Noname manuscript No.
(will be inserted by the editor)

Implementing cutting plane management and selection techniques

Franz Wesselmann · Uwe H. Suhl

Received: date / Accepted: date

Abstract One main objective of research in the area of mixed-integer programming is developing cutting plane techniques to improve the solvability of mixed-integer programs (MIPs). Various cutting plane separators are typically available in MIP solvers. The large number of cutting planes generated by these separators, however, can pose a computational problem. Therefore, a sophisticated cut management is indispensable. In this paper, we discuss cut quality measures and detail the architecture of the cut pool and the cut selection algorithm in the Mops (Mathematical OPtimization System) MIP solver. Furthermore, we analyze the impact that our algorithm has on the overall performance of the solver based on computational experiments. These experiments show that our algorithm succeeds in reducing the running times and the number of cutting planes added to the linear programming (LP) relaxation.

Keywords integer programming · cutting planes

1 Introduction

Cutting planes play a central role in solving mixed-integer programs (MIPs). Closely related to cut generation is the problem of reducing the number of cuts that are added to the linear programming (LP) relaxation. This problem in particular has become increasingly important since there are numerous cutting planes available in MIP solvers like Cplex, Xpress, Gurobi or Mops [30]. If the number of generated cuts is large, adding all of them represents a

Franz Wesselmann, Uwe H. Suhl
University of Paderborn
Warburger Str. 100
33098 Paderborn
Germany
Tel.: +49-5251-602385
Fax: +49-5251-603542
E-mail: {wesselmann, uwe.suhl}@dsor.de

computational burden for the LP solver, possibly leading to longer (node) solution times. Thus finding cut quality measures and developing cut selection algorithms are central issues in the study of cutting plane algorithms.

In this paper we address the problem of selecting only a subset of the generated cuts. To this end we discuss the reliability of different cut selection rules or cut quality measures respectively and point out their merits and demerits. We also describe a cut selection algorithm and highlight important implementation details. Finally, we present computational results to emphasize the importance of cut selection algorithms and study the impact of different cut quality measures on the overall performance of an MIP solver.

The Mops MIP solver is a high performance system for solving large-scale LP and MIP problems. The current version of Mops features state-of-the-art primal simplex, dual simplex and interior point algorithms to solve LP problems. Moreover, the system uses sophisticated heuristics, branch-and-bound (branch-and-cut) algorithms and cutting plane techniques to tackle MIPs.

2 Cut Selection

Crucial to solving MIPs is a strong LP relaxation. There are a number of techniques such as bound and coefficient reduction to improve the strength of the LP relaxation. However, cutting planes like Gomory mixed-integer cuts, cover cuts, flow cover cuts, mixed-integer rounding cuts, etc. are very important to obtaining a tight LP relaxation.

The primary aim of cutting plane selection is to choose the "right" cuts which help to solve an MIP problem more rapidly. The essential question is how to select such a subset of cuts. To answer this question reliable cut quality measures are needed. An additional aim of cutting plane selection is to keep the number of cuts that are added to the LP relaxation small.

2.1 Literature Review

Padberg and Rinaldi [29] emphasize that "finding a reasonable quality measure is one of the central issues in the area of polyhedral cutting-plane algorithms that is - as of today - not yet investigated satisfactorily" [29, p. 79]. Although this statement was made over fifteen years ago, the situation has not fundamentally changed. Several contributions on cutting plane theory address measuring cut quality as a secondary problem. Balas et al. [9] point out that using the improvement of the objective function as a cut quality measure has certain drawbacks (e.g. zero gap problems) and suggest calculating the Euclidean distance between a given LP solution and the cutting plane instead. Christof and Reinelt [12] propose preference be given to cutting planes that are as parallel as possible to the objective function and try to estimate the improvement of the objective function. A different proposal put forward by Ferris et al. [17] is called "cut selection by usage". Here the policy is to add all generated cuts tentatively and then track the pivots performed by the dual simplex method. If a cut is used, meaning that is was pivoted on, it is marked and later on only marked cuts are added. Andreello et al. [6] discuss a cut selection algorithm for $\{0, \frac{1}{2}\}$-Chvátal-Gomory cuts. To increase the diversity of cuts, they check cuts pairwise for parallelism (see also [17]). Achterberg [2] proposes a sophisticated cut selection algorithm which makes use of several cut quality measures. Fischetti and Salvagnin [18] discuss a framework where cuts are generated and then relaxed in a Lagrangian fashion. Amaldi et al. [4] propose a lexicographic multi-objective cutting plane generation scheme. They concentrate on generating cuts which are undominated and maximally diverse with respect to previously found cuts.

2.2 Cut Quality Measures

Assume that a mixed-integer linear program is given in the form

$$\max\left\{c^T x : Ax \leq b, Dx = d, l \leq x \leq u, x_j \in \mathbb{Z}, \forall j \in N_I\right\} \tag{1}$$

where $c, x, l, u \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, $D \in \mathbb{R}^{p \times n}$, $d \in \mathbb{R}^p$, and $N_I \subseteq N = \{1, \ldots, n\}$. Let x^* be an optimal solution to the corresponding LP relaxation. Suppose that cut separation algorithms generated a number of cutting planes

$$\left(\alpha^i\right)^T x = \sum_{j \in N} \alpha^i_j x_j \leq \beta^i, \quad \forall i \in L, \tag{2}$$

where $\alpha^i \in \mathbb{R}^n$ and $\beta^i \in \mathbb{R}$ and L is an arbitrary index set. To simplify the notation we omit the index i whenever possible and denote a single cut by

$$\alpha^T x = \sum_{j \in N} \alpha_j x_j \leq \beta. \tag{3}$$

Some quality measures for cutting planes are the following.

Violation The simplest quality measure for cutting planes is *violation* which is defined as

$$v\left(\alpha, \beta, x^*\right) = \alpha^T x^* - \beta. \tag{4}$$

Using violation to measure the quality of cutting planes has several drawbacks. For example, violation is not invariant under scaling. Given any integer $k \in \mathbb{Z}_+$, the cut $k\alpha^T x \leq k\beta$ is k times more violated than $\alpha^T x \leq \beta$. Observe that the value of $v(\alpha, \beta, x^*)$ is positive for violated cuts with this definition.

Relative Violation A straightforward approach to overcome issues related to scaling is to divide the violation value by the absolute value of the right-hand side, i.e. normalizing the right-hand side. The resulting quality measure is called *relative violation* (cf. Linderoth [24]) and is defined as follows:

$$rv\left(\alpha, \beta, x^*\right) = \frac{v\left(\alpha, \beta, x^*\right)}{|\beta|} \tag{5}$$

However, it is not possible to compute the relative violation if the right-hand side of the cut is zero. In this case we instead use the violation.

Distance A third possibility to measure the cut quality is to determine the *Euclidean distance* between an LP solution x^* and the hyperplane $\alpha^T x = \beta$. In other words, we consider the distance between x^* and its orthogonal projection on this hyperplane:

$$d(\alpha, \beta, x^*) = \frac{v(\alpha, \beta, x^*)}{\|\alpha\|} \tag{6}$$

This quantity is also well-known as the *distance cut off* by a cut. Using the Euclidean distance can be seen as an improvement of relative violation and overcomes some of its drawbacks. Like relative violation, the Euclidean distance is independent of scaling. In addition it can also be calculated for cuts whose right-hand sides take arbitrary values.

A problem of the Euclidean distance is that it does not work properly with a system of equations (cf. Cook et al. [13]). Specifically, a polyhedron whose inequality description contains equations is not full-dimensional. Measuring cut quality by the Euclidean distance thus does not take into account that a feasible solution must lie at the intersection of the hyperplanes defined by these equations.

Example 1 Consider the simple integer program

$$\max\{2x_1 + x_2 : x \in X\} \tag{7}$$

where $X = X^R \cap \mathbb{Z}^2$ and $X^R = \{x \in \mathbb{R}^2_+ : 6x_1 + 4x_2 = 20\}$. The optimal solution to the LP relaxation is $x^* = (3\frac{1}{3}, 0)$. The two inequalities $3x_1 + x_2 \leq 8$ and $x_1 \leq 2$ are valid for X and cut off x^*. The hyperplanes geometrically

	violation	rel. violation	distance
$3x_1 + x_2 \leq 8$	2	$\frac{1}{4}$	$\frac{2}{\sqrt{10}}$
$x_1 \leq 2$	$\frac{4}{3}$	$\frac{2}{3}$	$\frac{4}{3}$

Table 1 Cut quality measures

representing the two cuts have different Euclidean distances from x^*, meaning that one of them is more favorable (see Table 1). But this is actually not true. When individually adding each cut to the formulation, the LP relaxation has an optimal integral solution $x^* = (2, 2)$ after reoptimization in both cases.

Rotated Distance Cook et al. [13] propose to explicitly take into account a given system of equations when computing the Euclidean distance. Specifically, they consider the distance between an LP solution x^* and a "rotated" version of the cut hyperplane which is orthogonal to the affine space formed by the solutions of the system of equations. This rotated hyperplane has the characteristic that the orthogonal projection of any feasible LP solution on it satisfies the equality constraints, thus the distance between an LP solution and a point which is feasible with respect to the equality constraints is computed.

Given that $\alpha^T x \leq \beta$ is a valid inequality for the MIP (1), the inequality $\bar{\alpha}^T x \leq \bar{\beta}$ with $\bar{\alpha} = \alpha + D^T \lambda$ and $\bar{\beta} = \beta + d^T \lambda$ is valid for the set $\{x : Dx = d,\ \alpha^T x \leq \beta\}$ for any $\lambda \in \mathbb{R}^p$ and consequently for (1).

Cook et al. [13] propose to select λ such that the hyperplane $\bar{\alpha}^T x = \bar{\beta}$ is orthogonal to the set of feasible solutions $\{x : Dx = d\}$. If we require $\bar{\alpha}$ to satisfy $D\bar{\alpha} = 0$, we obtain $\lambda = -(DD^T)^{-1} D\alpha$ and thus

$$\bar{\alpha} = \alpha - D^T \left(DD^T\right)^{-1} D\alpha. \tag{8}$$

We assume D to have full rank, because then the $p \times p$ square matrix DD^T is symmetric positive definite and therefore its inverse exists. Consider the $n \times n$ square matrix $C = D^T(DD^T)^{-1}D$ and observe that $\bar{\alpha} = 0$ if $C\alpha = \alpha$. If $\alpha x \leq \beta$ is a valid inequality for (1) which is satisfied by all solutions $\{x : Dx = d\}$, then the vector α can then be written as a linear combination of the rows of D, i.e. $\alpha = D^T \tau$ with $\tau \in \mathbb{R}^p$, and we have $\bar{\alpha} = 0$. Since we consider cutting planes with respect to an LP solution x^*, we assume that $\bar{\alpha} \neq 0$ in what follows.

For any LP solution x^* we have $Dx^* = d$ and consequently $\bar{\alpha}^T x^* - \bar{\beta} = \alpha^T x^* - \beta + ((D^T\lambda)^T x^* - d^T\lambda) = \alpha^T x^* - \beta$, i.e. the original cut and its rotated version are violated by the same amount. The orthogonal projection of an LP solution x^* on the rotated hyperplane $\bar{\alpha}^T x = \bar{\beta}$ is given by $\hat{x} = x^* - \bar{\alpha}(\bar{\alpha}^T x^* - \bar{\beta})/\|\bar{\alpha}\|^2$. As noted above, we have $\bar{\alpha}^T \hat{x} = \bar{\beta}$ and $D\hat{x} = d$ indicating that the point $\hat{x}$ lies at the intersection of the equality constraints and the cut hyperplane.

The *rotated distance* of a cut is thus given by

$$rd\left(\alpha, \beta, x^*\right) = \frac{v\left(\alpha, \beta, x^*\right)}{\left\|\alpha - D^T\left(DD^T\right)^{-1} D\alpha\right\|}. \tag{9}$$

Example 2 (continued) We next compute the rotated distances for the cuts $3x_1 + x_2 \leq 8$ and $x_1 \leq 2$. For the first cut we obtain $\lambda = -\frac{11}{26}$ and the rotated cut reads $\frac{6}{13}x_1 - \frac{9}{13}x_2 \leq -\frac{6}{13}$. Concerning the second cut we get $\lambda = -\frac{3}{26}$ which translates into the rotated cut $\frac{4}{13}x_1 - \frac{6}{13}x_2 \leq -\frac{4}{13}$. According to the rotated distances both cuts are of the same quality (see Table 2).

	violation	distance	rot. distance
$3x_1 + x_2 \leq 8$	2	$\frac{2}{\sqrt{10}}$	$\frac{2}{\sqrt{\frac{117}{169}}} = \frac{26}{\sqrt{117}}$
$x_1 \leq 2$	$\frac{4}{3}$	$\frac{4}{3}$	$\frac{\frac{4}{3}}{\sqrt{\frac{52}{169}}} = \frac{26}{\sqrt{117}}$

Table 2 Cut quality measures (continued)

Distance with Bounds The orthogonal projection of an LP solution on a cut hyperplane is not necessarily respecting the bound constraints on the variables. Cook et al. [13] propose to compute the point which is nearest to the LP solution in terms of the Euclidean distance and satisfies the cut inequality and the bound constraints. The Euclidean distance between the LP solution and this point is then used to evaluate the quality of a cut. More formally, the *distance with bounds* is given by

$$db\,(\alpha, \beta, x^*) = \min\left\{\|x - x^*\| : \alpha^T x \leq \beta, l \leq x \leq u\right\}. \tag{10}$$

Let $\hat{x}$ be a solution to (10). Since an optimal LP solution x^* has to satisfy $l \leq x^* \leq u$, any point on the line segment between x^* and $\hat{x}$ also satisfies the bound constraints, i.e. $l \leq x^* + t(\hat{x} - x^*) \leq u$ with $0 \leq t \leq 1$. If the solution $\hat{x}$ does not lie on the cut hyperplane, i.e. $\alpha^T \hat{x} < \beta$, then there is a point p on the line segment which is nearer to x^* and satisfies $\alpha^T p = \beta$ and $l \leq p \leq u$. Thus any solution $\hat{x}$ to (10) with $\alpha^T \hat{x} < \beta$ cannot be optimal.

Example 3 (continued) In our example both variables are required to be non-negative, i.e. $x_1, x_2 \geq 0$. The orthogonal projection of $x^* = (3\frac{1}{3}, 0)$ on the first cut hyperplane $3x_1 + x_2 = 8$ is $\hat{x} = (2\frac{11}{15}, -\frac{1}{5})$. Since $\hat{x}_2 < 0$ this point does not satisfy the bound constraints. The optimal solution to (10) is $\tilde{x} = (2\frac{2}{3}, 0)$ and the distance between x^* and $\tilde{x}$ is $\frac{2}{3}$ (see Table 3). The orthogonal projection of x^* on the cut hyperplane $x_1 = 2$ is $\hat{x} = (2, 0)$. Since this point satisfies the bound constraints, the distance with bounds is equal to the plain distance.

	distance	distance with bounds
$3x_1 + x_2 \leq 8$	$\frac{2}{\sqrt{10}}$	$\frac{2}{3}$
$x_1 \leq 2$	$\frac{4}{3}$	$\frac{4}{3}$

Table 3 Cut quality measures (continued)

Rotated Distance with Bounds Concerning the rotated distance, the orthogonal projection of an LP solution on the rotated cut hyperplane satisfies the equality constraints $Dx = d$. This projected point, however, may in turn not comply with the bound constraints $l \leq x \leq u$. To cope with this issue, we compute the rotated cut $\bar{\alpha}^T x \leq \bar{\beta}$ with $\bar{\alpha}$ as defined in Equation (8) and impose this inequality as a constraint when solving (10) instead of the original cut inequality. We call the resulting quality measure *rotated distance with bounds*

$$rdb\,(\alpha, \beta, x^*) = db\left(\bar{\alpha}, \bar{\beta}, x^*\right). \tag{11}$$

Adjusted Distance An index $j \in N$ satisfying $\alpha_j > 0$ and $x_j^* = 0$ does not affect the violation of the cut. However, it increases the norm of the coefficient vector α.

Example 4 Consider two cuts $(\alpha^i)^T x \leq \beta^i$ and $(\alpha^k)^T x \leq \beta^k$ with $\beta^i = \beta^k$. Let (N_1, N_2) be a partition of N satisfying

$$\alpha_j^i = \alpha_j^k \geq 0, \quad \forall j \in N_1, \tag{12}$$
$$\alpha_j^i > \alpha_j^k \geq 0, \quad \forall j \in N_2. \tag{13}$$

Suppose that the LP solution x^* to the underlying problem fulfills the condition $x_j^* = 0, \forall j \in N_2$ and violates both cuts specified above. Clearly, cut i mathematically dominates cut k. However, identical (relative) violations are assigned to both cuts. Moreover, it follows from the assumed relations between α^i and α^k that $\|\alpha^i\| > \|\alpha^k\|$ and thus $d(\alpha^k, \beta^k, x^*) > d(\alpha^i, \beta^i, x^*)$.

A simple variation of the Euclidean distance can be obtained by modifying the norm for variables with index j satisfying $x_j^* = 0$. Define the vector $\bar{\alpha} \in \mathbb{R}^n$ with

$$\bar{\alpha}_j = \begin{cases} \alpha_j & \text{if } x_j^* \neq 0, \\ 0 & \text{otherwise,} \end{cases} \tag{14}$$

for $j \in N$. The *adjusted distance* is given by

$$ad\,(\alpha, \beta, x^*) = \frac{v\,(\alpha, \beta, x^*)}{\|\bar{\alpha}\| + 1}. \tag{15}$$

In order to avoid division by zero, we increase the denominator by one.

