# Objective scaling ensemble approach for integer linear programming

**Weili Zhang[1] · Charles D. Nicholson[1]**

## Abstract

The objective scaling ensemble approach is a novel two-phase heuristic for integer linear programming problems shown to be effective on a wide variety of integer linear programming problems. The technique identifies and aggregates multiple partial solutions to modify the problem formulation and significantly reduce the search space. An empirical analysis on publicly available benchmark problems demonstrate the efficacy of our approach by outperforming standard solution strategies implemented in modern optimization software.

**Keywords** Integer programming · Heuristics · Neighborhood search

## 1 Introduction

Integer programming (IP) is a fundamental approach to NP-hard combinatorial problems that arise in wide range of application areas including production, scheduling, finance, network design, and others (Nemhauser and Wolsey 1988; Zhang and Wang 2017; Zhang et al. 2018; Zhang and ZHAO 2010; Zhang and Nicholson 2016a; Zhang and Yao 2010; Zhang et al. 2017; Zhang and Wang 2016). There are both linear and non-linear formulations of IP problems. In this investigation we focus on the former. Broadly defined, an mixed integer linear program (MILP) aims at optimizing a linear objective function (without loss of generality we assume minimization) subject to a set of linear equality/inequality constraints over real and integer/binary variables. Borrowing notation from Fischetti and Lodi (2003), we define the MILP problem as

$$(\text{ILP}) \quad \min z(\mathbf{x}) = \mathbf{c}^T \mathbf{x} \tag{1}$$

$$A\mathbf{x} \geq \mathbf{b} \tag{2}$$

$$x_j \text{ integer} \quad \forall j \in \mathcal{G} \tag{3}$$

✉ Charles D. Nicholson
  cnicholson@ou.edu

[1] School of Industrial and Systems Engineering, University of Oklahoma, Norman, OK, USA

$$x_j \in \{0, 1\} \quad \forall j \in \mathcal{B} \tag{4}$$

$$x_j \geq 0 \quad \forall j \in \mathcal{N}. \tag{5}$$

Here, **c** is an $n$-dimensional vector of costs, **x** is an $n$-dimensional vector of decision variables, $A$ is an $m \times n$ constraint matrix, **b** is an $m$-dimensional vector of parameters, and $\mathcal{N}$ is a set of variable indices $\{1 \dots n\}$ partitioned into three sets, $\mathcal{N} = \{\mathcal{B}, \mathcal{G}, \mathcal{C}\}$ associated with binary, integer, and continuous variables, respectively. If $\mathcal{C}$ and $\mathcal{I} = \{\mathcal{B}, \mathcal{G}\}$ are non-empty, the problem is a *mixed integer* linear program (MILP). If $\mathcal{C} = \emptyset$ and $\mathcal{G} \neq \emptyset$, the problem is a *pure integer* problem. If $\mathcal{C}$ and $\mathcal{G}$ are empty, but $\mathcal{B} \neq \emptyset$, it is a *binary* programming (BP) problem. If $\mathcal{G}$ is empty, but $\mathcal{C}$ and $\mathcal{B}$ are not, it is a *mixed binary* problem (MBP). Finally, if $\mathcal{I}$ is empty, the problem is not an integer problem but a linear programming (LP) problem. Various IP solution approaches entail temporarily removing the integrality constraints in (3) and (4) solving the associated LP relaxation.

Commercial MILP solvers, such as CPLEX and Gurobi used in both academia and industry, leverage branch-and-bound and cutting planes algorithms with linear programming relaxation to find exact optimal solution (Lodi 2010). Due to the time and/or resource complexity of finding exact MILP solutions, there is value in obtaining near-optimal solutions to such problems quickly. A large body of research has been directed towards finding solution approaches applicable to particular subclasses of MILP problems, e.g. fixed-charge network flow problems (Bertsimas and Sim 2003), network design (Crainic et al. 2000), vehicle routing (Gulczynski et al. 2011), and scheduling (Hoffman and Padberg 1993; van den Akker et al. 2000; Beliën 2007). General-purpose MILP solution approaches on the other hand include "pivot and complement" for BP problems (Balas and Martin 1980), "pivot and shift" for MILP problems (Balas et al. 2004), "pivot, cut and dive" (Eckstein and Nediak 2007), OCTANE for BP problems (Balas et al. 2001), relaxation induced neighborhood search (RINS) (Danna et al. 2005), local branching (Fischetti and Lodi 2003), feasibility pump (Fischetti et al. 2005; Achterberg and Berthold 2007; Bonami et al. 2009), and others (Blum and Roli 2003; Patel and Chinneck 2007). Approximate solutions may be of sufficient quality to stand on their own or be used in combination with an exact procedure to find feasible solutions. Modern solvers incorporate many heuristics as part of the overall optimization strategy to improve time to solution (Linderoth and Lodi 2010).

In this paper, we introduce an heuristic technique and basic framework suitable for a wide variety of MILP problems. This algorithm, which we call the ensemble approach (OSEA), is inspired by RINS (Danna et al. 2005) and slope scaling (Kim and Pardalos 1999) techniques. Section 2 reviews these techniques, explains the motivation of OSEA, and formally defines the framework. Section 3 describes the problem testbed and reports the computational results. We summarize the work in Sect. 4.

## 2 Objective scaling ensemble approach

### 2.1 Motivation

Relaxation induced neighborhood search (Danna et al. 2005) is one of several heuristic techniques used in conjunction with exact solution approaches to MILP problems.

Branch-and-bound (or branch-and-cut) explore the MILP solution space by iteratively fixing one or more integer variables (a partial solution) and solving the remaining sub-problem as a linear relaxation. The partial solutions are typically referred to as nodes in the search tree. The best integer feasible solution found during the process is called an incumbent solution and is updated whenever a better feasible solution is found. The process is iterated until the entire solution space has been implicitly examined and a provably optimal solution is found (assuming a feasible solution exists). Many heuristic search techniques, on the other hand, search a *neighborhood*, a local space "close" to a particular point within the solution space (as defined by some distance measure), to find improved solutions, e.g. local branching (Fischetti and Lodi 2003), tabu search (Glover and Laguna 1999), simulated annealing (Kirkpatrick et al. 1983), and machine learning (Zhang and Nicholson 2016b; Nicholson and Zhang 2016) See Gendreau and Potvin (2010) for an excellent resource regarding a wide variety of metaheuristic techniques. The local search is then repeated based on the neighborhood of the improved solution. Neighborhood search procedures are often terminated based on some pre-specified criteria and do not guarantee the global optimality of the final solution. Climer and Zhang (2006) proposed a search strategy, a.k.a. cut-and-solve, which is proved to find optimal solution and termination without using common branching. The authors have demonstrated the performance on Asymmetric Traveling Salesman Problem.

RINS employs information from the branch-and-bound process to form a search neighborhood of an incumbent feasible solution. The intuition is that some subset of variables in a linear relaxation for a given search node will share values with the current incumbent solution. The variables which do agree are fixed to their incumbent values. The solution space for the resulting subproblem defines the neighborhood of the incumbent and this space is searched using an exact technique. Any integer feasible solution found is a globally feasible solution and possibly will improve the incumbent solution. The sub-IP problem is potentially large and some stopping criterion is used to terminate the local search. The master branch-and-bound process is resumed with a potentially improved incumbent. RINS can be employed at any search node. At each node the LP relaxation may result in a different solution and thus the overall search is diversified.

