is a matching $M^*$ of minimal cost in the weighted variable-value bipartite graph $G$. Such an optimal matching can be computed with the famous Hungarian algorithm [15]. For all edges $e \in E \setminus M^*$, there exists a perfect matching of cost less than $Z$ that contains $e$ and such that $c_{e} + c_{M^*} \leq Z$. Inconsistent values can be characterized as arcs that are not contained in any such cycles. This can be checked by computing the shortest path distances from $U$ to $V$ in $G_M$ with an all-pairs shortest path algorithm such as Johnson’s algorithm [14]. All $(i, j)$-optimal values are known at this stage and all inconsistent values can be removed. The complete procedure runs in time $O(n(d + m \log(m)))$.

A more practical and cheaper incomplete filtering technique is based on the use of linear reduced costs. It is based on the assignment problem and its Integer Programming (IP) formulation. Recall that $G = (U, V, E, c)$ denotes the weighted variable-value bipartite graph. Variables $x_{ij} \in \{0, 1\}$ encode the assignment so that $x_{ij} = 1$ means that $X_i$ is assigned to $j$. The IP formulation is as follows:

\[
\begin{align*}
\text{min} & \quad \sum_{(i,j) \in E} x_{ij}c_{ij} \\
\text{s.t.} & \quad \sum_{j \in U} x_{ij} = 1 & \quad \forall i \in U \quad (1) \\
& \quad \sum_{i \in V} x_{ij} = 1 & \quad \forall j \in V \quad (2) \\
& \quad x_{ij} \in \{0, 1\} & \quad \forall (i,j) \in E \quad (3)
\end{align*}
\]

The objective is to minimize the cost of the assignment. Constraint (1) states that each vertex $i$ of $U$ is assigned to exactly one vertex of $V$. Conversely, constraint (2) enforces each vertex $j$ of $V$ to be assigned to a single vertex of $U$. We denote by (P) the linear relaxation of (IP) i.e. the formulation where constraints (3) stating the domains $x_{ij} \in \{0, 1\}$ have been replaced by $x_{ij} \geq 0$. It is well-known that (P) is an ideal formulation so that an integer solution is returned by the simplex algorithm. Finally, the dual $(\mathcal{D})$ of (P) is the following:
Figure 1. Example of MINIMUM WEIGHT ALL DIFFERENT($X_1, X_2, X_3, Z, c$) with the weighted variable-value graph, the residual graph and the arc-consistent domains.

\[
\left\{ \begin{array}{l}
  w^* = \max \sum_{i \in U} u_i + \sum_{j \in V} v_j \\
  \text{s.t. } u_i + v_j \leq c_{ij} \quad \forall (i, j) \in E \\
  u_i \in \mathbb{R} \quad \forall i \in U \\
  v_j \in \mathbb{R} \quad \forall j \in V 
\end{array} \right.
\]

Variables $u_i$ and $v_j$ are respectively the dual variables related to constraints (1) and (2) of the primal. $u_i$ and $v_j$ are often referred to as the potentials of each $i \in U$ and $j \in V$. The reduced cost of an edge $(i, j) \in E$ for a dual solution $(u, v)$ is denoted $r_{u,v}$ and is the slack of constraint (4):

\[ r_{u,v} = c_{ij} - u_i - v_j \]

Recall that $Z$ is the maximum cost allowed, reduced cost based filtering (or variable fixing) is performed from an optimal dual solution $u^*$ of value $z^*$ and states that if $z^* + r_{u^*,v} > Z$ then $x_{ij}$ is set to 0 in any solution of cost less than or equal to $Z$. In other words, from the view point of our global constraint, value $j$ can be removed from the domain of $X_i$. Figure 2 shows the dual (D) of the example used previously and two dual optimal solutions $(u, v) = (2, 0, 0, -2, 0, 0)$ and $(u, v) = (0, 1, 0, -1, -1)$ with the corresponding reduced costs. It can be easily checked that the objective value is $z^* = w^* = 0$ for both solutions and that both are feasible. Since $Z = 2$, the first solution is able to filter value 1 from $X_2$ whereas the second solution filters value 2 from $X_1$. Thus, each of these dual solutions performs an incomplete filtering.