Distance Variant We now consider the deviation of the LP solution x^* from a given cut hyperplane in each variable direction and multiply them. In other words, given the LP solution x^*, let the point ρ_k be the intersection of the ray starting in x^* in coordinate direction k with the cut hyperplane $\alpha^T x = \beta$. The vectors x^* and ρ_k only differ in the k^{th} component by the amount Δx_k^* which can be obtained by calculating

$$\Delta x_k^* = \frac{\alpha^T x^* - \beta}{|\alpha_k|}, \tag{16}$$

where $k \in \bar{N} = \{j \in N : |\alpha_j| > 0\}$. To simplify the notation we assume that $\bar{N} = \{1, \ldots, r\}$ in the following.

Multiplying the deviations Δx_k^* for all $k \in \bar{N}$ yields

$$\prod_{k \in \bar{N}} \Delta x_k^* = \Delta x_1^* \cdot \Delta x_2^* \cdots \Delta x_r^* = \frac{\left(\alpha^T x^* - \beta\right)^r}{|\alpha_1 \cdot \alpha_2 \cdots \alpha_r|}. \tag{17}$$

In two-dimensional space, the product of the deviations computed in Equation (17) is twice the area of the triangle described by x^* and its projections ρ_k on the cut hyperplane $\alpha^T x = \beta$.

Typically, not all generated cutting planes have non-zero coefficients on the same number of variables. In order to improve the comparability we extract the r^{th} root and raise the resulting expression to the power of $k \geq 1$. We obtain

$$dv\left(\alpha,\beta,x^*,k\right)=\frac{v\left(\alpha,\beta,x^*\right)^k}{\left(\left|\alpha_1\cdot\alpha_2\cdots\alpha_r\right|\right)^{\frac{k}{r}}}. \tag{18}$$

Note that we assume that x^* violates the inequality $\alpha^T x \leq \beta$. Otherwise even values of k would change the sign of the numerator of Equation (18). It is easy to show that (18) is invariant to scaling.

Objective Function Parallelism Suppose that c and α are linearly dependent, for example $\alpha = kc$, $k \in \mathbb{R}$ and $k > 0$. After substitution of α the cut (3) reads $(kc)^T x \leq \beta$ which is equivalent to $c^T x \leq \frac{\beta}{k}$. This means that $\frac{\beta}{k}$ is an upper bound on the value of the objective function. Therefore cutting planes with α *as parallel as possible to the objective function* are preferable. Parallelism is measured by inspecting the cosine of the angle between c and α:

$$o\left(\alpha\right)=\frac{\alpha^T c}{\left\|\alpha\right\|\left\|c\right\|} \tag{19}$$

If $o(\alpha) = 1$, the cutting plane is parallel to the objective function.

Expected Improvement An intuitive method for measuring the quality of a cut is to inspect the improvement of the value of the objective function after the cut has been added and the LP has been reoptimized. Due to its computational cost (add cut, reoptimize, remove cut) this approach is normally not practicable. Although the exact improvement of the value of the objective function is therefore unknown, the improvement can be estimated under certain assumptions. Let x^*_{exp} be the *expected next LP solution* (cf. Christof and Reinelt [12]). As we are maximizing, we expect the value of the objective function to decrease (or remain the same) after a violated cut has been added, i.e. $c^T x^* - c^T x^*_{exp} \geq 0$. Taking $-\alpha$ as the direction of reoptimization, e.g. $x^*_{exp} = x^* - \delta\alpha$, and assuming that $\alpha^T x^*_{exp} = \beta$ gives

$$\delta=\frac{\alpha^T x^*-\alpha^T x^*_{exp}}{\left\|\alpha\right\|^2}=\frac{\alpha^T x^*-\beta}{\left\|\alpha\right\|^2}=\frac{v\left(\alpha,\beta,x^*\right)}{\left\|\alpha\right\|^2}. \tag{20}$$

The value of the objective function for x^*_{exp} takes the form

$$c^T x^*_{exp}=c^T x^*-\frac{\alpha^T c\cdot v\left(\alpha,\beta,x^*\right)}{\left\|\alpha\right\|^2}=c^T x^*-\frac{\alpha^T c}{\left\|\alpha\right\|}\cdot d\left(\alpha,\beta,x^*\right), \tag{21}$$

and the quality of a cut can, for instance, be measured by calculating

$$\begin{aligned} e\left(\alpha,\beta,x^*\right)=c^T x^*-c^T x^*_{exp}&=\frac{\alpha^T c}{\left\|\alpha\right\|}\cdot d\left(\alpha,\beta,x^*\right),\\ &=\left\|c\right\|\cdot o\left(\alpha\right)\cdot d\left(\alpha,\beta,x^*\right). \end{aligned} \tag{22}$$

It is quite easy to see from Equation (22) that the expected improvement is a combination of other previously defined quality measures. Not only is the prediction of the improvement of the objective function highly dependent on the assumptions, measuring cut quality this way is problematic when the objective gap between the optimal solution of the LP relaxation and the optimal MIP solution is zero.

Parallelism Let $(\alpha^i)^T x \leq \beta^i$ and $(\alpha^k)^T x \leq \beta^k$ be two cuts generated by some cut separation routine. If their defining hyperplanes are parallel, one of the two cuts is dominated by the other and should consequently not be added to the LP relaxation. Again, the absolute value of the cosine of the angle between the two hyperplanes (or vectors) is used to detect *parallelism*:

$$p\left(\alpha^i, \alpha^k\right) = \frac{\left|\left(\alpha^i\right)^T \alpha^k\right|}{\|\alpha^i\| \, \|\alpha^k\|} \tag{23}$$

If $p(\alpha^i, \alpha^k) = 1$, the hyperplanes which correspond to the two given cuts are parallel. In contrast, $p(\alpha^i, \alpha^k) = 0$ indicates that the two cuts are orthogonal.

Support The support of the coefficient vector α is the set of its non-zero positions $\bar{N} = \{j \in N : |\alpha_j| > 0\}$. One can compute the percentage of all variables that is present in the cut

$$s\left(\alpha\right) = \frac{\left|\bar{N}\right|}{|N|}. \tag{24}$$

Cutting planes which have a coefficient vector α which has a large value of $s(\alpha)$ are called *dense*. Dense cuts are likely to slow down computations in the simplex algorithm (e.g. the LU factorization) and can possibly cause numerical difficulties. Since adding these cuts is thus not desirable, we select cuts with minimal support, i.e. we use the quality measure $1 - s(\alpha)$.

Integral Support Let the set $\bar{N}$ be defined as in the previous paragraph. The set $\bar{N}_I = \bar{N} \cap N_I$ consists of those non-zero positions of α which correspond to integer-constrained variables. Using this information it is possible to calculate the ratio between the number of integer-constrained variables and the total number of variables present in a cut

$$i\left(\alpha\right) = \frac{\left|\bar{N}_I\right|}{\left|\bar{N}\right|}. \tag{25}$$

One may argue that a cut having non-zero coefficients on many (possibly fractional) integer variables is preferable to a cut which consists mostly of continuous variables.

3 Cut Pool and Cut Selection Algorithm

The term *cut pool* (cf. Padberg and Rinaldi [29]) is used for a data structure which is designed to offer an intermediate storage area for generated cutting planes. In our implementation the cut pool uses a packed storage, meaning that only non-zero entries and their indices are stored. All cuts are stored as less-or-equal inequalities, i.e. $\alpha^T x \leq \beta$.

Several cutting plane techniques are available in MOPS. So far, all generated cuts which are violated by the current solution are added to the LP relaxation. That this policy dramatically increases the problem size motivated the idea of studying and implementing cut management and selection techniques.

The MOPS MIP solver generates cuts at the root node of the branch-and-bound tree (cut-and-branch). Prior to the branch-and-bound algorithm the so-called supernode processing (cf. Suhl [30]) or IP preprocessing is performed. MOPS separates clique [15], bound implication [30], cover [20], flow cover [32], mixed-integer rounding [26], Gomory mixed-integer [19], lift-and-project [7, 8, 10], reduce-and-split [5, 14][1], pivot-and-reduce [35], strong Chvátal-Gomory [23], $\{0, \frac{1}{2}\}$-Chvátal-Gomory cuts [22] and two-row cuts [34] with the objective of tightening the LP relaxation. The cut generation process is iteratively continued until there is no improvement in the dual bound or a maximum number of rounds (typically twenty) is reached. There is no additional scheme for deactivating specific cut separators. Cut pool and cut selection are at the moment only used at root node. However, it should not take too much effort to adapt the implementation in such a way that it can be used at any branching node.

In the following we discuss the details of our implementation.

- In order to minimize work related to changes in the existing implementation of supernode processing, we changed the cut separation routines in such a way that they optionally add violated cuts to the cut pool instead of adding them directly to the LP relaxation. If possible, coefficients are scaled to integer values whenever a cut is added to the cut pool to avoid numerical problems.
- Quality measures which can be computed independently of the LP solution are computed and stored when a cut is added to the cut pool, i.e. support, integral support and objection function parallelism. We also compute the norm of the cut coefficient vector, the norm of the rotated cut coefficient vector (see Equation (8)) and the absolute value of the product of the cut coefficients (see Equation (18)).
- Quality measures whose values depend on the solution of the LP relaxation need to be updated after each LP resolve. In our implementation we recompute the violation (see Equation (4)), the adjusted norm (see Equation (15))

[1] Andersen et al. [5] and Cornuéjols and Nannicini [14] propose different variants of the reduce-and-split cut separator. To distinguish between these two, we call the original separator proposed by Andersen the reduce-and-split separator and the one proposed by Cornuéjols and Nannicini the reduce-and-split2 separator.

and the (rotated) distance with bounds (see Equations (10) and (11)). All other quality measures can be calculated from these values.

- If a cut $\alpha^T x \leq \beta$ is added to the cut pool, we compute the rotated cut coefficient vector $\bar{\alpha} = \alpha + D^T \lambda$ as defined in Equation (8) by solving the system
$$DD^T \lambda = -D\alpha \tag{26}$$
for λ. In order to be able to do this efficiently, we initially perform an LU decomposition of DD^T and then solve each system by forward transformation.
- We apply an active set algorithm proposed by Voglis and Lagaris [33] to compute the distance with bounds (see Equation (10)).
- After each round of cut generation, in which different cutting plane separators are executed, the cut selection is performed and the cut pool is scanned for cutting planes which seem to be promising with respect to some quality measure. If a cut is selected and *activated*, i.e. added to the problem formulation, it still remains in the cut pool. But in order to avoid adding this cut again, we mark it as activated. Having added the selected cuts, the LP is reoptimized and a new round of cut generation starts.
- We use the algorithm of Tomlin and Welch [31] to identify duplicate (i.e. exactly parallel) cuts. This algorithm is a very fast and specialized method which was originally developed to identify duplicate rows in an LP matrix in LP preprocessing. As this algorithm only works on a columnwise representation of the cut pool, we need to perform a sparse transpose.
- Suppose l cuts have been added to the cut pool and the cut pool has been sorted by non-ascending quality values. Furthermore, given $\delta = 1, \ldots, l-1$, let
$$\left(\alpha^i\right)^T x \leq \beta^i, \quad i = \text{poolsrt}[\delta] \tag{27}$$
be the cut of δ^{th} best quality and let
$$\left(\alpha^k\right)^T x \leq \beta^k, \quad k \in \{\text{poolsrt}[\delta+1], \ldots, \text{poolsrt}[l]\}, \tag{28}$$
be the set of cuts with smaller quality values. To reduce the size of the cut pool and increase the diversity of cuts we perform a check for parallelism which exploits the order of the cut pool. Thus we require all pairs (i, k) of cuts as defined in Equations (27) and (28) to satisfy the inequality
$$p\left(\alpha^i, \alpha^k\right) \leq \texttt{p\_max}. \tag{29}$$
We thereby remove all cutting planes with index k which are (almost) parallel to a better cut with index i from the cut pool.
- In order to avoid removing high-quality cuts from the cut pool, we additionally check whether the quality of such a cut is relatively large. If the quality value is larger than `skip_factor` times `best_qual` and the parallelism $p(\alpha^i, \alpha^k)$ of the two cuts is not larger than `p_max_ub` cut k is not removed.

- If $\alpha^i = w\alpha^k$ with $w \in \mathbb{R}$, then the hyperplanes $(\alpha^i)^T x = \beta^i$ and $(\alpha^k)^T x = \beta^k$ are parallel. This implies that $p(\alpha^i, c) = p(\alpha^k, c)$, i.e. being immediately parallel to the objective function is a required condition for the parallelism of two cutting planes. This detail can be used to save computational effort when checking for duplicate cuts.
- Whenever a cutting plane is found to be parallel to, a duplicate of or dominated by another cut it is marked as *deleted* instead of being directly removed from the cut pool. All parts of the algorithm simply ignore cuts which are marked as deleted. If an additional cut cannot be added to the cut pool because the maximum number of cuts in the cut pool is reached, a compression routine is executed. This compression routine efficiently removes all cuts which are marked as deleted from the cut pool.
- We require cuts which are inserted into the LP relaxation to have a minimum quality. Following suggestions described in [6] we do not fix this minimum quality `min_qual` a priori. At the first call of the selection routine `min_qual` is set to the minimum of `min_qual_ub` and 50% of the quality of the best cut in the cut pool. Whenever the cut pool does not contain cuts having the required minimum quality, the counter `fail` is incremented. If `fail` takes the value two, meaning that the cut pool did not contain adequate cuts in two rounds, `min_qual` is reduced to allow for cuts with smaller quality values.
- The parameters `total_factor` and `round_factor` restrict the number of cuts added to the LP relaxation during one round of cut generation and the whole supernode processing respectively. Our implementation does not allow for the total number of cuts (or number of cuts per round) to be larger than `total_factor` (or `round_factor`) times m, which is the number of constraints in the initial LP formulation.
- Cutting planes are *aged out* of the cut pool, meaning that cuts whose quality values were not large enough to be added to the LP relaxation in `max_age` rounds are deleted.
- Non-binding constraints are removed from the LP relaxation. Additional techniques such as aging are not used in the LP context. In particular, this means that cuts are not transferred from the LP formulation back to the cut pool.

4 Computational Experiments

In the preceding sections we discussed various cut quality measures and presented a cut selection algorithm. In this section we evaluate computationally the effectiveness of these cut quality measures. We further assess the effect that our cut selection algorithm has on the overall performance of Mops. The computations were performed on a subset of the instances from the following sources: ACC [28], CORAL [25], MIPLIB [3, 11, 21] and MILP [27]. Our test set consists of the 143 instances. We imposed a maximum computation time of one hour per instance. In order to obtain comparable results and to reduce the variability between test runs, we disabled the heuristics of Mops and provided the optimal value of the objective function as a cutoff value. The tests runs were conducted on an Intel Core 2 Duo PC with 3.16 GHz and 8 GB of main memory.

Apart from choosing a reliable cut quality measure, a number of parameters need to be set for our cut selection algorithm. In a series of preliminary experiments, we tested different default settings for these parameters while keeping the cut quality measure fixed. We first only limited the number of cutting planes added to the LP relaxation by setting the parameters `total_factor` and `round_factor` to the values 1.0 and 0.1 respectively. The results obtained with this parameter setting were not satisfying. After some tuning we found the following default setting which we used to compute all results presented in this paper. In particular, we pursue a quite aggressive strategy to reduce the number of cutting planes in the cut pool. Duplicate cuts are identified and even fairly parallel cutting planes (`p_max` $= 0.1$) are not considered. We avoid removing cuts with a relatively high quality (`skip_factor` $= 0.9$, `p_max_ub` $= 0.5$). In addition, we try to reduce the size of the cut pool by restricting the maximum age of a cut to three (`max_age` $= 3$). The minimum quality value we require a cut to have is 0.01 (`min_qual_ub` $= 0.01$). It turned out to be unnecessary further to limit the number of cuts added to the LP relaxation (`total_factor` $= \infty$, `round_factor` $= \infty$). Note that we do not examine the effect of each single parameter on the overall performance. These settings are therefore still first choices likely to be improved by further experiments.

We next investigate the individual computational effectiveness of the cut quality measures we presented. We compare the variants using a specific quality measure with a reference setting which does not apply a cut selection algorithm. To enhance readability we only present cumulative results in this section. In order to compare the performance of the different configurations, we compute the shifted geometric means (see Achterberg [2]) of the solution times, numbers of branching nodes, amounts of integrality gap closed at the root node and numbers of cuts added to the LP relaxations. To compute the shifted geometric means, we set the shifting parameter $s = 100$ for node counts, $s = 10$ for solution times, $s = 0.05$ (i.e. 5%) for integrality gaps and $s = 1$ for the number of cutting planes added to the LP relaxations. If an instance is not solved within the time limit, we assume a computation time of one hour and treat the current number of computed branching nodes as the final node count in our evaluations. For instances which do not have an integrality gap we assume a value of 1 (i.e. 100% gap closed) in the calculation of the mean values. Table 4 shows the percentage changes in the shifted geometric means calculated for the cut quality measures relative to the mean values obtained if no cut selection algorithm is applied. The detailed results can be found in Tables 6 to 10 in the appendix.