Slope-scaling, and in particular the dynamic slope scaling procedure (DSSP) (Kim and Pardalos 1999), was originally designed for the fixed-charge network flow (FCNF) problems and has been applied to a various problem types including the piecewise linear network flow problem (Kim and Pardalos 2000), the multicommodity fixed-charge network problem (Crainic et al. 2004), the multicommodity location problem (Gendron et al. 2003), the minimum toll booth problem (Bai et al. 2010), and stochastic integer programming (Shiina and Xu 2012). With respect to the original application, DSSP employs a series of linearizations of the FCNF discontinuous objective function in (6),

$$\min \sum_{(i,j)\in A} (c_{ij}x_{ij} + f_{ij}y_{ij}) \tag{6}$$

where $A$ is the set of arcs in the network, $\mathbf{x}$ is a vector of continuous arc flow values, $\mathbf{y}$ is a vector of binary decision variables, and the cost vectors $\mathbf{c}$ and $\mathbf{f}$ are the unit flow costs and fixed costs, respectively. DSSP removes the binary variables $\mathbf{y}$ from the objective, iteratively scales the fixed cost $\mathbf{f}$ and adds them to the variable flow cost $\mathbf{c}$. Similar to RINS, OSEA uses information from linearized formulations to form a smaller search space. The relaxation formulation differs from RINS in that the relaxation is not based on a search node (partial solution), but based on a series of slope-scaling inspired relaxations.

OSEA defines the search space based on variable agreement between (i) an incumbent solution and one or more solutions of the linearized formulations (i.e., an ensemble of solutions) or (ii) entirely from agreement between solutions in the ensemble. Solution ensembles have been exploited in a variety of ways such as variable fixing based on value agreement among solutions (e.g., as in RINS) or "voting" among the solutions (e.g., 4 of 5 solutions have variable $x_1 = 17$). The latter is commonly employed in the field of statistical learning (e.g., Breiman 1996). OSEA takes a relatively conservative approach in using the ensemble to define a sub-MIP problem to be solved exactly.

## 2.2 OSEA framework

In many instance of large, real-world IP problems, only a small percentage of the integer variables have non-zero values in the optimal solution. It is worth noting that among the larger MILP instances available in the IP benchmark problem library MIPLIB 2010 (Koch et al. 2011), the relative number of non-zero integer variables in the optimal solutions is very low. Table 1 shows the summary statistics (minimum, first quartile, median, mean, third quartile, and maximum) for the percentage of integer variables used in the optimal solution for the 36 MIPLIB 2010 solved problems that have at least 10,000 integer variables. The median is 1.3% and 75% of the optimal solutions use less than 2.89% of the possible integer variables. Motivated by this feature of IP problems, OSEA attempts to *eliminate* integer variables from the problem formulation and the ensemble aggregation method for OSEA is designed with this characteristic in mind.

Let $\mathcal{E} = \{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_k\}$ denote an ensemble of $k$ solutions (possibly including infeasible solutions) to an MILP problem. The set of solutions may be generated through slope-scaling techniques, LP relaxations, known feasible solutions, accumulating incumbent solutions in a branch-and-bound algorithm, or other methods. In particular, OSEA fixes the $j$th integer variable to 0 if for all solutions $s \in \mathcal{E}$, the $j$th variable, $s_j$, equals 0. That is,

**Table 1** Integer variables used in large MIPLIB instances

| Min (%) | Q1 (%) | Median (%) | Mean (%) | Q3 (%) | Max (%) |
| --- | --- | --- | --- | --- | --- |
| 0.01 | 0.28 | 1.30 | 3.69 | 2.89 | 34.41 |

$$x_j \leftarrow \begin{cases} \text{Fix to } 0 & \text{if } s_j = 0 \quad \forall s \in \mathcal{E} \\ \text{Do not fix} & \text{otherwise} \end{cases} \quad \forall j \in \mathcal{I}.$$

The integer variables which are left open in the corresponding sub-MIP problem form a reduced search space. An exact search of the reduced problem space produces the OSEA solution and objective.

It is important to note that the ensemble does not necessarily consist of high quality solutions to the original problem. In fact, from initial testing we place a priority on diversity of quality. If the solutions in $\mathcal{E}$ are diverse, then the variables that are unused by every solution in the ensemble share at least one characteristic: they are each "unattractive" to a wide range of solutions. Since OSEA fixes variables to 0, by allowing poorer quality solutions in the ensemble, we take a more conservative approach. That is, only variables that are not used among a variety of solutions (e.g., good, median, poor) are discarded.

Moreover, if a given integer variable is not used in a linearized optimal solution when the cost is adjusted to a fraction of its original cost, then the intuition is that the integer variable is not likely useful in the original problem. The absolute cost associated with a variable is not as important as the cost relative to other variables. The iterative scheme in slope-scaling techniques and in OSEA updates individual variable costs throughout the process. This update scheme dynamically affects the relative costs of the variables. Variables which may be too costly to use in a linearized solution during the earlier iterations may become cost-effective in the latter ones.

The ensemble $\mathcal{E}$ must be populated with solutions or partial solutions. For OSEA, this is primarily accomplished in the objective scaling iteration phase. The scaling process is now described.

OSEA scales the coefficients of the discrete variables and iteratively solves the relaxed problem (7),

$$(\text{LP}_n) \quad \min z_n^{\text{LP}}(\mathbf{x}) = \sum_{j \in \mathcal{C}} c_j x_j + \sum_{j \in \mathcal{I}} \bar{c}_j^n x_j$$

$$A\mathbf{x} \geq \mathbf{b} \tag{7}$$
$$0 \leq x_j \leq 1 \quad \forall j \in \mathcal{B}$$
$$x_j \geq 0 \quad \forall j \in \mathcal{N}$$

where $\bar{c}_j^n$ for $j \in \mathcal{I}$ is the scaled cost coefficient of the integer variable. Let $\tilde{\mathbf{x}}^n$ denote the solution to $\text{LP}_n$. Intuitively, the scaling factor can be used to effectively reflect the linear factor of the current objective function. It is updated to the latest marginal fixed cost based on the current statuses of the corresponding positive variable and will stay the same when it reaches the break-even point (a concept of economic viewpoint). This solution is used to update the integer coefficient for the next iteration $n + 1$ as follows,

$$\bar{c}_j^{n+1} \leftarrow \frac{c_j}{\tilde{x}_j^n + 1} \quad \forall j \in \{\mathcal{I} : \tilde{x}_j^n > 0\} \tag{8}$$

Note in DSSP, the fixed cost value is scaled by $1/\tilde{x}_j^n$, however there are two benefits that result from modifying this for general application. First, since MILP problems may have negative integer variable cost coefficients, as the relaxed solution approaches 0, the scaled costs may approach negative infinity,

$$\lim_{\tilde{x}_j^n \to 0} \frac{-1}{\tilde{x}_j^n} = -\infty$$

and a counterintuitive result ensues, namely the attractiveness of the variable increases without bound as the value of the variable decreases. This effect is bounded by a simple modification of the denominator in Eq. 8. And secondly, the resulting bound is intuitive

$$\lim_{\tilde{x}_j^n \to 0} \frac{c_j}{\tilde{x}_j^n + 1} = c_j.$$

Slope-scaling dynamically modifies the costs of different variables throughout the search process to alter their relative attractiveness in the relaxations. While it is true that modification in Eq. (8) impacts the appealing characteristic of the final iteration $N$ of DSSP in which the scaled objective value reflects the true solution cost (i.e., it includes the full fixed cost incurred in the corresponding network flow solution),

$$\sum_{j \in \mathcal{C}} c_j \tilde{x}_j^N + \sum_{j \in \mathcal{I}} \bar{c}_j^N [\tilde{x}_j^N > 0]$$

where $[x_j^N > 0]$ denotes the Iverson bracket which returns a 1 if $x_j^N > 0$ and 0, otherwise. However, this outcome is not critical to the success of DSSP. That is, the best solutions from DSSP are not necessarily found in the final iteration (Nahapetyan and Pardalos 2008). At earlier iterations $n < N$, the scaled fixed costs do not represent the true value:

$$\sum_{j \in \mathcal{I}} \bar{c}_j^n \tilde{x}_j^n \neq \sum_{j \in \mathcal{I}} \bar{c}_j^N [\tilde{x}_j^N > 0].$$

This suggests that the search path induced by the procedure is of more importance than the objective value of the final iteration.