If we consider $Z = 1$, it is possible to filter both values with a single dual solution such as $(0.5, 0, 0, -0.5, 0, 0)$. Dual values can also be used to detect variables that must be set to 1. This is meaningful when the filtering is incomplete (It is otherwise implied by the fact that all remaining values of the domain are forbidden). We do not discuss this aspect in the present paper and refer the reader to [11] for the general statement of the rules used for reduced cost filtering.

### 4 Analysis of reduced costs based filtering

Recall that $z^*$ is the optimal value of the assignment problem and $z_{ij}^*$ denotes the optimal value of formulation (P) with the additional constraint $x_{ij} = 1$. This restricted formulation is denoted (P$_{ij}^*$) and $z_{ij}^*$ is referred to as the $(i, j)$-optimal value. Both (P) and (P$_{ij}^*$) are known to have the integrality property and an integer solution can be found by solving the linear relaxation with the simplex algorithm.

**Definition 2.** The exact reduced cost $R_{ij}$ of an edge $(i, j) \in E$ is defined as

\[ R_{ij} = z_{ij}^* - z^* \]

### 4.1 Analysis

**Property 1.** For each edge $(k, l)$ of $E$, there exists an optimal dual solution $u^*$ such that $r_{u^*,kl} = R_{kl}$.

**Proof.** Let's build explicitly the dual solution $u^*$. Let $\tilde{P}$ be the primal problem identical to (P) except for the cost of edge $(k, l)$ so that:

\[
\begin{aligned}
  \tilde{c}_{kl} &= c_{kl} - (z_{kl}^* - z^*) \\
  \tilde{c}_{ij} &= c_{ij} \quad \forall (i, j) \in E \setminus \{(k, l)\}
\end{aligned}
\]

Recall that $(k, l)$ belongs to at least one perfect matching of $G$ so that $\tilde{c}_{kl}$ is finite. Let $z^*$ be the optimal value of $\tilde{P}$ and $u^*$ be any optimal dual solution for $\tilde{P}$, the dual of $\tilde{P}$. We show below that $R_{ij} = +\infty$ if $(i, j)$ does not belong to a perfect matching of $G = (U, V, E, c)$.
"u*" is not only feasible and optimal for (D) but also gives the exact reduced cost for edge (k, l):

- Since \( z_{kl}^* \geq z^* \) the modified cost \( \tilde{c}_{kl} \) is always lower than the original cost so that \( \tilde{c}_{kl} \leq c_{kl} \). As a result, \( v(i, j) \in E, \) \( \tilde{u}_i^* + \tilde{v}_j^* \leq \tilde{c}_{ij} \leq c_{ij} \) and \( \tilde{u}^* \) is feasible for (D).

- Let's show that the optimal value is unchanged i.e. \( \tilde{z}^* = z^* \). Since \( (\tilde{P}) \) is an ideal formulation, it has (at least) one optimal integer solution. Suppose such an optimal matching of \( (\tilde{P}) \) has a cost lower than \( z^* (\tilde{z}^* < z^*) \). On the one hand, if it didn't contain \( (k, l) \), it would have the same cost for (P) contradicting \( z^* \) as the optimum of (P). On the other hand, if it contained \( (k, l) \), its cost would be smaller than \( z_{kl}^* \) in (P) since \( (k, l) \) is the only modified cost by the quantity \( (z_{kl}^* - z^*) \). In either cases, it is not possible. Moreover, since any optimal matching in (P) has also a cost of \( z^* \) in \( (\tilde{P}) \), we have \( \tilde{z}^* = z^* \). Thus \( \tilde{u}^* \) is an optimal solution for (D) since it is feasible with value \( z^* \).

- Finally, an optimal solution of \( (\tilde{P}_{kl}) \) is an optimal solution of \( (\tilde{P}) \) of value \( z^* \) with the primal variable \( x_{kl} \) set to 1. From the complementary slackness theorem, the constraint associated with \( x_{kl} \) in \( \tilde{u}^* \) must be tight.