An immediate observation from Table 4 is that the performance of an LP-based cut-and-branch MIP solver like Mops can considerably be improved by carefully selecting only a subset of the cutting planes generated during IP

	test set	violation	rel. violation	distance	rot. distance	distance w/ bounds	rot. distance w/ bounds	obj. parallelism	exp. improvement	support	int. support	adj. distance	distance var.
time	ACC	+19	+339	-46	+356	-47	+375	+296	+296	+38	-9	-38	+379
	CORAL	-8	-18	-19	-14	-22	-13	+10	-15	+3	-6	-17	-14
	MIPLIB	-29	-35	-44	-43	-39	-38	-20	-27	-26	-30	-44	-33
	MILP	-32	-40	-32	-37	-36	-37	-26	-51	-25	-45	-32	-40
	total	-16	-23	-27	-23	-28	-20	-1	-23	-8	-19	-25	-21
nodes	ACC	+57	+83	-46	+40	-44	+179	+64	+64	+77	+12	-33	+105
	CORAL	+59	+44	+34	+38	+28	+41	+119	+48	+71	+68	+39	+53
	MIPLIB	-1	+3	-8	-24	-24	-21	+64	+34	-3	-11	-19	+16
	MILP	+55	+40	+49	+31	+37	+39	+91	+17	+33	+3	+41	+41
	total	+45	+35	+26	+23	+16	+28	+102	+40	+48	+37	+25	+44
gap	ACC	0	0	0	0	0	0	0	0	0	0	0	0
	CORAL	+5	+4	+3	+4	+4	+5	+20	+12	-4	+1	+5	+7
	MIPLIB	+7	+6	+17	+8	+4	+2	+29	+20	+1	+3	+6	+12
	MILP	+1	-9	0	-4	0	-6	+7	+11	-7	-6	+3	-1
	total	+4	+2	+5	+4	+3	+3	+19	+13	-4	0	+5	+6
cuts	ACC	-75	-90	-78	-76	-74	-77	-99	-99	-74	-71	-77	-77
	CORAL	-87	-88	-89	-88	-87	-87	-93	-91	-73	-72	-88	-90
	MIPLIB	-81	-86	-87	-85	-85	-86	-91	-91	-75	-78	-86	-86
	MILP	-90	-91	-93	-93	-93	-93	-95	-96	-80	-79	-92	-92
	total	-87	-88	-89	-89	-88	-88	-93	-92	-75	-74	-88	-89

Table 4 Performance impact of using particular cut quality measures. The values represent percentage changes in the shifted geometric mean compared with a reference setting in which no cut selection scheme is activated. Positive values indicate a deterioration while negative values signify an improvement.

preprocessing. Concerning the time spent on solving the instances in our test sets to optimality, our cut selection algorithm mostly leads to performance improvements in terms of the relative shifted geometric mean values. The ACC instances, however, are very prone to changes in the cut selection algorithm. For these instances the solution times can be reduced considerably if cut quality is measured by distance ("distance") or distance with bounds ("distance w/ bounds"). On the other hand, other quality measures lead to a dramatic deterioration in performance with respect to solution times. Aside from the ACC instances, the only increases in the solution times that Table 4 reports on arise from using the objective parallelism ("obj. parallelism") or support ("support") to measure cut quality for the CORAL instances. The effect of our cut selection algorithm on the enumeration during the branch-and-bound algorithm varies across the test sets and quality measures. The reference setting which adds all cutting planes generated during IP preprocessing to the LP relaxation in many cases computes the smallest number of nodes during the branch-and-bound search. Increases in the node counts are not surprising since our cut selection algorithm strongly reduces the number of cutting planes added to the LP relaxation, thereby affecting its tightness. The shifted geometric means of the number of cutting planes added to the LP relaxation decrease at least by about 70%. The search space the branch-and-bound algorithm needs to explore thus increases if the MIP formulation produced by our cut selection algorithm has a lower quality. The number of branching nodes, for example, increases by 119% for the CORAL instances if cut quality is measured by the objective parallelism. This increase corresponds to the increase in the running times mentioned above. Most of the other, sometimes dramatic, increases in the number of branching nodes do not, however, lead to increased running times. The greater part of the branch-and-bound algorithm's running time is typically spent on solving LP relaxations. Cutting planes increase the size of these LP relaxations, making them more difficult to solve. Our computational experiments indicate that the number of constraints in the LP relaxation is considerably increased by adding all cutting planes generated during IP preprocessing, which repeatedly leads to higher node solution times during the branch-and-bound process. If, on the other hand, only a careful subset of the generated cuts is added, solving the LP relaxation becomes computationally less expensive. Since the LP relaxation is solved for every branching node, the performance gain in LP optimization achieved by adding only selected cuts is likely to overcompensate the loss in the quality of the MIP formulation. This is also shown by the results in Table 4 where the overall solution times decrease even if the number of branching nodes increases. For instance, the configuration "distance w/ bounds" computes on average 16% more branching nodes while the average solution time decreases by 28%. Concerning the amounts of integrality gap closed at the root node, our results show that most cut quality measures produce dual bounds of good quality. Concerning the entire test set, considerable losses in the quality of the dual bound can be observed if cut quality is measured by the objective parallelism or expected improvement. On the other hand, some quality measures close more integrality gap than the reference setting although fewer cuts are added. For the MILP instances using relative violation ("rel. violation") increases the shifted geometric mean of the amounts of integrality gap closed at the root node by 9%.

Concerning solution times, Table 4 shows that the three distance-based quality measures proposed in the literature, i.e. distance, rotated distance and distance with bounds, and their combination rotated distance with bounds produce quite similar results in terms of the shifted geometric means. Figure 1 shows performance profiles (cf. Dolan and Moré [16]) of solver configurations using these quality measures. Figure 1 confirms the results reported in Table 4 and indicates that there is no clear winner among these quality measures. The configuration using distance is the fastest of the four configurations on about 32% of the instances in the test set. Rotated distance is fastest on about 31%, distance with bounds on about 42% and rotated distance with bounds on about 22% of the instances. Among all four configurations, the

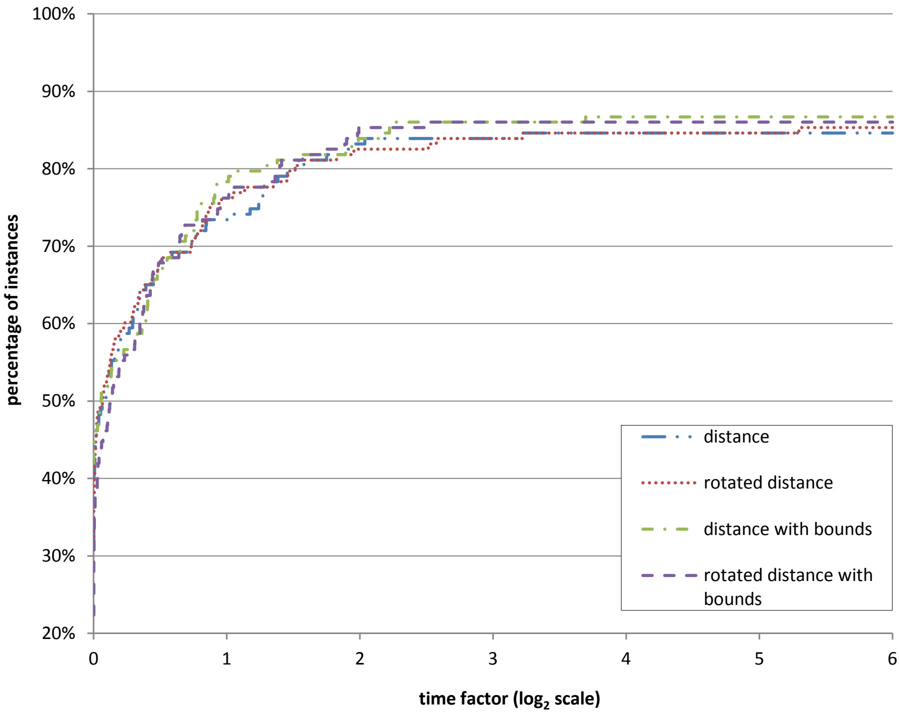

Fig. 1 Performance profiles of distance variants.

one using distance with bounds solves the largest percentage, namely 87%, of the instances to optimality within the time limit. With regard to the time needed to solve specific percentages of instances to optimality, Figure 1 shows that distance with bounds solves 80% of the instances to optimality within a time factor of about 2.4 of the best configuration. The configuration using rotated distance, however, solves 62% of the instances within a time factor of about 1.2 of the best configuration. Moreover, rotated distance with bounds solves 85% of the instances to optimality within a time factor of about 4 of the best configuration.

Figure 2 shows performance profiles of solver configurations measuring cut quality by plain distance, adjusted distance (15) and the distance variant (18). There again is no clear winner. While distance is fastest on about 47% of the instances in the test set and solves 85% percent of the instances to optimality within the time limit, the configuration using adjusted distance is fastest on 29% of the instances and succeeds in solving 87% of the instances to optimality.

We next consider the performance gain that can be achieved by applying our cut selection algorithm on a per-instance level. Figure 3 depicts performance profiles of two solver configurations. The first configuration adds all generated cutting planes to the LP relaxation while the second one applies our cut selection algorithm using distance (6) as cut quality measure. The configuration which does not use a cut selection algorithm is fastest on about 28% and solves 84%

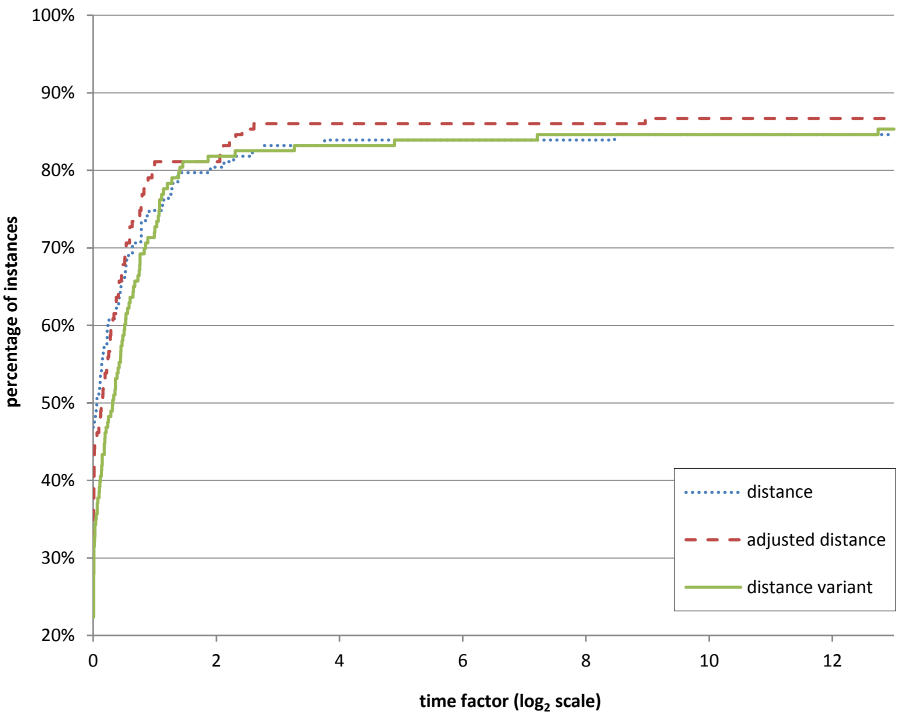

Fig. 2 Performance profiles of distance variants (2).

of the instances in the test set to optimality within the time limit. The configuration applying our cut selection algorithm is fastest on 64% of the instances in the test set. Figure 3 also shows that this configuration solves 85% of the instances to optimality within the time limit and 78% of them within a time factor of about 2 of the best configuration.

Achterberg [1, 2] proposes to measure cut quality by a weighted sum of distance, objective function parallelism and relative orthogonality. Relative orthogonality is defined as the minimal orthogonality between a given cut and all other cuts which are of higher quality. In the following we consider a weighted sum of distance (6), integral support (25) and objective parallelism (19) where the first quantity is weighted with 1.0 and the latter ones with 0.1. In our implementation orthogonality (or parallelism) does thus not have a direct influence on the quality value, i.e. is not part of the weighted sum. We use the parallelism condition to remove cuts a priori. Figure 4 depicts performance profiles of two solver configurations. The first one measures cut quality based on distance while the second one computes the weighted sum. Figure 4 shows that distance is fastest on about 46% and solves 85% of the instances in the test set to optimality within the time limit. The configuration using the weighted sum, on the other hand, is fastest on about 50% and solves 89% of the instances to optimality.

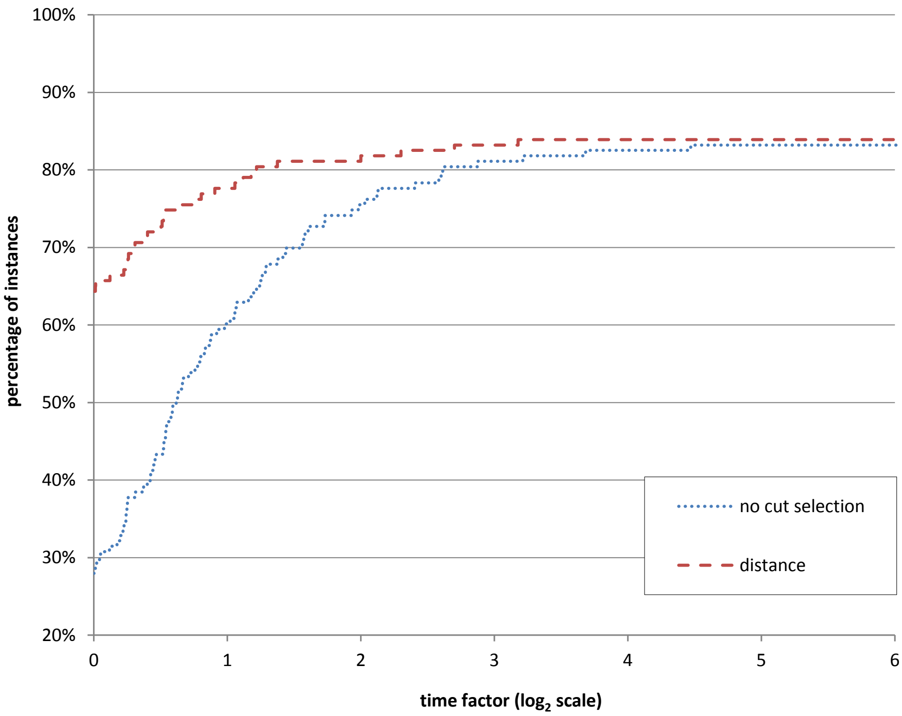

Fig. 3 Performance profiles of configurations using quality measure distance and no cut selection algorithm.

	Gomory		implication		clique		cover		flow cover		MIR		strong CG		lift & project		reduce & split		reduce & split2		pivot & reduce		zero-half		two-row	
separator	D	P	D	P	D	P	D	P	D	P	D	P	D	P	D	P	D	P	D	P	D	P	D	P	D	P
Gomory	32	10		4		1		2		1		4	4	8		6	4	9	2	9	1	5		1	2	7
implication		5	6	47		2		15		13		6		3		5		6		4		6		2		4
clique		1		1	3	15	8	1						1		2						1				1
cover		1	1		19	1	19	15				1		4						1		1		2		3
flow cover		2		1					3	32		9				2		2		3		3				2
mir		6		7		5		4		6	7	30		2		10		8		10		12				8
strong CG		2											38	9		1										1
lift & project		9		1		2		3		2		5		3	13	26		16		14	1	13		7		9
reduce & split		1															12	1	9			1				1
reduce & split2		2												1		2	9	4	14	3		2				2
pivot & reduce		7		2				3		3		5		3		9	2	10	2	11	22	12				7
zero-half		1				1		3						7		4		1		1		1	70	12		3
two-row		9		7				4		2		7		6		5		8		9		5		2	37	11
	32	56	7	70	22	27	27	50	3	59	7	67	42	47	13	72	27	65	27	65	24	62	70	26	39	59
	88		77		49		77		62		74		89		85		92		92		86		96		98	

Table 5 Percentages of cutting planes removed from cut pool as duplicates (D) or due to parallelism (P).

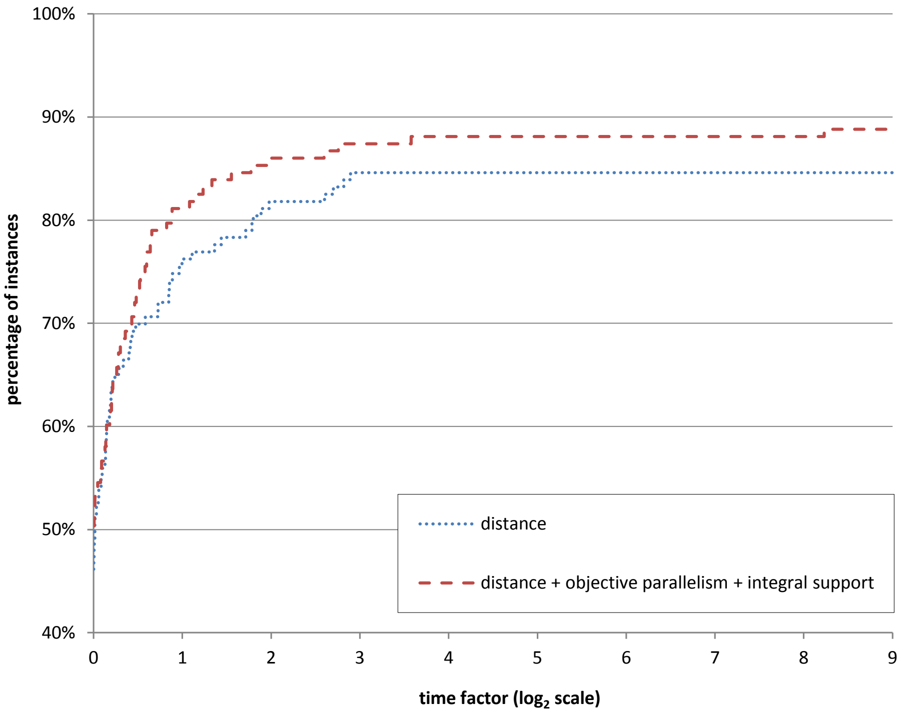

Fig. 4 Performance profiles of configurations using quality measure distance and a weighted sum of three quality measures.