The integer variable cost coefficients are initialized to a fraction of the original cost by scaling by the inverse of the relatively large value $M$,

$$M = \sum_{j \in \mathcal{I}} |c_j|.$$

Thus,

$$c_j^0 \leftarrow \frac{c_j}{M} \qquad \forall j \in \mathcal{I}$$

and the update scheme is then,

$$c_j^n = \begin{cases} \frac{c_j}{\tilde{x}_j^{n-1}+1} & \text{if } \tilde{x}_j^{n-1} > 0 \\ c_j^{n-1} & \text{otherwise} \end{cases} \quad \forall j \in \mathcal{I}.$$

In the objective scaling phase of OSEA there are $N > 0$ iterations and consequently $N$ linearized solutions: a subset of which will be added to the ensemble $\mathcal{E}$. For large $N$, the number of different solutions that could be added to $\mathcal{E}$ is also large. If $|\mathcal{E}|$ is too large, the reduced search space is potentially too large for practical purposes. Therefore, we select a subset of the iterated linear solutions to be added to the ensemble. Let $\mathcal{S}$ denote this subset. On the other hand, if $|\mathcal{E}|$ is too small or if it does not contain sufficient diversity, then there may be insufficient options in the search space to generate good solutions to the original MILP. A number of possible strategies can be designed to build the set $\mathcal{S}$. We devise one such possible strategy to emphasize ensemble diversity in Sect. 3.

If the ensemble $\mathcal{E}$ does not contain a feasible solution then OSEA may or may not produce a feasible solution. However, if a feasible solution is included in $\mathcal{E}$ then OSEA is guaranteed to find a feasible solution in the reduced search space. In our implementation of the framework, we take a hybrid approach in which we utilize the already existing pre-processing, heuristics, and branch-and-cut algorithms readily available in commercial software such as Gurobi and CPLEX to briefly search for a feasible solution that can be added to $\mathcal{E}$. This will be described in more detail in Sect. 3.

There are multiple possible stopping criterion for OSEA. Similar to DSSP, the iterative objective scaling phase will stop once there are no new cost coefficient updates for the integer variables. In some cases it might be prudent to provide an upper limit on the total number of iterations allowed for the iterative scaling procedure. Let $N_{\max}$ denote the max allotted iterations. Additionally, since OSEA is meant as an heuristic technique to quickly reduce the search space of complex problems, a time limit could also be imposed on the scaling phase. The complete OSEA logic (including the iteration limit, but not the time limit) is summarized in Fig. 1.

## 3 Computation results

### 3.1 Experimental design

The MIPLIB 2010 (Koch et al. 2011) is a publicly available library of pure and mixed integer programming problem instances assembled by researchers and practitioners over several years. This library of benchmark problems is used in evaluating software performance of commercial solvers (Gurobi Optimization 2012). The library contains 361 instances classified into 3 difficulty levels: 185 *easy*, 42 *hard*, and 134 *open* problems. The latter problem class contains the instances which have yet to be solved optimally. The instances are further described by 8 characterizations types: *benchmarks* (B) are solvable within 2 h on a PC, *infeasible* (I), *primal* (P) instances

**Data:** ILP instance, $N_{\max}$

**Result:** $z_{\text{OSEA}}$, $\mathbf{x}_{\text{OSEA}}$

**begin**

    compute $M$: $M \leftarrow \sum_{j \in \mathcal{I}} |c_j|$

    initialize integer costs : $c_j^0 \leftarrow \frac{c_j}{M}$ $\forall j \in \mathcal{I}$

    optional: initialize $\mathcal{E}$ by including one or more feasible solution(s) to MILP

    $n \leftarrow 0$

    **while** $n \leq N_{max}$ **do**

        $\tilde{}^n\mathbf{x} \leftarrow$ solution to Problem $\text{LP}_n$

$$\bar{c}_j^n \leftarrow \begin{cases} \dfrac{c_j}{(\tilde{x}_j^{n-1} + 1)} & \text{if } \tilde{x}_j^{n-1} > 0 \\ \bar{c}_j^{n-1} & \text{otherwise} \end{cases} \quad \forall j \in \mathcal{I}$$

        **if** $\bar{c}_j^n = \bar{c}_j^{n-1}$ $\forall j \in \mathcal{I}$ **then**

            **break**

        $n \leftarrow n + 1$

    $\mathcal{S} \leftarrow$ a subset of $\{\tilde{}^1\mathbf{x} . . , \tilde{}^N\mathbf{x}\}$

    $\mathcal{E} \leftarrow \mathcal{E} \cup \mathcal{S}$

$$x_j \leftarrow \begin{cases} \text{fix to } 0 & \text{if } s_j = 0 \ \forall s \in \mathcal{E} \\ \text{do not fix} & \text{otherwise} \end{cases} \quad \forall j \in \mathcal{I}$$

    $z_{\text{OSEA}}$, $\mathbf{x}_{\text{OSEA}} \leftarrow$ solve reduced MILP problem

**Fig. 1** Objective scaling ensemble approach for MILP problems

have the LP relaxation objective equal to the optimal objective, *extra-large* problems (X), *reoptimize* (R) instances require a relatively long time to solve the LP relaxations, *tree* (T) instances have a large number of enumeration trees, *unstable* (U) instances have poor numerical properties, and *challenge* (C) instances which are classified generally as difficult to solve. The majority of the instances in the library also include information relating to the problem application area (e.g. lot sizing, open pit mining, network design, etc.)

OSEA is tested on a 170 problem subset of the 361 MIPLIB problems. Since OSEA is appropriate only for problems with integer variables in the objective function, any problems without this characteristic are discarded (87 problems). Infeasible problems are also eliminated from testing (22 problems). Instances which exceed the memory capacity of our available computer equipment (23 problems) and those in which a feasible solution was not found within 60 s are not included in the experimentation (59 problems).

The experiments will be conducted as follows. To increase the likelihood that an integer feasible solution is included in the ensemble $\mathcal{E}$, we will employ the existing preprocessing and heuristic algorithms of the optimization software by attempting to solve each problem for one second using commercially available optimization software. If a feasible solution is found in the time limit, it will be added to the ensemble. Regardless, the scaling phase begins and we select solutions discovered during this iterative stage to be added to the ensemble. Based on initial testing we emphasize ensemble diversity by including three solutions from the scaling iterations into the ensemble: the solutions associated with the best, worst, and median MILP objective values.

The maximum running time for OSEA including initial one second search, objective scaling phase, and solving the reduced problems is set to 60 s. To compare the OSEA solution quality with the state-of-the-art exact techniques, we will use the Gurobi optimization software version 5.6.3 with a time limit of 60 s. According to the Gurobi

product website, this solver includes 14 different MIP heuristics, 16 cutting plane strategies, and a variety of presolve techniques (Gurobi Optimization 2014). We use the default parameter settings for Gurobi with all heuristics activated (including RINS). The best objective value found using the default settings of the Gurobi optimization software and the time to find that value are recorded. All tests are performed on a Windows 7 64 bit machine with Intel Xeon CPU E5-1620 and 8 GB RAM with a single thread.