Therefore

\[
\tilde{u}_k + \tilde{v}_l = \tilde{c}_{kl} \\
\tilde{u}_k^* + \tilde{v}_l^* = c_{kl} - (z_{kl}^* - z^*) \\
c_{kl} - \tilde{u}_k^* - \tilde{v}_l^* = z_{kl}^* - z^* \\
r_{u^* - kl} = R_{kl}
\]
Therefore, a single dual solution is enough to achieve arc-consistency of \textsc{AllDifferent}. Such a solution can be seen as an interior point of the dual problem since a positive reduced cost is a positive slack of a dual constraint. It can be found with the method given in [11] and sheds a different light on this result from a reduced cost point of view.

5 Towards a primal dual algorithm for filtering

We suggest a very simple enhancement, based on reduced costs, of the known algorithm to achieve arc-consistency initially proposed by [20] and refined in [22]. Note that we only intend to motivate our analysis and illustrate the design of anytime filtering algorithms based on primal/dual iterations. Consider the algorithm of [22]:

1. Remove from $E$, all edges $(i,j)$ that do not belong to a perfect matching of $G = (U, V, E, c)$ (see \textsc{AllDifferent}).
2. Solve the assignment problem with the Hungarian algorithm [15]. Let $u^*$ and $M^*$ respectively denote the optimal dual and primal solution found at the end.
3. For each $k \in U$
   - Replace all $c_{ij}$ by the reduced costs $r_{u^*,ij}$ so that all costs are positive (see Johnson’s algorithm [14] and [22]).
   - Compute the shortest path distances $d(l)$ from $k$ to all vertices of $l \in U \cup V$ in $G_{M^*}$ with Dijkstra algorithm to get the exact reduced-cost $R_{kl}$ for all $l \in V$ s.t. $(k, l) \in E$.

We propose to add two filtering steps over all edges of $E$ so that the algorithm can be interrupted at any time while still proving filtering over all domains. Firstly, at the end of step 2, $u^*$ can be used to perform reduced cost based filtering on all edges as done by [10]. Note that this can be done even if the Hungarian algorithm is interrupted by using the dual feasible solution it provides when interrupted. Secondly, after each iteration $k$ of step 3, reduced cost filtering for all remaining edges can be performed with an optimal dual solution built as follows: $\tilde{u}_i = u^*_i + d(i)$ and $\tilde{v}_j = v^*_j - d(j)$. It can be easily checked that it is dual feasible:
   - For each $(i, j) \in E \setminus M^*$, the shortest path distances satisfy the inequality $d(i) \leq d(j) + r_{u^*,ij}$ implying that: $\tilde{u}_i + \tilde{v}_j \leq u^*_i - d(j) + v^*_j - d(j) + r_{u^*,ij} = c_{ij}$
   - For each $(i, j) \in M^*$, $r_{u^*,ij} = 0$. Therefore $d(i) = d(j)$ and $\tilde{u}_i + \tilde{v}_j = c_{ij}$.

The time complexity remains in $O(n^3)$ in the worst case. The filtering algorithm can be seen as producing a sequence of dual feasible solutions $u^1, \ldots, u^{n+1}$ whose reduced-costs are used for filtering the entire domains. The only interest of this approach is to derive an anytime algorithm and to stop the filtering after $Q$ dual solutions: $u^1, \ldots, u^Q$. The $Q-1$ vertices (variables) from which the dual solutions are built can be chosen heuristically. In the experiments and as a mean of illustration we chose them randomly and fixed $Q = 1 + 0.1 n'$ where $n'$ is the number of ungrounded variables (initially $n$). Steps 2 and 3 enhanced with reduced cost filtering are illustrated in Figure 4. The variable-value graph is given in (a), an optimal dual solution (provided by the Hungarian) is shown in (b) and value 4 is filtered from $X_1$ (step 3) is shown in (c). The exact reduced costs of $D(X_i)$ lead to filtering value 2 from $D(X_1)$. Additionally, value 3 is removed from $D(X_2)$ by the same dual solution.