The overall reduction of the number of cutting planes is remarkable, but does not give insight into the performance of specific cutting plane separators within the cut selection framework. Specifically, we want to answer the question whether certain cutting planes are more sensitive to cut selection schemes than other ones. Table 5 shows statistics about the removal of cuts violating the maximum parallelism constraint (29) and duplicate cuts. Duplicate cuts can, for instance, arise if a specific separator generates the same cut in several rounds of IP preprocessing. Each column in Table 5 represents a specific cut separation algorithm (separator). For each of these separators the percentage of the generated cuts that is removed is shown. The corresponding rows show the separators that cause the removal. Table 5 shows that a large fraction of the generated cuts is removed from the cut pool, e.g. 85% of the lift-and-project cuts. It also indicates that the parallelism checks considerably reduce the number of cuts in the cut pool. For instance, 16% of the generated reduce-and-split cuts are removed from the cut pool because of parallelism to lift-and-project cuts. An interesting point is that about 30% of all generated MIR cuts are removed due to parallelism to other MIR cuts of higher quality. In state-of-the-art MIP solvers MIR cuts are separated heuristically using constraint aggregation. The number of cuts produced by this heuristic is usually quite large as cuts are generated from rows of the LP as well as from arbitrary combinations of these rows. Furthermore, the heuristic is likely to come up with many “similar” cuts due to the fact that partially the same rows were aggregated. The computational results indicate that our cut selection algorithm is especially helpful in dealing with such a separation heuristic. An additional information provided by Table 5 is that some separation routines are more likely to generate duplicate cuts than other ones. For instance, about 9% of all generated reduce-and-split2 cuts are removed from the cut pool due to the fact that they are duplicates of reduce-and-split cuts. These separation algorithms use different strategies to aggregate rows of the LP tableau. Our cut selection algorithm can reduce the redundancies which are created by both separation algorithms.

5 Conclusion

In this paper, we presented a cut selection algorithm and tested cut quality measures for their practical value. Our computational experiments showed that using this cut selection algorithm can reduce the running times. We also demonstrated that the reduction of the number of cutting planes is an important part of such algorithms. Moreover, we showed that our algorithm can be especially helpful when there are cut separators that tend to generate “similar” cuts. The developed algorithm is at the moment only used at the root node of the branch-and-bound tree. Future work could investigate the efficient usage of the cut pool and cut selection algorithm within a branch-and-cut algorithm.

References

1. Tobias Achterberg. *Constraint Integer Programming*. PhD thesis, Technische Universität Berlin, 2007.
2. Tobias Achterberg. SCIP: solving constraint integer programs. *Mathematical Programming Computation*, 1(1):1–41, 2009.
3. Tobias Achterberg, Thorsten Koch, and Alexander Martin. MIPLIB 2003. *Operations Research Letters*, 34(4):1–12, 2006. `http://miplib.zib.de`.
4. Edoardo Amaldi, Stefano Coniglio, and Stefano Gualandi. Coordinated cutting plane generation via multi-objective separation. *Mathematical Programming*, pages 1–24, 2012.
5. Kent Andersen, Gérard Cornuéjols, and Yanjun Li. Split closure and intersection cuts. *Mathematical Programming*, 102(3):457–493, 2005.
6. Giuseppe Andreello, Alberto Caprara, and Matteo Fischetti. Embedding $\{0, \frac{1}{2}\}$-Cuts in a Branch-and-Cut Framework: A Computational Study. *INFORMS Journal on Computing*, 19(2):229–238, 2007.
7. Egon Balas and Pierre Bonami. Generating lift-and-project cuts from the LP simplex tableau: open source implementation and testing of new variants. *Mathematical Programming Computation*, 1(2–3):165–199, 2009.
8. Egon Balas and Michael Perregaard. A precise correspondence between lift-and-project cuts, simple disjunctive cuts, and mixed integer Gomory

cuts for 0-1 programming. *Mathematical Programming*, 94(2–3):221–245, 2003.
9. Egon Balas, Sebastián Ceria, Gérard Cornuéjols, and N. Natraj. Gomory cuts revisited. *Operations Research Letters*, 19:1–9, 1996.
10. Egon Balas, Gérard Cornuéjols, Tamás Kiz, and Giacomo Nannicini. Combining Lift-and-Project and Reduce-and-Split. Technical report, Tepper School of Business, Carnegie Mellon University, 2011.
11. Robert E. Bixby, Sebastián Ceria, Cassandra M. McZeal, and Martin W. P. Savelsbergh. An updated mixed integer programming library: MIPLIB 3.0. *Optima*, 58:12–15, 1998. `http://www.caam.rice.edu/~bixby/miplib/miplib.html`.
12. Thomas Christof and Gerhard Reinelt. Algorithmic aspects of using small instance relaxations in parallel branch-and-cut. *Algorithmica*, 30(4):597–629, 2001.
13. William J. Cook, Ricardo Fukasawa, and Marcos Goycoolea. Effectiveness of different cut selection rules. presented at the workshop on Mixed Integer Programming, Miami, Florida, 2006.
14. Gérard Cornuéjols and Giacomo Nannicini. Practical strategies for generating rank-1 split cuts in mixed-integer linear programming. *Mathematical Programming Computation*, 3(4):281–318, 2011.
15. Harlan Crowder, Ellis L. Johnson, and Manfred W. Padberg. Solving large-scale zero-one linear programming problems. *Operations Research*, 31(5):803–834, 1983.
16. Elizabeth D. Dolan and Jorge J. Moré. Benchmarking optimization software with performance profiles. *Mathematical Programming*, 91::201–213, 2002.
17. Michael C. Ferris, Gábor Pataki, and Stefan Schmieta. Solving the seymour problem. *Optima*, 66:2–6, 2001.
18. Matteo Fischetti and Domenico Salvagnin. A relax-and-cut framework for Gomory mixed-integer cuts. *Mathematical Programming Computation*, 3 (2):79–102, 2011.
19. Ralph E. Gomory. An algorithm for the mixed integer problem. Technical Report RM-2597, The RAND Cooperation, 1960.
20. Zonghao Gu, George L. Nemhauser, and Martin W. P. Savelsbergh. Lifted Cover Inequalities for 0-1 Integer Programs: Computation. *INFORMS Journal on Computing*, 10(4):427–437, 1998.
21. Thorsten Koch, Tobias Achterberg, Erling Andersen, Oliver Bastert, Timo Berthold, Robert E. Bixby, Emilie Danna, Gerald Gamrath, Ambros M. Gleixner, Stefan Heinz, Andrea Lodi, Hans Mittelmann, Ted Ralphs, Domenico Salvagnin, Daniel E. Steffy, and Kati Wolter. MIPLIB 2010. *Mathematical Programming Computation*, 3(2):103–163, 2011.
22. Arie M. C. A. Koster, Adrian Zymolka, and Manuel Kutschka. Algorithms to Separate $\{0, \frac{1}{2}\}$-Chvátal-Gomory Cuts. *Algorithmica*, 55(2):375–391, 2009.
23. Adam N. Letchford and Andrea Lodi. Strengthening Chvátal-Gomory cuts and Gomory fractional cuts. *Operations Research Letters*, 30(2):74–82, 2002.

24. Jeffrey T. Linderoth. *Topics in Parallel Integer Optimization.* PhD thesis, Georgia Institute of Technology, 1998.
25. Jeffrey T. Linderoth and Theodore K. Ralphs. Noncommercial Software for Mixed-Integer Linear Programming. In John K. Karlof, editor, *Integer Programming: Theory and Practice*, Operations Research Series, pages 253–303. CRC Press, 2005.
26. Hugues Marchand and Laurence A. Wolsey. Aggregation and mixed integer rounding to solve MIPs. *Operations Research*, 49(3):363–371, 2001.
27. Hans Mittelmann. Decision tree for optimization software: Benchmarks for optimization software, 2010. `http://plato.asu.edu/bench.html`.
28. George L. Nemhauser and Michael A. Trick. Scheduling A Major College Basketball Conference. *Operations Research*, 46(1):1–8, 1998.
29. Manfred W. Padberg and Giovanni Rinaldi. A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems. *SIAM Review*, 33(1):60–100, 1991.
30. Uwe H. Suhl. MOPS - Mathematical OPtimization System. *European Journal of Operational Research*, 72:312–322, 1994.
31. John A. Tomlin and James S. Welch. Finding duplicate rows in a linear programming model. *Operations Research Letters*, 5(1):7–11, 1986.
32. Tony J. van Roy and Laurence A. Wolsey. Solving mixed integer programming problems using automatic reformulation. *Operations Research*, 35(1): 45–57, 1987.
33. Costas Voglis and Isaac E. Lagaris. BOXCQP: An Algorithm For Bound Constrained Convex Quadratic Problems. In *1st International Conference "From Scientific Computing to Computational Engineering"*, 2004.
34. Franz Wesselmann. *Generating General-Purpose Cutting Planes for Mixed-Integer Programs.* PhD thesis, Universität Paderborn, 2011.
35. Franz Wesselmann, Achim Koberstein, and Uwe H. Suhl. Pivot-and-reduce cuts: An approach for improving gomory mixed-integer cuts. *European Journal of Operational Research*, 214(1):15–26, 2011.

A Tables

	no cut selection					violation					relative violation				
instance	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts
30_05_100	T	>3600.0	1795	100.0	54	T	>3600.0	168839	100.0	1	T	>3600.0	169055	100.0	1
30_95_100		386.4	1632	-	55		611.4	5416	-	0		611.4	5416	-	0
30_95_98		387.0	1236	100.0	58		94.8	833	100.0	1		94.8	833	100.0	1
acc1		25.8	360	-	164		34.2	681	-	35		22.8	584	-	19
acc3		196.8	1951	-	37		217.8	2669	-	11	T	>3600.0	4087	-	3
aflow30a		261.6	32309	50.2	751		208.2	52817	48.0	262		346.2	68643	43.9	273
air04		304.2	4530	11.9	69		259.8	2925	12.2	24		331.8	1627	17.7	21
air05		929.4	17984	3.7	71		313.2	16878	4.5	10		247.8	18896	5.1	9
alignninq		105.0	4932	23.5	255		147.6	3082	20.7	44		115.8	4749	19.4	51
bienst1		276.6	37379	11.7	324		157.8	43960	16.3	93		253.8	40343	45.8	149
bienst2		2096.4	306049	6.4	525		894.0	277381	11.5	135		541.2	222737	15.8	120
cap6000		177.6	4775	41.4	59		78.0	4177	12.7	21		19.8	4255	3.9	1
dcmulti		42.0	137	78.6	723		42.0	146	82.1	327		39.6	61	85.3	299
eilB101		344.4	46056	7.0	47		372.0	56806	7.9	23		278.4	54698	8.6	13
fiber		45.6	189	92.2	537		55.2	218	92.0	141		57.0	129	92.9	161
gesa2_o		198.0	25552	80.4	1368		94.2	21294	71.4	344		833.4	481685	69.1	240
gesa3_o		22.2	134	80.4	1211		44.4	201	80.6	245		42.0	401	76.3	165
harp2		2986.8	465573	62.7	1214		1036.2	678649	59.0	236		2344.8	1043083	59.1	258
l152lav		166.2	9817	19.3	241		81.6	9139	26.9	34		45.0	6974	26.9	33
map18		3371.4	2829	28.6	1295		3366.6	7596	19.5	202	T	>3600.0	7698	26.5	226
map20		1890.6	2786	25.4	977		1372.2	3336	20.2	169		1196.4	2441	21.7	205
markshare_4_0		1150.8	1948760	0.0	26		1611.6	2265834	0.0	3		876.0	1610042	0.0	5
mas76		604.2	958189	7.5	68		426.0	960103	4.0	4		400.8	805703	2.8	3
mitre		762.0	47536	69.4	3025		242.4	963	94.9	1300		63.0	670	96.1	1342
mod011		262.2	1665	77.4	2061		109.8	4725	63.4	730		103.8	3703	68.8	738
neos-1053591	T	>3600.0	5031256	0.0	178		2184.0	3903474	0.0	4		1898.4	3278805	0.0	4
neos-1061020		396.0	3250	4.7	146		275.4	4731	4.3	29		202.8	4265	4.4	4
neos-1122047		82.2	51	-	13		69.0	51	-	0		69.6	51	-	0
neos-1151496		27.6	536	100.0	65	T	>3600.0	1013377	83.3	12	T	>3600.0	332257	66.7	9
neos-1171448		66.0	285	-	261		28.8	152	-	1		29.4	152	-	1
neos-1171737	T	>3600.0	80538	-	139		141.0	2637	-	4		141.0	2637	-	4
neos-1200887		685.2	398282	0.0	2		307.2	184967	0.0	1		307.2	184967	0.0	1
neos-1211578		1075.2	794568	0.0	107		453.6	730529	0.0	4		552.6	866133	0.0	4
neos-1228986	T	>3600.0	1279180	0.0	227	T	>3600.0	3801736	0.0	20		3513.6	5332577	0.0	7
neos-1367061		162.0	125	100.0	241		94.8	121	100.0	40		164.4	125	100.0	37
neos-1396125	T	>3600.0	123549	14.2	693		2471.4	579647	26.7	68		1904.4	523173	8.5	27
neos-1420205		4.2	2356	45.0	236		1.2	2668	37.5	16		1.2	2394	37.5	12
neos-1440447		735.0	776305	0.0	25		476.4	635811	0.0	5		476.4	635811	0.0	5
neos-1460265	T	>3600.0	980281	-	613		3.0	832	-	182		7.2	3352	-	131
neos-1489999		1980.6	395301	3.6	650		654.0	333747	9.5	41		490.2	220235	10.7	40
neos-1582420	T	>3600.0	100915	12.4	329		1155.6	80114	12.4	34		1029.6	76664	41.6	45
neos-1597104		75.0	637	-	7		60.0	478	-	6		22.8	80	-	6
neos-430149		3571.2	3355565	0.0	195		1617.6	1834265	0.0	29		3120.6	3292987	0.0	69
neos-480878		825.0	64569	64.1	50		2445.0	182806	64.1	9		2756.4	204514	64.1	9
neos-503737		23.4	292	-	198		84.6	3056	-	26		302.4	10165	-	21
neos-504674		2604.6	871292	45.5	1067		1486.2	898224	36.9	248		1171.8	746433	35.8	248
neos-504815		654.0	290196	31.1	835		478.8	290290	28.7	252		380.4	237711	32.1	271
neos-512201		580.2	189371	46.9	995		671.4	311564	41.1	279		663.0	298216	42.4	327
neos-522351	T	>3600.0	296886	97.4	825	T	>3600.0	360195	85.2	280	T	>3600.0	625637	97.2	260
neos-525149		667.2	81529	0.0	475		313.8	58161	0.0	136		196.8	68629	0.0	55