OSEA is inherently dependent on an IP solver. The technique itself is used in conjunction with a solver to reduce the IP search space. The empirical analysis will compare the results of using OSEA with a commercial solver against using the same commercial solver without OSEA. While the commercial solver is not being tuned specifically for each problem, the settings are identical for the OSEA test. That is, the only difference is the additional OSEA overhead to the commercial solver.

### 3.2 Experimental results

Let $t^p_{\text{OSEA}}$ and $t^p_{\text{standard}}$ denote the computing time to solve problem $p \in P$ using OSEA and the standard (solver without OSEA), respectively. Similarly, the best objective values found for problem $p \in P$ denoted by $z^p_{\text{OSEA}}$ and $z^p_{\text{standard}}$, respectively. Additionally, the solution gap (ILPgap) is used to evaluate solution quality. Let $G^p_X$ denote the MILP gap for approach $X \in \{\text{OSEA, standard}\}$ on problem $p \in P$,

$$
G^p_X = \frac{|z^p_{\text{bound}} - z^p_X|}{|z^p_X|} \times 100\%, \quad \forall X \in \{\text{OSEA, standard}\},
$$

where $z^p_{bound}$ is the known optimal objective value for *Easy* and *Hard* problems, and is the linear relaxed objective value for *Open* instances. Note if $z^p_X = 0$, a small positive value is added to the denominator.

Let $\gamma$ denote the percentage of integer variables removed in the objective scaling phase of OSEA. The distribution of $\gamma$ depicted in Fig. 2 shows that OSEA removes a significant percentage of integer variables for the majority of test bed. Overall, OSEA removes an average of 61.97% of the integer variables. Reducing the MILP solution space can lead to notable improvements in computation time.

The results are summarized in Table 2 where $n$ equals the number of problem instances in each cell. The detailed results for all datasets are appended in the "Appendix" ("Appendix A, B, C"). Overall, using a paired $t$-test, the difference in computing time and objective values of OSEA against the baseline are statistically significant. Both techniques find optimal solutions for 40% of the *Easy* and *Hard* problems, albeit the instances differ. That is, OSEA finds optimal solutions to certain instances that the standard approach failed to find within the time limit, and vice versa. Among the *Easy* and *Hard* problems, OSEA terminates faster than the standard technique, and while the average optimality gap is smaller for OSEA, the solution quality differences are not significant at a 95% confidence level. For the *Open* problems, the results are statistically significant: OSEA produces a higher quality solution within the time limit and does so in less time.
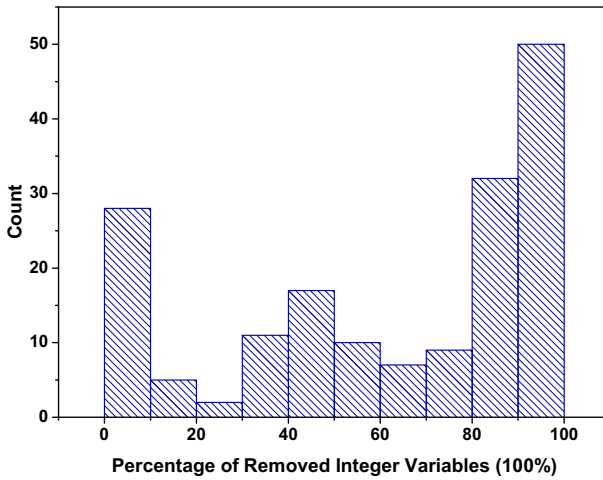
**Fig. 2** Distribution of the percentage of integer variables $\gamma$

**Table 2** Statistical analysis of performance by difficulty level

| Difficulty | $n$ | Computing time | | | Objective value | | |
|---|---|---|---|---|---|---|---|
| | | $t_{\text{OSEA}}$ | $t_{\text{standard}}$ | $p$-value | $G_{\text{OSEA}}$ (%) | $G_{\text{standard}}$ (%) | $p$-value |
| Easy | 108 | 19.84 | 41.41 | < 0.0001 | 3.12 | 4.9 | 0.066 |
| Hard | 32 | 30.39 | 60.03 | < 0.0001 | 8.4 | 12.51 | 0.072 |
| Open | 30 | 42.31 | 60.09 | 0.0004 | 1226.25 | 1227.02 | 0.028 |
| Overall | 170 | 25.26 | 47.99 | < 0.0001 | 212.73 | 214.77 | 0.014 |

We use the performance profile technique from Dolan and Moré (2002) to further evaluate OSEA. The baselines for comparisons on problem $p \in P$ are set as the best MILP gap and computing time, respectively. Let $r_{p,X}^t$ denote the performance ratio of computing time on problem $p \in P$ using technique $X$,

$$r_{p,X}^t = \frac{t_X^p}{\min\{t_{\text{OSEA}}^p, t_{\text{standard}}^p\}}, \quad \forall X \in \{\text{OSEA, standard}\}.$$

Let $\rho_X^t(\tau)$ denote the probability for approach $X$ that $r_{p,X}^t$ is within a factor $\tau$ of the best ratio in terms of computing time,

$$\rho_X^t(\tau) = \frac{1}{|P|}\text{size}\{p \in P : r_{p,X}^t \leq \tau\}, \ \forall X \in \{\text{OSEA, standard}\}.$$

Similarly, let $\rho_X^G(\tau)$ denote the probability that approach $X$ is within a factor of $\tau$ from the best MILP gap ratio. In both cases, larger values are preferred. The cumulative distributions of $\rho_X^t(\tau)$ and $\rho_X^G(\tau)$ form the respective performance profiles. Performance profiles evaluate the overall performance of a solution technique and when

$|P|$ is sufficiently large, are relatively robust with respect to performance outliers of individual problem instances (Dolan and Moré 2002).

The performance profile for solution times are presented in Figs. 3 and 4 with two different ranges for $\tau$ each. The probability that OSEA terminates earlier than the standard approach is 0.829 (see Fig. 3 when $\tau = 1$). OSEA solves 100% of the problems within a factor of 6.1 for the computation time ratio, i.e., $\rho^t_{\text{OSEA}}(6.1) = 1$, whereas $\rho^t_{\text{standard}}(6.1)$ is only 0.606. The performance profile for the standard approach demonstrates that OSEA is much faster for many problems, e.g., $\rho^t_{\text{standard}}(\tau) \leq 0.9$ for $\tau \leq 872$ (depicted in Fig. 4.) That is, OSEA terminates 872 times faster than the standard approach on 10% of benchmark problems.

While the reduced problems solve faster than the original problems, the solution qualities must be examined. Figure 5 shows the performance profile of solution qual-