}\end{proof}

\begin{figure}
\centering
\includegraphics[width=\textwidth]{figure3.png}
\caption{Edges $(X_2, 1)$ and $(X_4, 3)$ are incompatible.}
\end{figure}

property 4. There exists an optimal dual solution $u^* \text{ s.t.}$

\[ r_{u^*,ij} > 0 \ \forall (i,j) \in I \]

Proof. Since the $\{0,1\}$ encoding presented above implies $R_{ij} \geq 1 \ \forall (i,j) \in I$, we can consider a set of optimal dual solutions $\{\tilde{u}^{ij} : (i,j) \in I\}$ with $r_{\tilde{u}^{ij},ij} \geq 1$.

Let $\tilde{u}^*$ be the average solution of the previous set:

\[ \tilde{u}^* = \frac{\sum_{(i,j) \in I} \tilde{u}^{ij}}{|I|}. \]

This solution is feasible, optimal, and has a positive reduced cost for each $(i,j) \in I$. \end{proof}
6 Experimental results

We analyse experimentally the filtering obtained with reduced costs and the behavior of the primal-dual approach proposed in Section 5. Levels of consistency are reported as a percentage of the total number of inconsistent values (for a given upper bound $\tau$). All experiments ran on a laptop Dell Precision 5530 (i7-8850H 2.60GHz 16Go Ram, Linux Gentoo 64bits) using a single thread.

A single call to the filtering. The results reported on Figure 5 are obtained with random costs matrices of sizes ranging from $n = 20$ to $n = 400$ and where each $c_{ij}$ is drawn in $\{0, \ldots, 100\}$ with a uniform distribution. The value of $\tau = 1.2z^*$ is used to act as the upper bound of $z^*$. Figure 5.a shows the cpu time needed to achieve three levels of consistencies $(80\%, 98\%$ and $100\%$ of AC) as well as the time to perform and to prove that arc-consistency is enforced (denoted AC). Figure 5.b shows the level of consistency reached after each dual feasible solution for 20 random instances of size $n = 400$. Each line is one of the instances and a point $(x, y)$ means that $y\%$ of the complete filtering has been obtained with the $x$ first dual solutions $u^1, \ldots, u^x$ of the sequence produced by the primal dual approach. We include (in the sequence) the dual solutions provided after each primal/dual iteration of the Hungarian method itself to visualize better what would happen when interrupting the Hungarian thus the number of iterations can be larger than $n+1$ in the results. Table 1 presents the same results with more precise numerical values by explicitly giving the value of $x$ required to achieve a given percentage $p$ of filtering (namely $p \in \{0.80, 0.98, 1\}$) as well as the corresponding time in seconds (column T). Column Q gives the total number of dual solutions produced when proving arc-consistency. Moreover, we report the percentage of filtering that would be performed by the optimal dual solution alone (column F), i.e. the solution found at the end of the Hungarian algorithm which is the traditional approach for filtering with reduced-costs [10]. There are in average around 25000 values removed in the first case ($\tau = 1.20z^*$) and around 2500 in the second case ($\tau = 1.235z^*$).

- For $\tau = 1.20z^*$, most of all inconsistent values are identified in a fraction of the total time needed to enforce arc-consistency (See Figure 5.a and Table 1 for $p = 98\%$). Typically, $98\%$ of the values are removed in less than $10\%$ of the total time.

<table>
<thead>
<tr>
<th>$\tau$</th>
<th>$p = 80%$</th>
<th>$p = 98%$</th>
<th>$p = 100%$</th>
<th>AC</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>mean $x$</td>
<td>1.37</td>
<td>1.02</td>
<td>0.05</td>
<td>152.8</td>
<td>347.8</td>
</tr>
<tr>
<td>median $x$</td>
<td>1.0</td>
<td>0.00</td>
<td>0.06</td>
<td>455.5</td>
<td>465.5</td>
</tr>
<tr>
<td>max $x$</td>
<td>58</td>
<td>66</td>
<td>14</td>
<td>465</td>
<td>467</td>
</tr>
<tr>
<td>mean $y$</td>
<td>80.0</td>
<td>0.51</td>
<td>31.3</td>
<td>1.84</td>
<td>450.3</td>
</tr>
<tr>
<td>median $y$</td>
<td>58.0</td>
<td>1.38</td>
<td>308.5</td>
<td>1.35</td>
<td>451.5</td>
</tr>
<tr>
<td>max $y$</td>
<td>274</td>
<td>2.93</td>
<td>453</td>
<td>4.80</td>
<td>464</td>
</tr>
</tbody>
</table>