continued on next page

instance	no cut selection					violation					relative violation				
	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts
neos-530627		1670.4	3427553	46.1	44		1561.8	3369401	44.6	2		2248.2	3403291	44.6	2
neos-538867		1749.0	521094	0.0	496		303.6	308981	0.0	32		405.0	431656	0.0	26
neos-538916		1168.8	435041	0.0	409		480.6	459855	0.0	36		869.4	766255	0.0	55
neos-551991		2861.4	96651	31.4	941		1542.0	210228	14.3	39		610.8	63155	31.4	60
neos-555694		21.6	1274	0.0	385		27.6	6579	0.0	25		26.4	5725	0.0	22
neos-555771	T	>3600.0	189048	0.0	661		103.2	34015	0.0	34		43.8	11827	0.0	26
neos-570431		54.6	10040	25.1	277		76.8	11611	25.1	43		60.6	16398	25.1	28
neos-583731		4.8	931	-	1098		1.8	149	-	171		42.6	27658	-	146
neos-584851		1310.4	364254	0.0	635		183.0	193452	0.0	38		232.8	238743	0.0	26
neos-585192		602.4	99806	60.8	307	F	>3600.0	643740	60.8	123	F	>3600.0	647415	60.8	123
neos-595905		99.0	161	97.3	1095	T	>3600.0	453104	67.3	392		2269.2	493350	23.3	343
neos-598183		307.2	11062	92.3	1651		2143.2	141498	37.6	384		2749.2	330092	36.7	380
neos-603073	T	>3600.0	175077	7.7	1492	T	>3600.0	203084	7.6	507	T	>3600.0	567046	6.9	445
neos-611838		318.6	3729	77.3	1151		339.6	11905	75.0	658		313.8	10577	73.1	619
neos-612125		174.0	801	81.0	1218		132.6	1985	76.5	634		130.2	1477	76.3	670
neos-612143		223.2	1315	81.2	1167		154.2	2697	78.8	637		138.6	1927	78.7	634
neos-612162		359.4	3429	79.7	1256		315.0	7745	76.8	646		225.6	6711	77.4	630
neos-686190		2454.6	94803	5.8	209		1068.6	98936	6.8	47		524.4	90231	5.7	20
neos-709469		1.2	366	-	338		1542.6	1720775	-	26		2001.6	2135048	-	30
neos-775946	T	>3600.0	420876	62.4	532	T	>3600.0	579883	55.0	156	T	>3600.0	737156	70.5	76
neos-777800		127.8	2296	-	78		113.4	10229	-	0		113.4	10229	-	0
neos-785914		85.2	4903	-	518		42.6	2505	-	90		1152.6	155120	-	85
neos-791021	T	>3600.0	88725	30.8	805	T	>3600.0	565596	7.7	70	T	>3600.0	249629	53.8	161
neos-796608		0.2	1	100.0	50	T	>3600.0	3349596	0.0	11		0.2	1	100.0	15
neos-803219		93.0	53987	26.8	131		87.0	56417	25.8	60		87.0	53539	26.3	73
neos-803220		244.8	80283	32.1	217		78.0	77045	32.7	43		66.6	79406	30.0	30
neos-806323		67.2	15690	30.7	273		64.8	17603	26.6	113		46.2	18454	29.4	99
neos-807639		66.6	22264	21.6	329		66.6	21592	21.3	136		52.8	24196	26.2	99
neos-807705		46.8	11061	18.2	308		41.4	8619	20.9	117		32.4	9500	19.3	96
neos-808072		45.6	531	98.6	86		103.2	363	98.6	49		79.2	791	98.6	29
neos-810286		274.8	3460	-	23		104.4	1716	-	3		93.0	1660	-	4
neos-820879	T	>3600.0	89015	23.2	199		2148.6	76406	62.4	92		1138.2	29986	70.5	103
neos-824661		1248.6	753	-	501	T	>3600.0	3619	-	128	T	>3600.0	3613	-	128
neos-824695		180.0	2064	-	402		77.4	480	-	68		77.4	480	-	68
neos-825075		381.6	44921	0.0	572		123.6	46458	0.0	99		98.4	31245	0.0	133
neos-826224	T	>3600.0	24550	-	264	T	>3600.0	26188	-	70	T	>3600.0	17248	-	70
neos-826250		80.4	357	-	482		1063.8	10134	-	68		1525.2	11466	-	68
neos-826812		42.6	2	-	272		74.4	269	-	48		74.4	269	-	48
neos-827175		25.8	28	-	118		59.4	47	-	34		59.4	47	-	34
neos-829552		1735.8	8939	1.9	141		1681.8	8775	3.8	16		1256.4	8179	3.6	18
neos-839859		154.2	12966	9.4	48		147.0	16328	7.0	24		117.0	12841	7.4	17
neos-860300		73.8	607	38.3	4		74.4	602	38.3	3		74.4	602	38.3	3
neos-863472		1699.2	2418403	0.0	149		1510.8	2451175	0.0	10		1408.8	2258266	0.0	11
neos-881765		79.8	44135	-	33		3.6	301	-	4		4.2	301	-	4
neos-885086		248.4	1564	-	187		107.4	552	-	6		195.6	764	-	6
neos-892255		711.0	11719	0.0	698		187.8	11332	0.0	15		147.0	9682	0.0	10
neos-905856		3123.6	276976	100.0	139		2492.4	270431	100.0	35	T	>3600.0	443477	0.0	11
neos-906865		282.6	78683	18.6	86		158.4	50413	29.0	38		125.4	34642	37.4	38
neos-912015		10.8	1951	-	14		18.0	3363	-	8		5.4	575	-	8
neos-933638		457.8	233	-	108	T	>3600.0	5884	-	1		635.4	385	-	1
neos-933815		840.6	110663	98.9	1578	T	>3600.0	1051053	97.5	388		594.0	174013	98.2	388

continued on next page

instance	no cut selection					violation					relative violation				
	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts
neos-933966		261.0	220	-	110		1285.8	803	-	1		1285.8	803	-	1
neos-934531		52.8	43	100.0	641		51.6	16	100.0	99		52.8	71	100.0	86
neos-935627		774.0	260	-	82		184.8	176	-	1		184.2	176	-	1
neos-935769		646.2	484	-	82		540.6	1527	-	1		540.0	1527	-	1
neos-936660		2647.8	1247	-	72	T	>3600.0	3381	-	0	T	>3600.0	3384	-	0
neos-937446		639.0	1239	-	63		440.4	1152	-	4		307.2	413	-	1
neos-937511		468.6	325	-	64		618.0	1475	-	1		617.4	1475	-	1
neos-942323	T	>3600.0	547440	0.0	382		532.8	181953	0.0	104	T	>3600.0	1112292	0.0	115
neos-948268		968.4	1064	-	143		753.6	5456	-	2		52.8	1092	-	3
neos-955215		621.0	171709	99.8	1357	T	>3600.0	2177602	98.9	181	F	>3600.0	3308676	92.4	13
neos-957270	T	>3600.0	251292	-	151	T	>3600.0	193730	-	28	T	>3600.0	194007	-	23
neos10		209.4	18	62.2	7521		530.4	19	59.5	554		235.2	18	58.6	348
neos11	T	>3600.0	23627	0.0	660		767.4	28173	0.0	25		918.6	35900	0.0	2
neos2	T	>3600.0	460853	16.5	259		2439.0	599414	12.6	81	T	>3600.0	442833	18.4	157
neos20		663.6	132233	4.9	2080		509.4	318627	2.4	179		181.8	123131	0.0	87
neos21		204.6	28793	16.4	975		93.0	20763	16.4	29		123.0	20737	37.3	28
neos22		501.0	32505	99.6	1783		307.8	102057	99.4	111		124.8	33917	99.6	136
neos6	T	>3600.0	90269	-	347		1846.8	46234	-	18	T	>3600.0	81288	-	8
neos648910	T	>3600.0	1653045	0.0	1691	T	>3600.0	3413518	0.0	118	T	>3600.0	3797204	0.0	128
neos671048		88.8	2575	25.2	153		480.6	19947	46.7	54	F	>3600.0	214	49.9	44
neos8		143.4	1	98.3	1324		139.8	1	99.1	8		139.8	1	99.1	8
neos897005		1726.8	136	100.0	142		955.2	156	100.0	19		480.6	176	100.0	14
nw04		489.0	628	54.8	32		245.4	47	89.5	17		157.2	79	87.7	8
pg5_34		1360.8	159271	98.8	1849		1456.2	141165	97.4	1381	T	>3600.0	221705	84.6	1403
prod1		1135.8	1819248	52.4	71	T	>3600.0	4530050	48.6	29		1222.8	2420624	46.6	18
qap10		328.8	230	19.3	128		140.4	223	19.3	7		187.8	234	19.3	10
qiu		468.6	36557	4.1	286		197.4	33946	4.1	33		216.6	41299	2.6	8
qnet1		262.2	62	40.3	431		208.8	217	40.6	137		127.8	101	34.7	134
qnet1_o		88.2	79	70.6	402		160.2	95	71.6	123		128.4	88	73.1	148
ran10x26		358.8	61605	49.0	351		201.6	86842	44.5	158		240.0	97294	47.1	161
ran12x21		1065.0	209395	54.2	461		769.8	366329	53.5	233		693.6	296381	51.9	272
ran13x13		250.8	89033	54.2	288		155.4	108271	54.8	139		135.6	92412	55.5	138
ran16x16		2650.2	524562	59.9	658		2405.4	1036416	55.5	286		3517.2	1188290	55.2	292
rmatr100-p10		711.6	26827	3.8	354		480.6	27397	0.2	1		494.4	28213	1.8	1
rmatr100-p5		258.0	4183	1.8	265		181.8	3530	0.1	1		193.2	3259	1.4	1
rout	T	>3600.0	660079	7.5	290		1288.8	817188	12.8	111		967.2	557344	13.8	105
set1ch	T	>3600.0	664595	92.3	2093	T	>3600.0	1727655	86.3	492	T	>3600.0	1833855	91.7	486
sp98ir		405.6	41992	1.8	473		295.2	25181	6.9	23		377.4	45403	4.4	17
swath1		117.6	5849	14.2	152		94.8	12798	13.9	38		86.4	15050	14.0	41
swath2	T	>3600.0	171938	23.7	168		974.4	147301	23.1	61		1522.8	229555	24.1	59
Test3		60.0	175	98.2	767		127.2	5987	91.6	157		51.0	255	98.0	153
vpm2		61.8	24796	80.1	708		61.8	40037	80.6	258		45.0	32801	79.1	261

Table 6: Detailed results (1). The columns headed “S” contain a “T” if an instance hit the time limit or an “F” if the solver failed. The columns headed “time (sec.)” and “# nodes” show the solution time and the number of branching nodes explored. The columns headed “gap cl. (%)” and “# cuts” contain the amount of integrality gap closed at the root node and the number of cutting planes which were added to the LP relaxation.

instance	distance					rotated distance					distance with bounds				
	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts
30_05_100	T	>3600.0	168764	100.0	1	T	>3600.0	168844	100.0	1	T	>3600.0	168666	100.0	1
30_95_100		611.4	5416	-	0		615.6	5416	-	0		298.8	2349	-	1
30_95_98		94.8	833	100.0	1		96.0	833	100.0	1		94.8	833	100.0	1
acc1		19.2	346	-	30		25.2	495	-	35		18.0	372	-	40
acc3		80.4	640	-	10	T	>3600.0	2818	-	10		80.4	640	-	10
aflow30a		181.8	48858	53.1	199		184.8	46701	53.7	191		189.0	55627	46.2	181
air04		196.2	2131	15.7	14		205.8	2131	15.7	14		243.0	2710	12.1	17
air05		175.8	16142	3.7	9		319.2	17156	4.1	15		278.4	13115	4.1	9
aligninq		237.6	4156	35.3	30		237.6	4156	35.3	30		315.0	3251	39.2	41
bienst1		193.2	51481	10.5	108		192.0	52837	14.1	103		197.4	54295	12.9	89
bienst2		874.8	221373	8.7	113		1196.4	357613	12.8	76		1147.8	285432	10.3	134
cap6000		40.8	4113	3.9	15		40.8	4113	3.9	15		40.2	4305	30.1	8
dcmulti		41.4	181	81.6	271		45.0	85	82.0	277		46.8	151	88.5	292
eilB101		291.6	52490	9.2	16		193.8	42127	5.9	16		312.6	50579	9.0	24
fiber		55.8	78	95.2	143		56.4	213	95.4	180		55.8	150	93.5	133
gesa2_o		66.6	12375	91.6	288		60.6	8872	93.8	320		27.6	4001	97.3	258
gesa3_o		57.6	189	82.0	182		21.0	221	82.8	187		54.6	197	85.5	270
harp2		1152.0	1157865	54.1	134		1101.0	1044540	57.7	160		1886.4	1472133	60.6	178
l152lav		61.2	14131	22.3	22		109.8	8124	22.3	33		82.2	7756	19.3	54
map18		1966.8	2752	26.3	341		1957.8	2752	26.3	341		1093.8	1081	29.4	371
map20		1157.4	1645	27.2	289		1126.2	1645	27.2	289		1577.4	2594	27.8	284
markshare_4_0		1377.0	1750919	0.0	5		1377.6	1750919	0.0	5		1376.4	1750919	0.0	5
mas76		400.8	805703	2.8	3		401.4	805703	2.8	3		400.2	805703	2.8	3
mitre		82.2	2217	90.8	1161		116.4	50	98.6	1273		100.2	35	93.7	1130
mod011		126.0	3057	71.1	770		127.8	3365	69.4	759		125.4	3291	70.5	789
neos-1053591		2196.6	3943366	0.0	4		2196.0	3943366	0.0	4		1909.2	3325015	0.0	4
neos-1061020		227.4	3780	4.5	12		235.2	3780	4.5	12		316.8	4674	4.5	14
neos-1122047		69.0	51	-	0		69.0	51	-	0		75.6	36	-	2
neos-1151496	T	>3600.0	804479	83.3	10		27.6	465	100.0	10		36.0	380	100.0	14
neos-1171448		28.8	152	-	1		28.8	152	-	1		29.4	152	-	1
neos-1171737	T	>3600.0	112389	-	3	T	>3600.0	112473	-	3	T	>3600.0	112333	-	3
neos-1200887		307.2	184967	0.0	1		306.6	184967	0.0	1		306.6	184967	0.0	1
neos-1211578		741.6	1213044	0.0	3		741.0	1213044	0.0	3		588.0	947955	0.0	4
neos-1228986		1648.8	2751269	0.0	9		1647.0	2751269	0.0	9		3327.0	4948360	0.0	9
neos-1367061		163.2	113	100.0	41		164.4	113	100.0	41		150.6	121	100.0	37
neos-1396125		1493.4	370698	26.4	33		2737.2	668932	21.8	29		2804.4	696609	23.4	27
neos-1420205		423.6	966767	45.0	29		423.6	966767	45.0	29		423.6	966767	45.0	29
neos-1440447		1377.0	1509190	0.0	5		1377.0	1509190	0.0	5		482.4	646810	0.0	5
neos-1460265		10.8	4536	-	180	T	>3600.0	1524204	-	184	T	>3600.0	1610857	-	189
neos-1489999		453.0	197408	11.9	47		441.6	270384	6.0	23		452.4	197408	11.9	47
neos-1582420		1164.0	63005	41.6	28		1161.0	63005	41.6	28		1248.0	88823	41.6	26
neos-1597104		156.0	1486	-	6		52.2	357	-	5		155.4	1486	-	6
neos-430149		1888.8	2060747	0.0	59		1887.0	2060747	0.0	59		1885.2	2060747	0.0	59
neos-480878	T	>3600.0	259776	64.1	7	T	>3600.0	260048	64.1	7	T	>3600.0	260097	64.1	7
neos-503737		211.8	7781	-	15		307.2	9380	-	23		198.0	6777	-	18
neos-504674		3088.2	2087820	34.0	187		3090.6	2087820	34.0	187		3089.4	2087820	34.0	187
neos-504815		196.8	98224	26.8	245		195.6	98224	26.8	245		406.8	258369	27.1	229
neos-512201		489.0	258016	41.7	250		489.6	258016	41.7	250		489.0	258016	41.7	250
neos-522351	T	>3600.0	296580	83.2	358	T	>3600.0	385462	85.2	338	T	>3600.0	144271	86.7	373
neos-525149		363.6	58673	0.0	87		363.6	58673	0.0	87		331.2	58524	0.0	112

continued on next page

	distance					rotated distance					distance with bounds				
instance	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts
neos-530627		0.2	1	100.0	5		0.2	1	100.0	5		0.2	1	100.0	5
neos-538867		288.0	260368	0.0	55		366.6	397090	0.0	25		297.0	264979	0.0	55
neos-538916		479.4	487577	0.0	20		505.8	506881	0.0	22		443.4	448732	0.0	20
neos-551991		1209.0	149391	31.4	64		1208.4	149391	31.4	64		1210.2	149391	31.4	64
neos-555694		25.8	5664	0.0	22	T	>3600.0	1474548	0.0	24		24.0	4901	0.0	21
neos-555771		72.6	22308	0.0	24		70.8	21527	0.0	24		63.6	18194	0.0	34
neos-570431		77.4	17507	43.8	53		78.0	17507	43.8	53		72.6	14800	25.1	52
neos-583731		1.8	229	-	135		70.8	45495	-	130		8.4	4681	-	138
neos-584851		901.8	939790	0.0	32		375.0	398845	0.0	43		898.8	939790	0.0	32
neos-585192	F	>3600.0	618223	60.8	138		903.0	186422	60.0	62	F	>3600.0	622773	60.8	138
neos-595905	T	>3600.0	719892	65.4	332	T	>3600.0	521234	40.3	369		1450.8	264024	31.9	368
neos-598183		667.2	106548	93.9	478		377.4	51983	93.6	533		1767.0	413669	92.6	526
neos-603073	T	>3600.0	511419	7.8	439		487.2	45498	6.5	446	T	>3600.0	493610	6.4	453
neos-611838		218.4	7057	76.9	562		192.0	5461	78.4	589		297.6	8983	76.1	547
neos-612125		127.8	1275	77.5	557		135.0	1333	77.1	562		136.8	1725	77.0	556
neos-612143		161.4	1871	78.3	522		147.0	2807	79.1	532		161.4	2475	78.5	538
neos-612162		182.4	4033	78.6	520		131.4	2471	78.4	524		153.6	4457	78.8	530
neos-686190		807.0	87992	6.8	51		999.6	89173	7.5	55		1079.4	100041	7.3	51
neos-709469		7.8	11226	-	34		18.6	24838	-	32		7.8	11226	-	34
neos-775946	T	>3600.0	717014	56.6	71	T	>3600.0	638012	70.4	138	T	>3600.0	578895	66.5	178
neos-777800		113.4	10229	-	0		113.4	10229	-	0		112.2	3936	-	2
neos-785914		50.4	4190	-	76		50.4	4190	-	76		5.4	128	-	77
neos-791021	T	>3600.0	851996	7.7	94	T	>3600.0	663588	7.7	100	T	>3600.0	717047	7.7	97
neos-796608	T	>3600.0	3232882	0.0	15	T	>3600.0	3437371	0.0	14	T	>3600.0	3323004	0.0	8
neos-803219		85.8	53459	27.3	69		92.4	53789	28.3	76		85.2	52723	27.3	70
neos-803220		73.8	80214	32.5	37		74.4	80214	32.5	37		73.8	80214	32.5	37
neos-806323		45.0	19860	29.5	83		47.4	19154	29.5	96		45.0	19860	29.5	83
neos-807639		67.2	22825	35.1	133		78.6	24883	23.1	134		85.2	34393	32.9	129
neos-807705		40.2	7445	28.0	149		36.6	7381	19.4	128		39.0	9913	22.2	142
neos-808072		65.4	442	98.6	29		91.2	553	98.6	32		89.4	353	98.6	37
neos-810286		93.0	1660	-	4		54.6	816	-	5		93.6	1660	-	4
neos-820879		964.8	45231	65.6	78		2409.6	81240	63.5	97		945.6	33417	69.7	79
neos-824661	T	>3600.0	3614	-	128	T	>3600.0	2330	-	128	T	>3600.0	3595	-	128
neos-824695		77.4	480	-	68		274.2	1280	-	68		78.0	480	-	68
neos-825075		62.4	26786	0.0	60		26.4	14433	0.0	21		41.4	11917	0.0	105
neos-826224	T	>3600.0	10894	-	93	T	>3600.0	13604	-	94	T	>3600.0	10135	-	96
neos-826250		114.6	560	-	106	T	>3600.0	22554	-	108		46.8	204	-	106
neos-826812		74.4	269	-	48		110.4	445	-	48		74.4	269	-	48
neos-827175		60.0	47	-	34		99.6	77	-	34		60.0	47	-	34
neos-829552		1484.4	8479	3.6	9		1285.8	8425	4.0	20		1853.4	8521	3.6	19
neos-839859		153.0	13490	9.1	22		140.4	12200	13.3	21		108.0	16846	4.5	14
neos-860300		62.4	600	38.3	2		62.4	600	38.3	2		62.4	600	38.3	2
neos-863472		1236.6	2093339	0.0	10		1235.4	2093339	0.0	10		1236.6	2093339	0.0	10
neos-881765		3.6	301	-	4		3.6	301	-	4		3.6	301	-	4
neos-885086		160.8	446	-	4		160.8	446	-	4		96.6	292	-	4
neos-892255		186.6	9970	0.0	3		156.0	11406	0.0	4		152.4	10990	0.0	3
neos-905856	T	>3600.0	210389	0.0	19	T	>3600.0	326313	0.0	12	T	>3600.0	222090	0.0	19
neos-906865		184.2	51367	27.1	24		185.4	68453	20.2	25		184.2	51367	27.1	24
neos-912015		15.6	3681	-	8		4.2	337	-	9		15.6	2803	-	8
neos-933638		153.0	114	-	1		153.0	114	-	1		153.0	114	-	1
neos-933815		3363.0	1431774	98.5	326		2402.4	1145131	98.6	342		879.0	390871	98.7	322