**Fig. 3** Computation time performance profile of the probability each method ($\rho^t_X$) is within factor ($1 \leq \tau \leq 10$) from the best MILP gap ratio
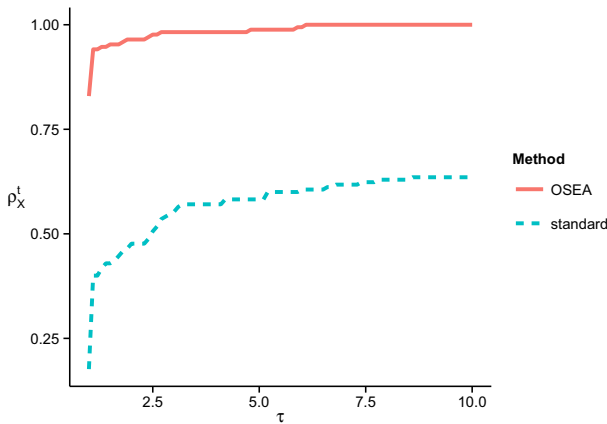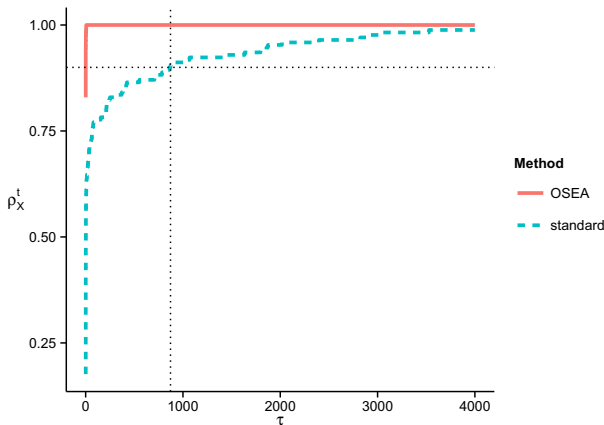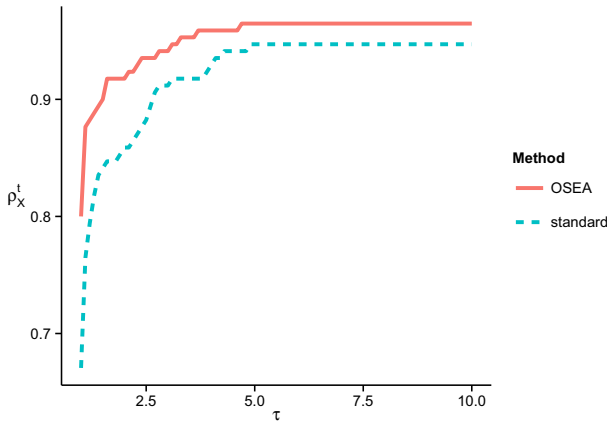


**Fig. 4** Computation time performance profile of the probability each method ($\rho^t_X$) is within factor ($1 \leq \tau \leq 4000$) from the best MILP gap ratio

**Fig. 5** Solution quality performance profile of the probability each method ($\rho_X^t$) is within factor ( $1 \leq \tau \leq$ 10) from the best MILP gap ratio

ity for $\tau \in [1, 10]$. OSEA is more likely to outperform the standard method, i.e., $\rho_{\text{OSEA}}^G(1) = 80\% > \rho_{\text{standard}}^G(1) = 67\%$. OSEA solves 90% of all problems within a factor of 1.5 of the best technique. The OSEA solution quality performance profile is equal to or superior than the standard performance profile across all values of $\tau$. Note the minimum values of $\tau$ necessary to capture all problems: $\rho_{\text{OSEA}}^G(86) = 100\%$, whereas $\rho_{\text{standard}}^G(1173) = 100\%$ (not shown in figure) implying that OSEA performs, at worst, 86 times as bad as the standard technique, whereas the standard technique performed, up to 1173 times worse than OSEA.

## 4 Conclusions

The objective scaling ensemble approach is a novel, two-phase heuristic solution procedure that iteratively solves scaled linear versions of the original MILP problem and uses a subset of the LP relaxation results to form an ensemble of solutions. This ensemble is aggregated in such a way to identify integer variables which are not likely to be used in an optimal solution. These variables are removed from the MILP to create a reduced problem space. Exact techniques such as branch-and-cut are applied to the revised problem formulation. If the reduced search space is sufficiently large, a feasible and even possibly optimal solution for the original MILP can be found. If the space is small enough, the revised problem space can be searched more efficiently.

The inspiration for OSEA comes from well known and successful heuristic approaches which have been used in conjunction with other techniques to produce a more efficient search of complex problem spaces. Many advanced heuristic approaches are often invoked by default in commercial optimization software. We compare the solution quality of OSEA in the first 60 s of optimization time to that of the assortment of heuristics, cutting plane strategies, and exact search algorithms implemented in Gurobi 5.6.3. OSEA successfully reduces the search space in a way which is competitive with industry leading optimization software.

The empirical results on 170 publicly available benchmark integer programming problems and rigorous analysis indicate that OSEA can improve MILP solution quality on a wide range of problems without compromising the computation time. Among the benchmark problems, many are well documented and related to published work (e.g., Fischetti et al. 2005; Bley et al. 2010; Raack et al. 2011). The instances include a wide variety of problem types and application areas including network design, open pit mine production, the p-Median problem, crew scheduling, and lot sizing, among others. The problems range in size from hundreds of integer variables to several orders of magnitude more. For certain problem types evaluated, OSEA performs exceptionally well, e.g. open pit mining. For others, the results while promising, are mixed, e.g. network design problems. In future work we will examine the particular nature of certain problem formulations to understand whether or not the outstanding results are generalizable to the problem class. In addition, the common characteristics of problems that OSEA has stable good performance will be investigated to provide a general guideline to use OSEA . Finally, the improvement of OSEA on the problems that it perform bad in this work will be investigated, for example, integrating with other principled heuristics.

OSEA can be easily applied to any MILP problem with integer variables in the objective function. However, OSEA is not meant to be used exclusive of traditional IP solvers, but ideally to be incorporated as yet another of the integrated heuristics used in software. In our initial experimentation to this end, we find OSEA to improve the solution performance at the root node of the branch-and-bound algorithm, but not to work well at subsequent nodes. While some heuristics (e.g., RINS) are activated at various nodes of branch-and-bound, earlier indications are that OSEA is a beneficial initial heuristic applied specifically at the root node to find better incumbent solutions early on.