Table 1. Numerical details of the results presented in Figure 5. (20 random instances; $n=400$)

Figure 4. Example. (a): Variable-value graph ; (b): Hungarian solution ; (c): Dual solution given by the Dijkstra algorithm from $X_1$.

Figure 5. (a): cpu time (in s) needed to filter 80%, 98% and 100% of the inconsistent values for different problem sizes $n$ ranging from 20 to 400. The line AC shows the time needed to prove arc-consistency. (b): Level of consistency depending on the cumulative number of dual feasible solutions enumerated $(n = 400, c_{ij} \in \{0, \ldots, 100\}$ and $\tau = 1.2z^*$).
• Filtering is gathered with intermediate dual solutions.
• The optimal dual solution alone provides nearly all the filtering for $\tau = 1.20z^*$ but the algorithm can be interrupted earlier with very little loss.
• As the gap increases to $\tau = 1.235z^*$, the transition is abrupt and achieving more than 80% of the filtering requires to go beyond the dual solution of the Hungarian algorithm.

Preliminary results during resolution. We investigate the Resource Constrained Assignment Problem (RCAP) [1] to illustrate the same ideas during resolution. The problem is to find a minimum weight assignment that satisfies one or several resource constraints. It is NP-Hard and can be formulated with the traditional model ($P_{IP}$) given in Section 3 (using a complete set $E = U \times V$ of edges) and a set $K$ of additional resource constraints:

$$\sum_{i=1}^{n} \sum_{j=1}^{n} d_{ij} x_{ij} \leq b^k \ \forall k \in K$$

$d_{ij}^k$ denotes the consumption of resource $k$ by assigning value $j$ to variable $i$ and $b^k$ is the capacity of the same resource. Moreover we denote by $b'$ (resp. $d'$) the sum of all capacities (resp. consumption) i.e $d'_j = \sum_{k \in K} b^k$ and $d_{ij}' = \sum_{k \in K} d_{ij}^k$. Each resource can be seen as an assignment problem and the CP model can be written as follow:

$$\begin{align*}
\min Z \\
\text{s.t.} \quad Z = \sum_{i=1}^{n} c_i x_i & \quad (7) \\
\text{MINWALLDIFF}(X_1, \ldots, X_n, Z, c) & \quad (8) \\
\text{MINWALLDIFF}(X_1, \ldots, X_n, d^k, b^k) & \quad \forall k \in R \quad (9) \\
\text{MINWALLDIFF}(X_1, \ldots, X_n, d', b') & \quad (10)
\end{align*}$$

Constraint (7) is implemented using Element constraints. Constraint (8) enforces the $X$ variables to be all different and gives the strong filtering dedicated to the objective function. Constraints (9) model the resources. Finally (10) is a redundant constraint that was found useful when resources are tight. Overall, this model involves $K + 2$ MINWALLDIFF constraints. We compare the following filtering algorithms:

• The arc-consistency algorithm of [22] denoted $ac$.
• The initial approach of [10] denoted $hung$.
• A version of the primal-dual filtering with $Q = 1 + 0.1n'$ (denoted $pd10$) where $n'$ is the number of ungrounded variables (initially $n$). The $0.1n'$ variables to build the dual solutions are chosen randomly. The intention is that only 10% of the variables are filtered precisely following the original algorithm of [22] but dual solutions provide filtering over potentially all remaining variables.