continued on next page

	distance					rotated distance					distance with bounds				
instance	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts
neos-933966		1285.8	803	-	1		1285.2	803	-	1		1285.8	803	-	1
neos-934531		51.0	26	100.0	71		51.6	26	100.0	65		51.6	8	100.0	250
neos-935627	T	>3600.0	9653	-	1		184.8	176	-	1	T	>3600.0	9655	-	1
neos-935769		540.6	1527	-	1		540.6	1527	-	1		540.0	1527	-	1
neos-936660	T	>3600.0	3375	-	0	T	>3600.0	3375	-	0	T	>3600.0	11448	-	10
neos-937446		307.8	413	-	1		341.4	572	-	3		307.8	413	-	1
neos-937511		618.0	1475	-	1		618.0	1475	-	1		617.4	1475	-	1
neos-942323	T	>3600.0	1223237	0.0	113	T	>3600.0	1384855	0.0	82		1536.6	389673	0.0	119
neos-948268		132.0	3056	-	1		177.6	644	-	1		174.6	644	-	1
neos-955215		348.0	364407	99.8	185		259.2	206289	99.6	190		133.2	88629	99.8	203
neos-957270		10.8	544	-	18		8.4	265	-	23		10.8	544	-	18
neos10		351.6	16	61.3	196		417.6	16	62.2	440		237.0	21	59.5	198
neos11		1555.2	55449	0.0	2		1555.8	55449	0.0	2		1554.6	55449	0.0	2
neos2		1718.4	391323	14.0	49	T	>3600.0	609795	19.2	95		1258.2	271860	12.9	39
neos20		441.6	280066	0.0	110		972.6	613248	0.0	127		444.6	276341	0.0	121
neos21		100.8	21052	37.3	24		100.2	21052	37.3	24		100.2	21052	37.3	24
neos22		39.0	3479	99.9	195		39.0	3479	99.9	195		154.8	49989	99.5	81
neos6	T	>3600.0	105917	-	8		240.0	4752	-	8		3108.0	76425	-	10
neos648910	T	>3600.0	3666028	0.0	117	T	>3600.0	3666718	0.0	117	T	>3600.0	3511302	0.0	123
neos671048		103.8	4118	24.9	5		104.4	4118	24.9	5		103.8	4118	24.9	5
neos8		139.2	1	100.0	1		139.8	1	99.6	2		139.2	1	99.6	2
neos897005		944.4	185	100.0	5		1236.6	244	100.0	5		1055.4	192	100.0	6
nw04		81.6	892	1.1	4		466.2	364	42.9	14		367.2	346	40.1	14
pg5_34		1095.6	139585	98.8	1270		1095.0	139585	98.8	1270		891.6	128303	98.7	1299
prod1	T	>3600.0	4460357	45.9	20	T	>3600.0	4461708	45.9	20	T	>3600.0	4461764	45.9	20
qap10		207.6	232	19.3	43		208.8	232	19.3	43		216.0	220	19.3	43
qiu		118.8	27941	0.0	1		118.2	27941	0.0	1		118.2	27941	0.0	1
qnet1		285.0	138	51.3	81		113.4	140	45.2	66		211.8	69	46.3	88
qnet1_o		109.2	53	68.6	93		104.4	75	72.8	100		181.2	116	68.4	123
ran10x26		202.2	73974	52.5	135		168.6	59063	53.1	131		160.8	51082	51.8	130
ran12x21		822.0	345039	53.4	249		457.8	173220	56.1	257		864.0	398333	52.2	245
ran13x13		119.4	97027	57.3	122		101.4	77731	59.2	118		94.8	68612	57.3	130
ran16x16	T	>3600.0	1337270	52.0	276		2362.8	1062596	56.5	268	T	>3600.0	1414626	55.5	275
rmatr100-p10		494.4	28213	1.8	1		494.4	28213	1.8	1		494.4	28213	1.8	1
rmatr100-p5		192.6	3259	1.4	1		193.2	3259	1.4	1		192.6	3259	1.4	1
rout		790.2	486954	13.0	90		930.6	461991	17.2	110		1335.0	858224	13.3	94
set1ch	T	>3600.0	1698251	90.6	511	T	>3600.0	2025712	90.6	519	T	>3600.0	1997430	90.5	518
sp98ir		307.2	44401	0.2	10		307.2	44401	0.2	10		342.6	42851	4.2	14
swath1		155.4	33055	13.6	23		134.4	28930	13.9	26		47.4	2520	14.4	46
swath2		1729.2	425495	22.9	32		1299.6	303787	22.6	30		1454.4	372726	22.9	30
Test3		52.2	41	99.8	140		46.8	35	99.8	148		79.8	2233	92.6	141
vpm2		39.0	35281	81.2	190		32.4	18942	81.5	209		35.4	33718	80.1	179

Table 7: Detailed results (2). The columns headed "S" contain a "T" if an instance hit the time limit or an "F" if the solver failed. The columns headed "time (sec.)" and "# nodes" show the solution time and the number of branching nodes explored. The columns headed "gap cl. (%)" and "# cuts" contain the amount of integrality gap closed at the root node and the number of cutting planes which were added to the LP relaxation.

instance	rotated distance with bounds					objective parallelism					expected improvement				
	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts
30_05_100	T	>3600.0	169056	100.0	1	T	>3600.0	86260	100.0	1	T	>3600.0	166669	100.0	1
30_95_100		299.4	2349	-	1		394.8	2632	-	1		616.2	5416	-	0
30_95_98		95.4	833	100.0	1	T	>3600.0	121401	100.0	1		95.4	833	100.0	1
acc1		28.2	570	-	33		16.8	545	-	1		16.8	545	-	1
acc3	T	>3600.0	9470	-	10	T	>3600.0	3516	-	1	T	>3600.0	3515	-	1
aflow30a		247.2	79472	48.5	184		222.6	68194	42.8	202		276.0	83750	43.4	240
air04		249.6	2710	12.1	17		135.6	3312	7.1	4		81.0	3387	5.7	3
air05		463.2	18207	4.1	16		183.6	16971	3.5	4		148.8	16653	3.5	4
aligninq		313.2	3251	39.2	41		70.8	2571	15.8	59		70.8	2571	15.8	59
bienst1		191.4	52837	14.1	103		164.4	49888	16.4	36		111.0	55645	1.9	11
bienst2		999.6	345279	12.8	77		786.0	249331	11.2	59		647.4	244301	11.8	66
cap6000		40.2	4305	30.1	8		45.6	4193	28.9	17		89.4	4757	40.7	10
dcmulti		45.6	103	79.0	258		46.2	400	87.8	249		42.6	93	86.8	307
eilB101		199.2	42932	5.9	16		178.8	51122	0.8	4		192.0	51978	0.9	5
fiber		73.2	86	95.0	150		59.4	607	75.1	170		61.2	185	91.9	145
gesa2_o		44.4	6989	95.3	309		619.2	292712	58.6	227	T	>3600.0	1737283	73.5	340
gesa3_o		54.0	210	82.2	174		57.0	415	73.0	196		59.4	198	81.5	216
harp2	F	>3600.0	2763993	59.0	175		1739.4	1971834	27.4	73		1562.4	1385093	51.4	238
l152lav		82.2	8414	28.4	35		120.0	8548	17.7	14		54.0	8443	16.2	44
map18		1094.4	1081	29.4	371	T	>3600.0	6205	12.4	62		2964.6	6131	18.6	77
map20		1579.2	2594	27.8	284		1782.0	4095	10.7	53		1866.6	4192	18.8	69
markshare_4_0		1378.2	1750919	0.0	5		1819.2	2279786	0.0	1		1726.8	2268060	0.0	1
mas76		400.8	805703	2.8	3		423.0	896595	4.8	4		397.8	748631	2.6	1
mitre		90.0	113	94.0	1225	T	>3600.0	531342	63.3	966		319.2	9698	91.8	1465
mod011		129.0	3185	67.9	736		159.0	4113	69.1	776		113.4	2603	71.7	795
neos-1053591		2074.8	3698146	0.0	4		3483.6	6068556	0.0	1		2061.0	3666884	0.0	3
neos-1061020		324.6	4674	4.5	14		283.2	4534	6.4	24		296.4	3813	6.0	25
neos-1122047		74.4	36	-	2		69.6	51	-	0		72.0	8	-	1
neos-1151496		27.6	465	100.0	10	T	>3600.0	351113	33.3	3	T	>3600.0	351021	33.3	3
neos-1171448		29.4	152	-	1		42.0	228	-	14		29.4	152	-	1
neos-1171737	T	>3600.0	112487	-	3	T	>3600.0	75978	-	7		907.2	20073	-	4
neos-1200887		307.2	184967	0.0	1		592.2	348943	0.0	1		307.8	184967	0.0	1
neos-1211578		588.6	947955	0.0	4		543.0	907310	0.0	1		775.2	1206815	0.0	6
neos-1228986		3328.2	4948360	0.0	9		3333.0	5163603	0.0	1		2412.6	3764512	0.0	3
neos-1367061		151.8	121	100.0	37		133.2	173	100.0	15		140.4	129	100.0	35
neos-1396125		2337.6	594080	23.4	34	F	>3600.0	153069	0.0	3		1974.0	404074	12.1	45
neos-1420205		424.2	966767	45.0	29		152.4	279379	45.0	48		1.8	3534	45.0	34
neos-1440447		483.0	646810	0.0	5		616.2	817773	0.0	1		1110.6	1263418	0.0	6
neos-1460265	T	>3600.0	1613159	-	183		402.6	249317	-	185		4.2	1677	-	149
neos-1489999		441.0	270384	6.0	23		525.6	253775	7.1	36		609.0	270996	13.1	42
neos-1582420		1245.6	88823	41.6	26		876.6	89954	12.4	9		1134.6	64341	41.6	33
neos-1597104		99.6	905	-	5		29.4	147	-	4		173.4	1610	-	5
neos-430149		1886.4	2060747	0.0	59		1327.2	1639235	0.0	3		1872.6	2340691	0.0	3
neos-480878	T	>3600.0	260166	64.1	7	T	>3600.0	262519	64.1	7	T	>3600.0	270767	64.1	2
neos-503737		51.6	1386	-	22		82.2	3041	-	1		334.8	12907	-	17
neos-504674		3090.6	2087820	34.0	187	T	>3600.0	1628843	31.7	177		1404.6	839978	32.7	166
neos-504815		406.8	258369	27.1	229		416.4	287388	26.8	153		483.0	326286	26.2	129
neos-512201		489.0	258016	41.7	250		460.8	236554	33.0	167		955.8	622690	41.3	152
neos-522351	T	>3600.0	127553	84.7	364	T	>3600.0	239213	36.6	46	T	>3600.0	1003810	28.6	28
neos-525149		331.2	58524	0.0	112		190.2	61733	0.0	1		154.2	46290	0.0	9

continued on next page

	rotated distance with bounds					objective parallelism					expected improvement				
instance	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts
neos-530627		0.2	1	100.0	5		1266.6	2733988	47.9	5		0.2	7	100.0	4
neos-538867		366.6	397090	0.0	25		198.0	197334	0.0	28		382.8	394312	0.0	28
neos-538916		505.2	506881	0.0	22		499.2	506507	0.0	15		516.0	442025	0.0	69
neos-551991		1207.8	149391	31.4	64		1524.0	134655	14.3	24		1087.2	117060	31.4	62
neos-555694		24.0	4901	0.0	21		13.2	281	0.0	7		12.0	770	0.0	22
neos-555771		64.8	18960	0.0	26	T	>3600.0	1536784	0.0	5		54.0	16011	0.0	17
neos-570431		72.6	14800	25.1	52		96.0	25493	25.1	16		73.2	10769	25.1	27
neos-583731	T	>3600.0	2195418	-	149		90.0	49543	-	233		15.0	8010	-	148
neos-584851		373.8	398845	0.0	43		196.8	254613	0.0	4		519.6	606327	0.0	7
neos-585192		2682.0	487192	60.0	54		1695.0	322844	59.6	10		764.4	163453	59.7	7
neos-595905		1600.2	323051	40.8	340	T	>3600.0	723163	73.2	312		3467.4	737655	66.8	373
neos-598183		703.8	133191	91.4	448		1188.6	120333	86.2	413		1073.4	212422	37.1	460
neos-603073		1801.2	152617	6.6	457	T	>3600.0	586156	6.3	373		269.4	15192	7.3	403
neos-611838		300.6	9867	75.4	568		667.2	26729	64.0	524		298.8	9529	73.4	583
neos-612125		150.0	1321	76.3	563		205.2	2799	65.9	510		142.2	1219	75.1	553
neos-612143		163.2	1931	79.1	551		229.2	6539	64.7	471		140.4	3191	76.4	556
neos-612162		139.8	2401	78.7	523		510.0	20343	67.4	518		306.6	9411	75.0	580
neos-686190		999.0	92328	7.5	53		669.6	93673	3.4	9		558.6	93217	7.5	9
neos-709469		4.8	6473	-	33		1.2	414	-	17		25.2	37665	-	14
neos-775946	T	>3600.0	628877	62.5	142	T	>3600.0	543101	37.4	73	T	>3600.0	554163	18.5	69
neos-777800		112.2	3936	-	2		28.8	1580	-	1		114.0	10229	-	0
neos-785914		5.4	128	-	77		10.2	514	-	19		114.0	8331	-	39
neos-791021	T	>3600.0	595602	7.7	102		229.8	565	76.9	229	T	>3600.0	655706	7.7	53
neos-796608	T	>3600.0	3446541	0.0	14	T	>3600.0	3065544	0.0	3	T	>3600.0	5166629	0.0	3
neos-803219		92.4	53789	28.3	76		63.0	54604	25.3	44		102.6	53413	29.5	87
neos-803220		74.4	80214	32.5	37		70.2	80078	30.8	31		82.2	76835	32.6	50
neos-806323		55.8	13970	30.5	134		52.8	15716	24.0	150		48.6	12377	24.7	127
neos-807639		64.8	24138	22.7	127		61.2	23714	21.6	75		58.2	26313	32.6	91
neos-807705		28.8	9294	16.1	105		34.8	12616	8.4	62		35.4	8450	17.5	112
neos-808072		102.0	593	98.6	32	T	>3600.0	91566	43.1	3		57.6	437	98.6	19
neos-810286		54.6	816	-	5		100.2	1613	-	0		99.6	1613	-	0
neos-820879		1666.2	45415	61.9	117		1755.0	49583	63.6	122		2088.0	82120	54.2	155
neos-824661		2862.6	1698	-	128		844.2	4046	-	157	T	>3600.0	3559	-	128
neos-824695		448.8	1366	-	68		1543.2	23980	-	117		78.0	480	-	68
neos-825075		51.0	28299	0.0	23		34.8	20327	0.0	5		58.2	35445	0.0	5
neos-826224	T	>3600.0	39992	-	95	T	>3600.0	17748	-	110	T	>3600.0	25745	-	94
neos-826250		174.6	1280	-	103		33.0	337	-	115		104.4	544	-	105
neos-826812		117.6	396	-	48		40.2	177	-	67		75.6	269	-	48
neos-827175		74.4	40	-	34		60.0	47	-	34		60.0	47	-	34
neos-829552		1660.2	7541	5.5	24		1026.6	9589	0.5	1		1029.0	9331	1.0	2
neos-839859		108.6	16846	4.5	14		90.6	17245	5.0	5		158.4	13973	9.0	27
neos-860300		62.4	600	38.3	2		87.0	602	38.3	3		62.4	594	38.3	0
neos-863472		1236.0	2093339	0.0	10		1169.4	2074536	0.0	0		1405.2	2329050	0.0	2
neos-881765		3.6	301	-	4		3.6	301	-	4		4.2	301	-	4
neos-885086		96.6	292	-	4		150.6	556	-	7		202.8	908	-	6
neos-892255		156.0	11406	0.0	4		172.8	10257	0.0	80		191.4	9859	0.0	30
neos-905856	T	>3600.0	439137	0.0	12	T	>3600.0	424894	0.0	5	T	>3600.0	271285	0.0	7
neos-906865		192.0	71311	20.2	25		465.0	232463	0.0	1		192.6	62971	25.3	35
neos-912015		10.2	1704	-	9		61.2	14329	-	2		61.2	14329	-	2
neos-933638		153.6	114	-	1		405.6	248	-	1		259.8	165	-	1
neos-933815		2911.2	1396420	98.8	308	T	>3600.0	1693673	78.4	106	T	>3600.0	595100	88.3	313