# Appendices

## A. Experimental results of *Easy* instances

| Name | $\gamma$ (%) | $G_{OSEA}$ (%) | $T_{OSEA}$ | $G_{Standard}$ (%) | $T_{Standard}$ |
|---|---|---|---|---|---|
| 30_70_45_095_100 | 11.82 | 0.00 | 2.21 | 0.00 | 2.79 |
| 30n20b8 | 99.09 | 0.00 | 0.38 | 0.00 | 2.58 |
| 50v-10 | 88.78 | 0.12 | 0.92 | 0.32 | 60.00 |
| aflow40b | 87.20 | 0.00 | 0.94 | 0.00 | 60.00 |
| air04 | 94.12 | 0.00 | 11.34 | 0.00 | 9.22 |
| app1-2 | 90.18 | 0.00 | 1.34 | 0.00 | 47.06 |
| beasleyC3 | 64.36 | 0.00 | 0.24 | 0.00 | 10.53 |
| berlin_5_8_0 | 64.23 | 0.00 | 0.03 | 0.00 | 60.00 |
| biella1 | 93.09 | 1.41 | 11.60 | 5.87 | 60.00 |
| binkar10_1 | 42.65 | 0.00 | 0.40 | 0.00 | 8.32 |
| co-100 | 96.65 | 0.01 | 25.07 | 45.55 | 60.02 |
| core2536-691 | 90.62 | 0.00 | 16.56 | 0.00 | 31.48 |
| cov1075 | 0.00 | 0.00 | 5.97 | 0.00 | 3.32 |
| dfn-gwin-UUM | 58.95 | 0.00 | 2.00 | 0.00 | 60.00 |
| eil33-2 | 97.87 | 5.43 | 5.56 | 0.00 | 36.46 |
| eilB101 | 90.74 | 3.13 | 2.27 | 3.02 | 60.00 |
| enlight13 | 35.80 | 0.00 | 0.01 | 0.00 | 3.44 |
| enlight15 | 53.78 | 0.00 | 0.01 | 0.00 | 14.42 |
| gmu-35-40 | 92.92 | 0.02 | 0.02 | 0.03 | 60.00 |
| gmu-35-50 | 95.98 | 0.01 | 0.03 | 0.03 | 60.00 |
| go19 | 16.99 | 0.00 | 60.00 | 1.18 | 60.00 |
| harp2 | 94.85 | 0.00 | 0.11 | 0.00 | 25.31 |
| ic97_potential | 2.63 | 0.18 | 60.00 | 0.08 | 60.00 |
| iis-100-0-cov | 0.00 | 0.00 | 60.00 | 0.00 | 60.01 |
| iis-bupa-cov | 14.97 | 0.00 | 60.00 | 0.00 | 60.00 |
| iis-pima-cov | 55.14 | 2.94 | 60.00 | 2.94 | 60.00 |
| k16x240 | 87.95 | 0.00 | 0.02 | 0.00 | 60.00 |
| lectsched-4-obj | 43.26 | 0.00 | 0.21 | 0.00 | 0.65 |
| m100n500k4r1 | 95.20 | 4.17 | 0.01 | 4.17 | 60.00 |
| macrophage | 53.41 | 0.27 | 0.06 | 0.00 | 60.00 |
| mc11 | 48.53 | 0.00 | 1.33 | 0.00 | 9.79 |
| mine-166-5 | 41.08 | 0.00 | 0.30 | 0.00 | 19.57 |
| mine-90-10 | 40.11 | 0.07 | 0.14 | 0.07 | 60.00 |
| mzzv11 | 96.57 | 3.91 | 0.24 | 0.00 | 14.93 |
| n3div36 | 99.69 | 0.00 | 24.96 | 0.15 | 60.01 |
| n3seq24 | 98.91 | 2.25 | 33.61 | 2.61 | 60.04 |
| n4-3 | 24.14 | 0.85 | 60.00 | 0.19 | 60.00 |
| n9-3 | 43.65 | 3.17 | 60.00 | 9.84 | 60.00 |
| neos-1109824 | 91.78 | 0.00 | 2.08 | 0.00 | 26.80 |
| neos-1112782 | 88.94 | 0.00 | 0.15 | 0.00 | 8.67 |
| neos-1112787 | 89.29 | 0.00 | 0.13 | 0.00 | 1.83 |
| neos-1224597 | 66.51 | 0.00 | 0.25 | 0.00 | 0.25 |
| neos-1225589 | 81.48 | 0.00 | 0.08 | 0.00 | 0.14 |
| neos-1337307 | 59.33 | 0.00 | 24.75 | 0.00 | 25.67 |
| neos-1396125 | 10.49 | 0.00 | 25.63 | 0.00 | 10.39 |

| Name | $\gamma$ | $G_{OSEA}$ | $T_{OSEA}$ | $G_{Standard}$ | $T_{Standard}$ |
|---|---|---|---|---|---|
| neos-1440225 | 67.21 | 0.00 | 0.76 | 0.00 | 23.18 |
| neos-1616732 | 0.00 | 0.62 | 60.00 | 0.62 | 60.00 |
| neos-1620770 | 0.00 | 0.00 | 60.00 | 0.00 | 60.01 |
| neos-506428 | 94.65 | 77.78 | 7.77 | 77.78 | 60.01 |
| neos-520729 | 99.44 | 0.00 | 14.53 | 0.00 | 60.02 |
| neos-555424 | 49.78 | 0.00 | 0.31 | 0.00 | 5.06 |
| neos-631710 | 92.67 | 46.30 | 32.32 | 63.49 | 60.03 |
| neos-686190 | 90.61 | 0.59 | 2.46 | 10.98 | 60.00 |
| neos-693347 | 49.96 | 1.68 | 60.00 | 1.27 | 60.05 |
| neos-777800 | 94.89 | 0.00 | 7.04 | 0.00 | 1.17 |
| neos-824661 | 92.85 | 0.00 | 3.73 | 0.00 | 9.13 |
| neos-824695 | 91.61 | 0.00 | 1.09 | 0.00 | 2.70 |
| neos-826650 | 85.79 | 0.00 | 0.25 | 0.00 | 60.00 |
| neos-826694 | 86.73 | 0.00 | 1.02 | 0.00 | 1.97 |
| neos-826812 | 86.81 | 0.00 | 0.64 | 0.00 | 0.87 |
| neos-826841 | 84.78 | 0.00 | 0.26 | 0.00 | 40.41 |
| neos-885524 | 99.78 | 0.00 | 1.55 | 0.00 | 0.58 |
| neos-932816 | 93.66 | 0.00 | 0.88 | 0.00 | 0.91 |
| neos-933638 | 83.64 | 0.36 | 60.00 | 0.00 | 33.14 |
| neos-933966 | 89.00 | 0.00 | 31.99 | 0.00 | 6.68 |
| neos-934278 | 77.74 | 8.45 | 60.00 | 0.00 | 42.49 |
| neos-935627 | 47.56 | 10.51 | 60.00 | 0.19 | 60.00 |
| neos-935769 | 41.23 | 1.95 | 60.00 | 0.00 | 25.78 |
| neos-937511 | 58.84 | 0.00 | 60.00 | 0.00 | 10.30 |
| neos-941262 | 46.16 | 1.10 | 60.02 | 1.10 | 60.00 |
| neos-941313 | 79.80 | 0.00 | 32.07 | 0.00 | 39.15 |
| neos-957389 | 91.98 | 0.00 | 0.39 | 0.00 | 1.16 |
| neos15 | 0.00 | 0.90 | 60.00 | 0.90 | 60.00 |
| neos16 | 32.10 | 0.22 | 60.00 | 0.22 | 60.00 |
| neos18 | 4.83 | 0.00 | 0.44 | 0.00 | 41.12 |
| net12 | 58.07 | 0.00 | 0.75 | 16.08 | 60.00 |
| nobel-eu-DBE | 93.91 | 2.05 | 45.11 | 0.82 | 60.00 |
| noswot | 78.00 | 0.00 | 0.01 | 0.00 | 42.38 |
| nu60-pr9 | 90.84 | 1.58 | 0.78 | 3.18 | 60.35 |
| p80x400b | 76.01 | 0.17 | 0.07 | 0.00 | 60.00 |
| pg | 0.00 | 0.25 | 60.00 | 0.25 | 60.00 |
| pg5_34 | 0.00 | 0.03 | 60.00 | 0.03 | 60.00 |
| pigeon-10 | 82.00 | 0.00 | 0.04 | 0.00 | 60.00 |
| pigeon-11 | 82.03 | 0.00 | 0.02 | 0.00 | 60.00 |
| pw-myciel4 | 33.24 | 0.00 | 60.00 | 0.00 | 60.00 |
| pw-myciel4 | 33.24 | 0.00 | 60.00 | 0.00 | 60.00 |
| rail507 | 98.24 | 0.00 | 48.07 | 0.00 | 60.01 |
| ran14x18 | 74.42 | 0.64 | 0.16 | 0.64 | 60.00 |
| ran16x16 | 72.83 | 0.10 | 0.06 | 0.00 | 48.51 |
| reblock166 | 45.54 | 0.03 | 0.83 | 0.06 | 60.00 |
| reblock67 | 38.36 | 0.21 | 0.06 | 0.59 | 60.00 |
| rmatr100-p10 | 0.00 | 0.00 | 28.41 | 0.00 | 47.48 |
| rmatr100-p5 | 0.00 | 6.69 | 60.00 | 6.96 | 60.00 |
| rmine6 | 31.66 | 0.01 | 0.24 | 0.00 | 60.00 |
| rococoB10-011000 | 87.43 | 2.54 | 0.15 | 7.00 | 60.00 |
| rococoC10-001000 | 85.95 | 0.04 | 0.08 | 0.50 | 60.00 |
| satellites1-25 | 85.80 | 0.00 | 20.80 | 117.24 | 60.00 |

| Name | $\gamma$ | $G_{OSEA}$ | $T_{OSEA}$ | $G_{Standard}$ | $T_{Standard}$ |
|---|---|---|---|---|---|
| satellites2-60-fs | 82.49 | 144.19 | 37.60 | 139.58 | 60.50 |
| sp97ar | 97.61 | 0.26 | 22.35 | 0.88 | 60.13 |
| sp98ic | 98.70 | 0.19 | 25.07 | 0.65 | 60.00 |
| sp98ir | 92.81 | 0.13 | 1.40 | 0.00 | 30.46 |
| tanglegram1 | 84.22 | 0.00 | 0.87 | 0.00 | 4.48 |
| tanglegram2 | 87.36 | 0.00 | 0.11 | 0.00 | 0.55 |
| toll-like | 55.22 | 0.33 | 0.08 | 1.61 | 60.00 |
| uct-subprob | 5.85 | 1.57 | 60.00 | 0.95 | 60.37 |
| umts | 85.21 | 0.12 | 1.77 | 0.13 | 60.00 |
| wachplan | 86.88 | 0.00 | 6.99 | 0.00 | 60.00 |
| zib54-UUE | 0.00 | 2.23 | 60.00 | 2.23 | 60.00 |