We use a 600s time limit and the Choco 3 solver [17]. Instances are generated with a uniform distribution for costs/resources in {0, ..., 100} and capacities between 0.1 and 0.6 of 100n. We consider 6 classes of instances with $n \in \{100, 200, 500\}$ and $k \in \{2, 6\}$ with 5 instances per class so 30 in total. The search is performed by ordering the variables lexicographically in non-increasing resource consumption. The consumption is computed (for $X_i$) as $\sum_{j=1}^{n} \sum_{k \in K} d_{ij}^k$ where $d$ is the consumptions normalized in [0, 1] to be able to compare resources with various capacities. The value ordering heuristic picks the value $j$ with the minimum resource consumption ($j = \arg \min_{c \in \Delta D(X_i)} \sum_{k \in K} \frac{b^k}{d_{ij}^k}$). Finally, to make sure the tree is traversed at various depths and the search does not remain stuck in a subtree at a very high or very low depth, we use a Limited Discrepancy Search [12] provided by the solver. Table 2 reports the number of nodes opened per second (nodes/s), the gap ($\Delta$ Sol) between the best solution found compared to the one found by $ac$ (a negative gap is an improvement), the number of times the algorithm obtained the best solution among the three approaches ($\%$ best Sol) and the total percentage of the filtering that was performed compared to arc-consistency ($\%$ AC). This last metric was obtained by instrumenting the code and re-running the solving process for the same number of nodes, to find (at each call to the filtering) the number of values arc-consistency would have removed and the number of values actually removed by the algorithm under evaluation.

<table>
<thead>
<tr>
<th>n</th>
<th>k</th>
<th>Alg</th>
<th>nodes/s</th>
<th>$%$ AC</th>
<th>$\Delta$ Sol</th>
<th>$%$ best Sol</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>2</td>
<td>ac</td>
<td>105.7</td>
<td>100%</td>
<td>-8.6%</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>hung</td>
<td>619.2</td>
<td>87.1%</td>
<td>-16.2%</td>
<td>4</td>
</tr>
<tr>
<td>100</td>
<td>6</td>
<td>ac</td>
<td>73</td>
<td>100%</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>hung</td>
<td>366.5</td>
<td>73.2%</td>
<td>-1.6%</td>
<td>0</td>
</tr>
<tr>
<td>200</td>
<td>2</td>
<td>ac</td>
<td>27.6</td>
<td>100%</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>hung</td>
<td>249.4</td>
<td>84.6%</td>
<td>-18.5%</td>
<td>0</td>
</tr>
<tr>
<td>200</td>
<td>6</td>
<td>ac</td>
<td>16.9</td>
<td>100%</td>
<td>-9.8%</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>hung</td>
<td>90.2</td>
<td>87.5%</td>
<td>-13.6%</td>
<td>4</td>
</tr>
<tr>
<td>500</td>
<td>2</td>
<td>ac</td>
<td>0.9</td>
<td>100%</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>hung</td>
<td>36.1</td>
<td>68.9%</td>
<td>-2.4%</td>
<td>0</td>
</tr>
<tr>
<td>500</td>
<td>6</td>
<td>ac</td>
<td>11.1</td>
<td>100%</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>hung</td>
<td>15.7</td>
<td>58.9%</td>
<td>-∞</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>pd10</td>
<td>12</td>
<td>73.6%</td>
<td>-∞</td>
<td>5</td>
</tr>
</tbody>
</table>

Table 2. Results on the RCAP problem.

7 Conclusion

We conducted an analysis of reduced cost based filtering for a very fundamental global constraint related to the assignment problem: MINIMUMWEIGHTALLDIFFERENT. Reduced-cost filtering was proposed in 1999 by Focacci and al [10] on the very same global constraint and has been used since without detailed analysis. The present work shows that arc-consistency can be achieved with the use of reduced costs and that a minimum number of $n$ dual feasible solutions are always required in the worst-case. It also shows that arc-consistency of ALLDIFFERENT can be established with the reduced costs of a single dual solution giving a different view-point on the result of [11]. The analysis is based on the LP formulation of the constraint and its integrality property (ideal formulation). In particular, it does not rely on the shortest path sub-problems of the dedicated arc-consistency algorithm or the flow structure of [20]. The proofs presented are new and we aim at generalizing these results to global constraints with an ideal LP formulation and a cost variable which encompasses a large class of constraints. The next step is to turn this analysis into more efficient filtering algorithms based on primal/dual iterations.
References