continued on next page

	rotated distance with bounds					objective parallelism					expected improvement				
instance	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts
neos-933966		1285.2	803	-	1		611.4	1936	-	2		1291.8	803	-	1
neos-934531		52.2	27	100.0	228		52.8	12	100.0	242		52.2	23	100.0	4
neos-935627		185.4	176	-	1	T	>3600.0	7523	-	1		123.6	88	-	1
neos-935769		540.6	1527	-	1		414.6	993	-	1		393.0	960	-	1
neos-936660	T	>3600.0	11450	-	10	T	>3600.0	3348	-	0		3519.0	6798	-	2
neos-937446		819.0	2418	-	3	T	>3600.0	3571	-	0		468.0	764	-	1
neos-937511		617.4	1475	-	1		368.4	737	-	1		1930.8	7057	-	1
neos-942323	T	>3600.0	1281594	0.0	82	T	>3600.0	1119281	0.0	109	T	>3600.0	1998909	0.0	7
neos-948268		178.2	644	-	1		958.2	6196	-	35		958.2	6196	-	35
neos-955215		134.4	88629	99.8	203	F	>3600.0	2621248	92.4	61	F	>3600.0	2771399	92.4	10
neos-957270	T	>3600.0	277172	-	25	T	>3600.0	258794	-	38	T	>3600.0	259767	-	38
neos10		286.8	21	63.1	219		288.6	17	55.9	225		139.2	30	45.1	67
neos11		1555.2	55449	0.0	2		1134.0	40644	0.0	1		1562.4	55449	0.0	2
neos2		3315.6	449194	20.1	105		429.0	95454	9.4	74		241.2	80882	10.5	38
neos20		354.0	211642	2.4	178		1314.6	943137	0.0	5		1108.2	885890	0.0	5
neos21		100.2	21052	37.3	24		115.2	27228	16.4	14		88.8	23607	16.4	11
neos22		154.2	49989	99.5	81		460.8	178241	99.3	121		213.0	34257	99.6	183
neos6	T	>3600.0	101669	-	10	T	>3600.0	61648	-	4		94.2	436	-	2
neos648910	T	>3600.0	3512433	0.0	123	T	>3600.0	2864248	0.0	426		261.0	270904	0.0	416
neos671048		103.8	4118	24.9	5		70.2	2032	49.9	20		121.2	3570	25.2	29
neos8		139.8	1	100.0	3		141.6	1	100.0	47		141.0	1	100.0	7
neos897005		719.4	124	100.0	6		301.8	105	100.0	9		650.4	156	100.0	1
nw04		324.6	1187	40.1	8		646.2	777	26.5	6		294.0	194	79.1	9
pg5_34		892.2	128303	98.7	1299	T	>3600.0	235485	31.1	440	T	>3600.0	129429	51.1	788
prod1	T	>3600.0	4461023	45.9	20		772.8	1210679	52.2	43		1204.2	2029987	49.5	30
qap10		217.8	220	19.3	43		151.2	222	19.3	6		148.8	225	19.3	6
qiu		118.2	27941	0.0	1		142.8	34279	0.0	1		118.8	27941	0.0	1
qnet1		222.6	127	50.6	81		177.0	63	42.5	123		203.4	77	40.1	112
qnet1_o		135.6	57	64.2	118		139.8	76	62.3	151		171.6	408	64.6	134
ran10x26		167.4	54992	51.4	130		231.0	96014	48.3	171		251.4	92730	49.9	166
ran12x21		621.0	267296	55.4	258		657.6	250855	51.3	276		565.2	243742	57.2	258
ran13x13		102.0	83208	58.9	118		200.4	189079	44.5	116		111.0	78844	57.7	129
ran16x16		3221.4	1291029	55.6	266		3021.0	1360885	53.5	289		2642.4	876080	57.7	283
rmatr100-p10		494.4	28213	1.8	1		392.4	26079	0.0	0		391.8	26079	0.0	0
rmatr100-p5		193.2	3259	1.4	1		138.0	2729	0.0	0		138.0	2729	0.0	0
rout		1005.6	524179	18.8	117		524.4	999916	0.0	2		715.8	1274557	0.0	2
set1ch	T	>3600.0	1422201	88.9	523	T	>3600.0	1645291	87.7	530	T	>3600.0	1837191	90.6	491
sp98ir		343.2	42851	4.2	14		227.4	30763	0.2	9		246.0	28472	1.7	43
swath1		46.2	8956	14.2	21		122.4	22721	13.7	18		108.0	18725	14.1	26
swath2		765.6	110936	23.5	36		1536.6	413050	12.0	10		1531.8	217863	12.1	20
Test3		53.4	35	99.8	143		1028.4	89567	22.8	1		465.0	38087	33.1	64
vpm2		33.0	24319	82.7	215		74.4	59250	78.0	294		73.8	66270	77.9	197

Table 8: Detailed results (3). The columns headed “S” contain a “T” if an instance hit the time limit or an “F” if the solver failed. The columns headed “time (sec.)” and “# nodes” show the solution time and the number of branching nodes explored. The columns headed “gap cl. (%)” and “# cuts” contain the amount of integrality gap closed at the root node and the number of cutting planes which were added to the LP relaxation.

instance	support					integral support					adjusted distance				
	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts
30_05_100	T	>3600.0	104159	100.0	1		1073.4	21561	100.0	1	T	>3600.0	166901	100.0	1
30_95_100		451.8	3233	-	1		300.6	2349	-	1		614.4	5416	-	0
30_95_98		3586.2	176912	100.0	1		95.4	833	100.0	1		95.4	833	100.0	1
acc1		47.4	907	-	36		31.8	597	-	47		26.4	524	-	31
acc3		221.4	2568	-	11		140.4	1553	-	11		80.4	640	-	10
aflow30a		206.4	35927	52.5	372		286.2	50603	48.0	313		282.0	78032	48.5	215
air04		154.2	1765	14.2	24		291.6	1410	17.4	36		244.8	2836	14.9	31
air05		287.4	15467	7.7	11		320.4	18468	3.5	17		227.4	15243	8.5	7
aligninq		162.0	31330	19.4	41		70.2	2571	15.8	59		138.0	2258	33.2	19
bienst1		266.4	54255	12.6	80		192.0	50449	20.7	95		150.0	33917	13.2	123
bienst2		1871.4	369343	8.6	189		649.8	287671	12.9	83		1267.8	368179	7.2	154
cap6000		85.2	4729	40.1	12		63.0	4065	43.2	18		58.2	4925	35.1	10
dcmulti		46.2	432	79.9	346		50.4	128	81.8	333		47.4	87	87.4	309
eilB101		287.4	48904	8.5	23		370.2	53489	10.1	32		199.8	51356	6.4	10
fiber		60.0	248	91.7	183		64.2	230	93.7	251		59.4	287	91.5	157
gesa2_o		58.8	4016	94.0	473		52.8	10211	92.7	433		60.0	7575	95.6	286
gesa3_o		34.8	189	80.2	336		59.4	258	78.7	335		35.4	224	81.7	170
harp2		2425.2	793707	44.0	453		2430.6	1019920	62.1	578	T	>3600.0	2079843	59.1	188
l152lav		45.6	9653	14.7	26		75.6	10719	20.8	49		68.4	10240	16.2	63
map18		3412.8	5099	24.6	633		2831.4	6769	20.5	345		2683.2	3110	28.1	352
map20		1512.6	3246	24.6	536		1640.4	4361	16.5	253		673.8	1177	24.9	235
markshare_4_0		1873.2	2342467	0.0	5		1710.0	2142002	0.0	4		885.0	1610042	0.0	5
mas76		582.6	1377011	5.2	5		421.8	896595	4.8	4		403.2	805703	2.8	3
mitre		927.0	32279	81.7	2985		269.4	308	95.7	3010		99.0	40	99.8	1217
mod011		145.8	2775	74.1	1196		120.6	4717	67.0	921		87.6	3523	69.3	658
neos-1053591		2862.6	4538266	0.0	39		3296.4	5398248	0.0	38		2197.2	3903474	0.0	4
neos-1061020		364.8	5897	6.0	36		328.2	4765	4.3	37		277.2	4367	5.0	21
neos-1122047		75.0	36	-	2		75.6	36	-	2		69.6	51	-	0
neos-1151496		26.4	420	100.0	15		27.0	493	100.0	15		40.8	529	100.0	18
neos-1171448		48.6	204	-	92		55.2	284	-	92		28.8	152	-	1
neos-1171737	T	>3600.0	95751	-	58	T	>3600.0	85461	-	53		144.0	2637	-	4
neos-1200887		308.4	184967	0.0	1		307.8	184967	0.0	1		309.0	184967	0.0	1
neos-1211578		807.0	1213474	0.0	14		1799.4	2701387	0.0	10		787.2	1268927	0.0	5
neos-1228986		3517.2	4315220	0.0	28		2328.6	3125930	0.0	14		2786.4	4071365	0.0	10
neos-1367061		129.0	129	100.0	85		172.2	89	100.0	87		126.6	129	100.0	47
neos-1396125		2453.4	484104	4.9	118		2368.2	440432	8.1	66		1740.0	426378	19.5	50
neos-1420205		1749.0	2471259	45.0	66		384.6	683415	45.0	56		1.2	2498	37.5	23
neos-1440447		1171.2	1303754	0.0	11		790.8	980967	0.0	9		570.0	749719	0.0	5
neos-1460265	T	>3600.0	1293864	-	240	T	>3600.0	1595077	-	248	T	>3600.0	1674118	-	184
neos-1489999		544.2	232459	10.7	43		610.2	253644	10.7	64		505.2	249940	7.1	33
neos-1582420		2354.4	183489	41.6	60		3376.8	310729	12.4	56		2327.4	194941	41.6	41
neos-1597104		204.0	2089	-	7		76.2	637	-	7		22.8	80	-	6
neos-430149		2870.4	2942043	0.0	83		1764.0	1862269	0.0	57		2846.4	3120087	0.0	49
neos-480878	T	>3600.0	270013	64.1	21		2040.0	155525	64.1	12		3464.4	252179	64.1	8
neos-503737		128.4	3811	-	56		115.8	2491	-	120		48.6	1369	-	23
neos-504674		1843.8	853595	44.8	472		1381.2	733931	43.7	331		1323.0	831830	33.3	222
neos-504815		185.4	63002	38.3	434		209.4	110056	32.0	324		220.2	128549	34.9	277
neos-512201		529.8	208234	45.7	411		444.0	188399	44.0	344		477.0	257369	37.8	233
neos-522351		507.6	78419	98.9	346	T	>3600.0	232281	95.1	338	T	>3600.0	508181	98.3	241
neos-525149		241.2	53133	0.0	91		259.8	72883	0.0	134		360.0	63683	0.0	95

continued on next page

	support					integral support					adjusted distance				
instance	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts
neos-530627		1573.2	3369401	44.6	2		1589.4	3415389	46.2	6		0.2	1	100.0	3
neos-538867		399.6	261449	0.0	110		496.8	421503	0.0	98		372.6	376334	0.0	71
neos-538916		720.0	609564	0.0	63		553.2	526869	0.0	53		414.0	389657	0.0	39
neos-551991		1299.0	104614	31.4	78		1014.0	106899	31.4	153		901.2	96596	31.4	52
neos-555694	T	>3600.0	788930	0.0	134	T	>3600.0	667108	0.0	149		24.0	4590	0.0	23
neos-555771		85.8	22721	0.0	86		91.2	22050	0.0	125		325.8	94341	0.0	57
neos-570431		69.6	20472	25.1	26		69.6	21897	25.1	33		107.4	29416	25.1	15
neos-583731		68.4	27677	-	373		13.8	7326	-	279		899.4	572973	-	159
neos-584851		397.2	252459	0.0	150		226.2	225134	0.0	51		66.6	75223	0.0	32
neos-585192	F	>3600.0	613740	60.9	193	F	>3600.0	1000	60.2	154		546.6	99260	60.7	130
neos-595905		463.8	37244	69.5	520		2283.0	225263	41.4	477	T	>3600.0	546657	39.8	349
neos-598183		2842.8	266804	90.1	626		937.8	126366	38.7	482		2888.4	407793	36.5	346
neos-603073		2271.0	124973	6.7	582		1669.8	128935	7.2	588		749.4	57907	6.5	455
neos-611838		519.6	8717	74.4	733		113.4	6411	73.4	610		378.6	13303	74.3	593
neos-612125		178.2	1299	74.8	742		106.8	1963	71.6	624		161.4	1953	77.2	635
neos-612143		222.6	2341	77.9	711		85.2	4071	72.1	608		151.2	2385	77.9	610
neos-612162		399.6	5191	73.8	701		90.0	4295	74.3	589		322.8	6819	77.4	636
neos-686190		1041.0	101018	7.3	57		1069.8	94688	7.5	70		1082.4	95215	7.8	54
neos-709469	T	>3600.0	2534001	-	124		17.4	21462	-	109		5.4	6125	-	30
neos-775946	T	>3600.0	589588	50.8	211	T	>3600.0	588496	61.4	223	T	>3600.0	622393	74.0	141
neos-777800		126.6	10500	-	2		112.8	3936	-	2		114.6	10229	-	0
neos-785914		130.2	9282	-	76		45.6	3797	-	101		113.4	7275	-	83
neos-791021		115.8	964	100.0	595		63.6	484	100.0	637		115.8	3473	76.9	159
neos-796608		0.2	1	100.0	15		0.2	1	100.0	15	T	>3600.0	3511334	0.0	10
neos-803219		100.8	51131	29.5	95		108.0	52810	29.5	96		96.0	54987	28.1	75
neos-803220		95.4	76702	33.8	73		91.2	81723	34.0	61		67.2	78794	30.4	32
neos-806323		46.8	13401	30.7	149		54.0	20605	23.8	101		49.2	12066	34.9	124
neos-807639		104.4	27415	23.9	196		73.2	23668	25.7	120		76.8	30166	24.6	112
neos-807705		42.6	8191	18.6	160		40.2	8742	16.7	119		33.6	6973	19.6	123
neos-808072		56.4	666	98.6	45		63.6	330	98.6	92		73.8	294	98.6	35
neos-810286		69.0	1168	-	7		78.0	1236	-	7		94.2	1660	-	4
neos-820879		1968.0	49030	65.3	213		1073.4	20729	74.2	285		1377.6	54936	61.6	100
neos-824661	T	>3600.0	4159	-	197	T	>3600.0	3867	-	227	T	>3600.0	3555	-	128
neos-824695		580.8	1389	-	155		708.0	3224	-	157		78.0	480	-	68
neos-825075		166.8	68214	0.0	73		88.2	29403	0.0	144		24.0	6547	0.0	115
neos-826224	T	>3600.0	56670	-	132	T	>3600.0	46538	-	119	T	>3600.0	12919	-	95
neos-826250		61.8	684	-	144		193.2	5126	-	157		477.0	5708	-	108
neos-826812		31.8	1	-	117		37.8	16	-	112		75.0	269	-	48
neos-827175		34.2	117	-	35		23.4	53	-	35		60.0	47	-	34
neos-829552		1309.2	8401	3.8	24		1716.6	9097	3.3	17		1809.6	8423	5.0	19
neos-839859		225.0	14601	17.3	43		148.8	14465	12.5	35		130.2	16290	9.5	17
neos-860300		74.4	607	38.3	4		74.4	607	38.3	4		63.0	600	38.3	1
neos-863472		1522.2	2439548	0.0	20		1314.6	2083936	0.0	19		1291.8	2168100	0.0	9
neos-881765		3.6	301	-	4		3.6	301	-	4		3.6	301	-	4
neos-885086		307.8	1608	-	88		247.8	844	-	77		117.6	564	-	5
neos-892255		250.8	8438	0.0	189		232.8	11918	0.0	160		157.8	11356	0.0	5
neos-905856		1032.6	105842	100.0	39	T	>3600.0	352387	50.0	41	T	>3600.0	222090	0.0	19
neos-906865		225.6	68483	25.4	47		323.4	95807	23.3	36		262.2	96849	16.6	15
neos-912015		51.6	10868	-	8		13.2	2401	-	10		15.0	2795	-	8
neos-933638		314.4	248	-	3		342.0	364	-	4		154.2	114	-	1
neos-933815		642.6	154625	98.7	578	T	>3600.0	832441	97.6	585	T	>3600.0	1237061	98.3	350