## B. Experimental results of *Hard* instances

| Name | $\gamma$ (%) | $G_{OSEA}$ (%) | $T_{OSEA}$ | $G_{Standard}$ (%) | $T_{Standard}$ |
|---|---|---|---|---|---|
| a1c1s1 | 9.90 | 0.26 | 60.00 | 0.02 | 60.00 |
| b2c1s1 | 3.91 | 5.84 | 60.00 | 6.75 | 60.03 |
| bg512142 | 5.67 | 5.44 | 60.00 | 8.75 | 60.00 |
| dg012142 | 7.68 | 20.70 | 60.00 | 25.81 | 60.00 |
| dolom1 | 81.32 | 92.38 | 60.00 | 97.80 | 60.02 |
| germany50-DBM | 1.33 | 2.13 | 60.00 | 1.46 | 60.00 |
| d10200 | 79.27 | 0.10 | 23.79 | 0.10 | 60.00 |
| dc1c | 95.24 | 5.95 | 19.50 | 91.22 | 60.00 |
| janos-us-DDM | 25.04 | 0.05 | 60.00 | 0.08 | 60.00 |
| lotsize | 44.77 | 0.47 | 60.00 | 1.22 | 60.00 |
| eilA101-2 | 99.54 | 7.68 | 23.50 | 7.68 | 60.14 |
| n3-3 | 46.72 | 3.71 | 60.00 | 10.18 | 60.00 |
| neos-948126 | 33.25 | 4.22 | 60.00 | 4.22 | 60.01 |
| neos-984165 | 39.81 | 23.97 | 60.00 | 23.97 | 60.01 |
| mkc | 96.67 | 0.02 | 0.11 | 0.00 | 60.00 |
| rmatr200-p10 | 24.30 | 2.04 | 60.00 | 10.36 | 60.01 |
| nu120-pr3 | 92.10 | 3.99 | 1.12 | 4.38 | 60.43 |
| p100x588b | 75.65 | 0.65 | 0.29 | 1.71 | 60.00 |
| p6b | 86.36 | 0.00 | 0.02 | 0.00 | 60.00 |
| pigeon-12 | 82.07 | 0.00 | 0.03 | 0.00 | 60.00 |
| pigeon-13 | 82.10 | 0.00 | 0.03 | 0.00 | 60.00 |
| protfold | 83.98 | 34.78 | 0.07 | 40.91 | 60.00 |
| queens-30 | 95.89 | 8.11 | 0.03 | 8.11 | 60.07 |
| r80x800 | 84.42 | 0.06 | 0.30 | 0.06 | 60.00 |
| reblock354 | 39.29 | 0.01 | 0.28 | 0.04 | 60.02 |
| reblock420 | 45.93 | 15.28 | 10.03 | 15.68 | 60.06 |
| rmatr200-p20 | 20.87 | 0.12 | 60.01 | 4.67 | 60.00 |
| rmatr200-p5 | 18.83 | 6.84 | 60.01 | 9.24 | 60.01 |
| seymour | 29.68 | 0.24 | 60.00 | 0.24 | 60.10 |
| rococoC11-011100 | 90.29 | 3.14 | 0.18 | 6.97 | 60.15 |
| tw-myciel4 | 38.03 | 0.00 | 0.14 | 0.00 | 60.00 |
| wnq-n100-mw99-14 | 90.58 | 14.24 | 22.73 | 9.44 | 60.03 |

## C. Experimental results of *Open* instances

| Name | $\gamma$ (%) | $G_{OSEA}$ (%) | $T_{OSEA}$ | $G_{Standard}$ (%) | $T_{Standard}$ |
|---|---|---|---|---|---|
| core4872-1529 | 67.32 | 3.96 | 60.00 | 3.59 | 60.01 |
| bab1 | 98.70 | 31.16 | 22.35 | 31.16 | 60.01 |
| cdma | 83.99 | 32138.99 | 20.17 | 32138.99 | 60.01 |
| dc1l | 95.68 | 8.00 | 60.06 | 10.30 | 61.98 |
| d20200 | 85.20 | 0.30 | 5.37 | 0.31 | 60.00 |
| ex1010-pi | 86.93 | 9.93 | 60.00 | 11.02 | 60.00 |
| ger50_17_trans | 97.54 | 10.82 | 60.00 | 15.56 | 60.00 |
| momentum3 | 68.69 | 76.50 | 60.02 | 76.50 | 60.03 |
| methanosarcina | 56.53 | 100.00 | 0.28 | 100.00 | 60.00 |
| n3700 | 92.85 | 23.33 | 60.00 | 24.64 | 60.00 |
| n3705 | 92.32 | 24.13 | 60.00 | 24.22 | 60.00 |
| n370a | 91.65 | 23.71 | 60.00 | 25.09 | 60.00 |
| neos-937815 | 48.89 | 0.56 | 60.00 | 0.56 | 60.00 |
| ns4-pr3 | 0.99 | 0.20 | 60.00 | 0.19 | 60.00 |
| ns4-pr9 | 0.00 | 0.16 | 60.00 | 0.16 | 60.00 |
| pigeon-19 | 82.33 | 5.56 | 0.07 | 5.56 | 60.00 |
| ramos3 | 18.69 | 44.77 | 60.13 | 48.30 | 60.11 |
| rmine10 | 34.92 | 2.45 | 14.60 | 2.46 | 60.01 |
| rmine14 | 96.73 | 2426.06 | 1.54 | 2426.06 | 60.08 |
| rococoC12-111000 | 93.73 | 25.93 | 0.28 | 25.54 | 60.00 |
| rvb-sub | 99.39 | 94.70 | 22.05 | 93.47 | 60.01 |
| siena1 | 79.86 | 92.48 | 60.00 | 97.73 | 60.12 |
| sing2 | 47.56 | 2.30 | 60.00 | 9.22 | 60.01 |
| stockholm | 55.71 | 99.50 | 60.01 | 99.49 | 60.05 |
| sts405 | 0.00 | 60.98 | 60.04 | 60.98 | 60.01 |
| sts729 | 0.00 | 62.44 | 60.01 | 62.44 | 60.12 |
| t1717 | 97.80 | 43.80 | 60.02 | 43.80 | 60.01 |
| t1722 | 97.21 | 35.94 | 60.00 | 32.35 | 60.00 |
| usAbbrv.8.25_70 | 69.84 | 21.49 | 0.08 | 22.13 | 60.00 |
| van | 0.00 | 95.17 | 60.01 | 95.17 | 60.01 |