continued on next page

	support					integral support					adjusted distance				
instance	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts
neos-933966		733.2	1896	-	3		264.0	545	-	23		1291.2	803	-	1
neos-934531		53.4	25	100.0	259		52.2	12	100.0	374		51.6	12	100.0	77
neos-935627	T	>3600.0	9693	-	3	T	>3600.0	7891	-	5	T	>3600.0	9615	-	1
neos-935769		379.2	1145	-	9		249.0	679	-	17		543.6	1527	-	1
neos-936660	T	>3600.0	6655	-	4		3150.0	7741	-	10	T	>3600.0	3353	-	0
neos-937446		354.6	1112	-	7		459.0	1756	-	6		308.4	413	-	1
neos-937511		341.4	1353	-	7		2307.6	19673	-	9		621.0	1475	-	1
neos-942323		703.8	172451	0.0	149	T	>3600.0	954783	0.0	157	T	>3600.0	1223510	0.0	76
neos-948268		97.2	1936	-	15		958.8	6196	-	35		132.6	3056	-	1
neos-955215		1437.0	490815	97.5	378	T	>3600.0	1776629	99.3	398		674.4	536203	99.6	217
neos-957270	T	>3600.0	233226	-	36	T	>3600.0	259609	-	38	T	>3600.0	251965	-	20
neos10		453.0	16	62.2	819		406.2	17	67.6	1532		604.2	20	70.3	410
neos11		778.8	26514	0.0	63		1606.2	55148	0.0	124		768.0	26685	0.0	2
neos2	T	>3600.0	442475	21.7	180		882.0	114495	17.9	150		1477.8	277044	16.9	116
neos20		321.6	89378	7.3	1086		225.6	79602	4.9	887		377.4	266999	0.0	113
neos21		111.0	23947	37.3	49		115.2	29185	16.4	56		135.0	34170	16.4	14
neos22		124.8	12353	99.7	502		144.6	18951	99.6	464		71.4	11891	99.7	151
neos6		2851.2	39627	-	143		1197.0	20550	-	53	T	>3600.0	83145	-	8
neos648910	T	>3600.0	2488807	0.0	662	T	>3600.0	2449897	0.0	596	T	>3600.0	3399244	0.0	108
neos671048		131.4	4126	50.1	137		82.2	2608	35.2	89		105.0	4118	24.9	5
neos8		142.2	1	100.0	343		136.2	1	100.0	266		141.6	1	99.6	2
neos897005		604.2	108	100.0	20		309.6	6	100.0	40		549.6	97	100.0	5
nw04		297.6	37	90.2	18		196.8	247	56.7	12		163.2	90	87.7	14
pg5_34		2562.0	157043	98.8	1406		1837.8	163649	98.8	1322		994.2	144151	98.8	1304
prod1		2689.8	3845517	52.0	59		1233.6	1974887	54.3	79	T	>3600.0	4299257	48.5	27
qap10		144.6	229	19.3	10		214.2	231	19.3	47		186.0	232	19.3	23
qiu		197.4	31603	4.4	57		169.8	28935	5.1	10		118.8	27941	0.0	1
qnet1		127.2	49	40.2	151		151.8	67	45.1	159		216.0	109	40.9	109
qnet1_o		241.8	127	70.7	145		220.2	90	72.0	194		60.6	85	70.2	96
ran10x26		209.4	64821	49.1	212		223.8	83450	47.8	182		180.6	67072	49.8	144
ran12x21		492.6	140569	54.4	367		457.2	181415	53.7	301		598.2	298402	52.8	218
ran13x13		111.0	60945	55.1	204		118.8	82865	56.6	157		154.8	99761	56.8	130
ran16x16		2308.8	563364	58.6	512		2019.0	847421	54.8	268		3021.0	1688716	54.2	248
rmatr100-p10		432.0	23163	0.1	1		487.8	25389	0.0	1		497.4	28213	1.8	1
rmatr100-p5		204.6	4189	0.1	1		204.6	4189	0.1	1		193.8	3259	1.4	1
rout		2694.0	1112468	6.0	160		1534.2	1021712	7.5	128		802.8	1196404	0.0	13
set1ch	T	>3600.0	1050846	91.7	750	T	>3600.0	1433527	86.7	616	T	>3600.0	1401621	89.1	485
sp98ir		414.6	47102	2.2	188		291.6	34971	1.8	198		311.4	33943	4.7	22
swath1		51.0	3180	14.3	71		219.0	15953	14.0	77		388.2	41502	13.9	52
swath2		664.2	60971	22.2	79	T	>3600.0	336451	23.4	66		1657.8	138555	22.1	91
Test3		66.6	139	97.8	395		76.2	1705	93.1	284		89.4	2666	93.5	133
vpm2		54.6	31314	80.6	383		37.8	19477	80.7	342		39.6	31262	80.2	209

Table 9: Detailed results (4). The columns headed "S" contain a "T" if an instance hit the time limit or an "F" if the solver failed. The columns headed "time (sec.)" and "# nodes" show the solution time and the number of branching nodes explored. The columns headed "gap cl. (%)" and "# cuts" contain the amount of integrality gap closed at the root node and the number of cutting planes which were added to the LP relaxation.

instance	distance variant					distance + obj. parallelism + int. support				
	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts
30_05_100	T	>3600.0	166820	100.0	1	T	>3600.0	169141	100.0	1
30_95_100		615.6	5416	-	0		611.4	5416	-	0
30_95_98		95.4	833	100.0	1		94.8	833	100.0	1
acc1		28.8	744	-	32		22.2	432	-	36
acc3	T	>3600.0	4125	-	10		486.0	5205	-	10
aflow30a		384.0	101899	44.5	206		199.8	43662	53.1	271
air04		325.2	2326	10.9	16		305.4	2459	16.2	20
air05		229.8	20121	3.5	7		126.6	13791	2.7	4
aligninq		148.8	2218	22.8	25		319.8	2018	36.5	62
bienst1		189.0	45755	24.0	63		182.4	50317	13.4	99
bienst2		1239.0	329627	16.3	114		1079.4	334637	10.2	99
cap6000		48.0	3943	22.7	7		75.0	3965	18.9	16
dcmulti		49.2	179	71.3	221		45.6	79	82.9	303
eilB101		360.0	50505	9.4	25		301.2	47595	8.8	18
fiber		78.0	432	86.2	131		63.6	152	93.1	191
gesa2_o		119.4	31106	72.7	275		64.2	9704	96.7	346
gesa3_o		22.2	385	73.4	144		34.8	217	79.8	263
harp2		2539.2	2654366	54.5	110		1399.8	899172	61.7	344
l152lav		67.8	11248	16.2	43		55.8	11418	17.7	41
map18		1918.8	2973	18.4	149		2092.8	3732	27.6	391
map20		1422.6	3261	19.0	131		2047.2	4354	29.9	287
markshare_4_0		1392.0	1750919	0.0	5		1380.6	1750919	0.0	5
mas76		554.4	1405143	4.8	3		399.6	805703	2.8	3
mitre		300.0	7442	95.4	1071		207.0	3447	92.8	2044
mod011		112.8	5591	63.0	725		118.8	2989	72.2	878
neos-1053591		2347.8	4024718	0.0	3		1861.8	3303428	0.0	16
neos-1061020		311.4	4844	4.6	23		325.8	4998	5.0	40
neos-1122047		69.6	51	-	0		69.0	51	-	0
neos-1151496		58.8	372	100.0	16		48.0	476	100.0	15
neos-1171448		29.4	152	-	1		45.0	214	-	41
neos-1171737	T	>3600.0	93362	-	2	T	>3600.0	120201	-	33
neos-1200887		308.4	184967	0.0	1		306.0	184967	0.0	1
neos-1211578		547.2	848885	0.0	3		588.0	947955	0.0	4
neos-1228986	T	>3600.0	5123244	0.0	2		2491.8	3768421	0.0	11
neos-1367061		95.4	121	100.0	40		164.4	89	100.0	92
neos-1396125		2338.2	586694	23.4	36		2353.2	458022	24.2	78
neos-1420205		1.2	2316	37.5	15		123.0	250320	45.0	35
neos-1440447		1198.2	1256872	0.0	6		690.0	843891	0.0	6
neos-1460265		1.8	444	-	163	T	>3600.0	1716175	-	200
neos-1489999		583.2	280880	9.5	39		579.6	314943	7.1	35
neos-1582420		1965.6	151674	12.4	33		1621.8	128018	41.6	46
neos-1597104		60.6	478	-	6		21.0	60	-	6
neos-430149		1988.4	2220811	0.0	40		1708.2	1857697	0.0	64
neos-480878	T	>3600.0	266197	64.1	7	T	>3600.0	269120	64.1	7
neos-503737		467.4	21224	-	19		211.8	6589	-	35
neos-504674		1450.2	958030	40.2	197		835.2	481187	42.3	304
neos-504815		349.8	212814	31.3	200		223.2	113911	35.1	268
neos-512201		881.4	538171	41.4	215		427.2	181020	42.8	323
neos-522351	T	>3600.0	494796	63.8	104	T	>3600.0	439697	81.9	340
neos-525149		193.8	48680	0.0	47		243.0	66257	0.0	72
neos-530627		1647.6	3419721	45.3	3		0.2	1	100.0	5
neos-538867		246.6	270164	0.0	16		339.6	303811	0.0	54
neos-538916		373.8	362381	0.0	23		508.8	506244	0.0	23
neos-551991		1225.2	90865	31.4	57		918.6	107038	31.4	80
neos-555694		31.8	6650	0.0	28		15.6	2021	0.0	91
neos-555771		70.2	13950	0.0	69		64.2	13973	0.0	104
neos-570431		105.6	25387	25.1	35		58.2	12169	25.1	40
neos-583731		53.4	28934	-	207		540.6	293461	-	220
neos-584851		329.4	296311	0.0	75		259.2	188740	0.0	142
neos-585192		503.4	97399	60.7	97		2950.2	482160	60.7	150
neos-595905	T	>3600.0	854726	39.8	340		1845.0	401148	63.4	407
neos-598183		757.2	125639	37.3	355		894.0	162672	90.8	468
neos-603073	T	>3600.0	387687	6.6	409	T	>3600.0	269044	7.7	517
neos-611838		471.6	16885	65.2	454		197.4	6193	77.2	581
neos-612125		193.8	2155	68.1	473		139.8	1493	74.0	585
neos-612143		172.2	4855	66.9	443		157.8	3129	76.7	573
neos-612162		306.0	7811	64.7	462		149.4	3843	78.0	555
neos-686190		894.6	90416	6.5	26		814.8	86389	5.7	40
neos-709469		799.8	906893	-	20		1.2	409	-	65
neos-775946	T	>3600.0	514954	29.1	98	T	>3600.0	534903	59.7	152
neos-777800		114.0	10229	-	0		28.8	1580	-	1
neos-785914		22.8	1611	-	89		172.2	14546	-	77
neos-791021	F	>3600.0	490676	7.7	68	T	>3600.0	639014	30.8	109
neos-796608	T	>3600.0	3343092	0.0	12	T	>3600.0	3429114	0.0	12
neos-803219		79.2	54451	25.8	59		84.0	55325	35.4	56
neos-803220		69.6	81907	30.0	30		85.2	76613	30.5	46
neos-806323		64.2	15692	27.2	124		49.2	11131	35.2	147
neos-807639		59.4	26373	23.3	99		49.8	20033	29.2	148
neos-807705		45.0	10288	17.5	119		46.2	9495	24.0	144
neos-808072		150.6	455	98.6	42		58.8	647	98.6	29
neos-810286		62.4	1048	-	2		81.6	1615	-	4
neos-820879		1930.8	61876	64.5	99		2269.2	62556	59.7	109
neos-824661	T	>3600.0	3555	-	128		1726.2	2748	-	140
neos-824695		78.0	480	-	68		523.2	2977	-	71
neos-825075		58.2	35445	0.0	5		94.8	37798	0.0	88
neos-826224	T	>3600.0	21134	-	70	T	>3600.0	124613	-	100

continued on next page

	distance variant					distance + obj. parallelism + int. support				
instance	S	time (sec.)	# nodes	gap cl. (%)	# cuts	S	time (sec.)	# nodes	gap cl. (%)	# cuts
neos-826250		300.0	3154	-	68		104.4	2678	-	179
neos-826812		75.6	269	-	48		40.2	87	-	53
neos-827175		60.0	47	-	34		59.4	47	-	34
neos-829552		1719.6	9049	3.4	16		1776.6	8501	3.5	20
neos-839859		107.4	18550	7.4	14		132.6	13407	6.9	27
neos-860300		63.0	600	38.3	1		61.8	595	38.3	4
neos-863472		1414.8	2259330	0.0	11		1500.6	2484084	0.0	13
neos-881765		3.6	301	-	4		3.6	301	-	4
neos-885086		143.4	652	-	4		221.4	821	-	29
neos-892255		160.8	11116	0.0	4		216.6	10204	0.0	126
neos-905856		3454.2	373205	100.0	38	T	>3600.0	405660	0.0	20
neos-906865		49.2	13625	30.8	23		172.2	51543	23.7	48
neos-912015		18.6	3363	-	8		33.0	6684	-	9
neos-933638		154.2	114	-	1		347.4	249	-	2
neos-933815	T	>3600.0	607800	89.9	365		477.0	149757	98.8	454
neos-933966		1291.8	803	-	1		211.8	305	-	15
neos-934531		53.4	51	100.0	87		51.6	6	100.0	143
neos-935627		185.4	176	-	1	T	>3600.0	7562	-	1
neos-935769		543.0	1527	-	1		540.6	1527	-	1
neos-936660	T	>3600.0	3353	-	0		2902.8	6503	-	1
neos-937446		309.0	413	-	1		307.2	413	-	1
neos-937511		621.0	1475	-	1		316.2	613	-	12
neos-942323		340.8	100368	0.0	112		720.0	193393	0.0	132
neos-948268		132.6	3056	-	1		1574.4	6252	-	33
neos-955215	T	>3600.0	2237195	99.3	184		385.2	263955	99.6	317
neos-957270	T	>3600.0	271158	-	28		4.2	72	-	22
neos10		381.6	18	63.1	258		337.8	18	62.2	561
neos11		712.8	28329	0.0	2		1555.2	55449	0.0	2
neos2		354.0	129354	10.4	60	T	>3600.0	566187	17.3	146
neos20		470.4	326473	2.4	166		244.2	117230	0.0	271
neos21		105.6	27334	16.4	24		124.2	28673	37.3	25
neos22		106.8	25559	99.6	160		152.4	20621	99.6	206
neos6		2526.0	54360	-	30	T	>3600.0	88419	-	38
neos648910	F	>3600.0	3371502	0.0	100		453.6	523861	0.0	200
neos671048		106.8	4118	24.9	5		149.4	4573	36.4	54
neos8		141.0	1	100.0	7		139.2	1	98.2	72
neos897005		931.2	108	100.0	18		437.4	216	100.0	9
nw04		166.8	67	87.7	14		239.4	106	66.4	9
pg5_34	T	>3600.0	339340	91.5	1235		955.8	126937	98.8	1289
prod1		1558.2	2852684	46.4	15	T	>3600.0	4976742	46.5	24
qap10		156.0	229	19.3	13		205.8	224	19.3	42
qiu		101.4	18195	2.6	17		118.2	27941	0.0	1
qnet1		183.0	113	33.0	80		157.8	90	35.8	165
qnet1_o		126.0	191	63.0	125		201.6	77	70.3	169
ran10x26		198.6	95558	44.8	129		183.0	57643	49.8	171
ran12x21		641.4	322335	50.8	206		457.2	204846	53.8	238
ran13x13		172.8	141441	49.1	99		89.4	45634	62.0	160
ran16x16	T	>3600.0	1597061	53.0	229		2919.0	1091173	56.1	293
rmatr100-p10		456.6	24307	0.6	1		499.2	27207	1.8	1
rmatr100-p5		218.4	4823	0.5	1		203.4	4287	1.4	1
rout		784.2	522381	10.8	82		1088.4	618583	12.7	125
set1ch	T	>3600.0	1445694	77.1	415	T	>3600.0	1668067	86.6	507
sp98ir		393.0	47830	4.2	10		280.8	33381	3.8	40
swath1		63.6	12042	13.9	20		47.4	2759	14.3	73
swath2		2437.8	351729	22.8	40		638.4	48402	22.1	101
Test3		83.4	1933	91.7	136		78.0	2565	91.6	196
vpm2		66.0	73254	68.1	144		55.8	38474	81.3	262

Table 10: Detailed results (5). The columns headed "S" contain a "T" if an instance hit the time limit or an "F" if the solver failed. The columns headed "time (sec.)" and "# nodes" show the solution time and the number of branching nodes explored. The columns headed "gap cl. (%)" and "# cuts" contain the amount of integrality gap closed at the root node and the number of cutting planes which were added to the LP relaxation.