# References

Achterberg, T., Berthold, T.: Improving the feasibility pump. Discrete Optim. **4**, 77–86 (2007)

Bai, L., Hearn, D., Lawphongpanich, S.: A heuristic method for the minimum toll booth problem. J. Glob. Optim. **48**, 533–548 (2010)

Balas, E., Ceria, S., Dawande, M., Margot, F., Pataki, G.: Octane: a new heuristic for pure 0–1 programs. Oper. Res. **49**, 207–225 (2001)

Balas, E., Martin, C.: Pivot and complement-a heuristic for 0–1 programming. Manag. Sci. **26**, 86–96 (1980)

Balas, E., Schmieta, S., Wallace, C.: Pivot and shift-a mixed integer programming heuristic. Discrete Optim. **1**, 3–12 (2004)

Beliën, J.: Exact and heuristic methodologies for scheduling in hospitals: problems, formulations and algorithms. 4OR **5**, 157–160 (2007)

Bertsimas, D., Sim, M.: Robust discrete optimization and network flows. Math. Program. **98**, 49–71 (2003)

Bley, A., Boland, N., Fricke, C., Froyland, G.: A strengthened formulation and cutting planes for the open pit mine production scheduling problem. Comput. Oper. Res. **37**, 1641–1647 (2010)

Blum, C., Roli, A.: Metaheuristics in combinatorial optimization: overview and conceptual comparison. ACM Comput. Surv. (CSUR) **35**, 268–308 (2003)

Bonami, P., Cornuéjols, G., Lodi, A., Margot, F.: A feasibility pump for mixed integer nonlinear programs. Math. Program. **119**, 331–352 (2009)

Breiman, L.: Bagging predictors. Mach. Learn. **26**, 123–140 (1996)

Climer, Sharlee, Zhang, Weixiong: Cut-and-solve: an iterative search strategy for combinatorial optimization problems. Artif. Intell. **170**, 714–738 (2006)

Crainic, T., Gendreau, M., Fravolden, J.: A simplex-based tabu search method for capacitated network design. INFORMS J. Comput. **12**, 223–236 (2000)

Crainic, T., Gendron, B., Hernu, G.: A slope scaling/Lagrangian perturbation heuristic with long-term memory for multicommodity capacitated fixed-charge network design. J. Heuristics **10**, 525–545 (2004)

Danna, E., Rothberg, E., Le Pape, C.: Exploring relaxation induced neighborhoods to improve mip solutions. Math. Program. **102**, 71–90 (2005)

Dolan, E., Moré, J.: Benchmarking optimization software with performance profiles. Math. Program. **91**, 201–213 (2002)

Eckstein, J., Nediak, M.: Pivot, cut, and dive: a heuristic for 0–1 mixed integer programming. J. Heuristics **13**, 471–503 (2007)

Fischetti, M., Glover, F., Lodi, A.: The feasibility pump. Math. Program. **104**, 91–104 (2005)

Fischetti, M., Lodi, A.: Local branching. Math. Program. **98**, 23–47 (2003)

Gendreau, M., Potvin, J.-Y. (eds.): Handbook of Metaheuristics, 2nd ed. Springer, New York (2010). https://doi.org/10.1007/978-1-4419-1665-5

Gendron, B., Potvin, J.-Y., Soriano, P.: A tabu search with slope scaling for the multicommodity capacitated location problem with balancing requirements. Ann. Oper. Res. **122**, 193–217 (2003)

Glover, F., Laguna, M.: Tabu Search. Springer (1999)

Gulczynski, D., Golden, B., Wasil, E.: The multi-depot split delivery vehicle routing problem: an integer programming-based heuristic, new test problems, and computational results. Comput. Ind. Eng. **61**, 794–804 (2011)

Gurobi Optimization, Inc. (2012) Gurobi solves the hardest models. http://www.gurobi.com/company/news/gurobi-solves-the-previously-unsolvable. Accessed 20 Nov 2015

Gurobi Optimization, Inc. (2014) Features and benefits overview. http://www.gurobi.com/products/gurobi-optimizer/features-and-benefits. Accessed 31 Dec 2014

Hoffman, K.L., Padberg, M.: Solving airline crew scheduling problems by branch-and-cut. Manag. Sci. **39**, 657–682 (1993)

Kim, D., Pardalos, P.: A solution approach to the fixed charge network flow problem using a dynamic slope scaling procedure. Oper. Res. Lett. **24**, 195–203 (1999)

Kim, D., Pardalos, P.: Dynamic slope scaling and trust interval techniques for solving concave piecewise linear network flow problems. Networks **35**, 216–222 (2000)

Kirkpatrick, S., Gelatt, C., Vecchi, M.: Optimization by simulated annealing. Science **220**, 671–680 (1983)

Koch, T., Achterberg, T., Andersen, E., Bastert, O., Berthold, T., Bixby, R., Danna, E., Gamrath, G., Gleixner, A., Heinz, S., et al.: Miplib 2010. Math. Program. Comput. **3**, 103–163 (2011)

Linderoth, J., Lodi, A.: MILP software. In: Cochran, J., Cox, L., Keskinocak, P., Kharoufeh, J., Smith, J. (eds.) Wiley Encyclopedia of Operations Research and Management Science. Wiley, New York (2010)

Lodi, A.: Mixed integer programming computation. In: Jünger, M., Liebling, T., Naddef, D., Nemhauser, G., Pulleyblank, W., Reinelt, G., Rinaldi, G., Wolsey, L. (eds.) 50 Years of Integer Programming 1958–2008, pp. 619–645. Springer, Berlin (2010)

Nahapetyan, A., Pardalos, P.: Adaptive dynamic cost updating procedure for solving fixed charge network flow problems. Comput. Optim. Appl. **39**, 37–50 (2008)

Nemhauser, G., Wolsey, L.: Integer and Combinatorial Optimization, vol. 18. Wiley, New York (1988)

Nicholson, C., Zhang, W.: Optimal network flow: a predictive analytics perspective on the fixed-charge network flow problem. Comput. Ind. Eng. **99**, 260–268 (2016)

Patel, J., Chinneck, J.: Active-constraint variable ordering for faster feasibility of mixed integer linear programs. Math. Program. **110**, 445–474 (2007)

Raack, C., Koster, A., Orlowski, S., Wessäly, R.: On cut-based inequalities for capacitated network design polyhedra. Networks **57**, 141–156 (2011)

Shiina, T., Xu, C.: Dynamic slope scaling procedure to solve stochastic integer programming problem. J. Comput. Model. **2**, 133–148 (2012)

van den Akker, J., Hurkens, C., Savelsbergh, M.: Time-indexed formulations for machine scheduling problems: column generation. INFORMS J. Comput. **12**, 111–124 (2000)

Zhang, W., Lin, Pi, Wang, N., Nicholson, C., Xue, X.: Probabilistic prediction of postdisaster functionality loss of community building portfolios considering utility disruptions. J. Struct. Eng. **144**, 04018015 (2018)

Zhang, W., Nicholson, C.: A multi-objective optimization model for retrofit strategies to mitigate direct economic loss and population dislocation. Sustain. Resilient Infrastruct. **1**, 123–136 (2016a)

Zhang, W., Nicholson, C.: Prediction-based relaxation solution approach for the fixed charge network flow problem. Comput. Ind. Eng. **99**, 106–111 (2016b)

Zhang, W., Wang, N.: Resilience-based risk mitigation for road networks. Struct. Saf. **62**, 57–65 (2016)

Zhang, W., Wang, N.: Bridge network maintenance prioritization under budget constraint. Struct. Saf. **67**, 96–104 (2017)

Zhang, W., Wang, N., Nicholson, C.: Resilience-based post-disaster recovery strategies for road-bridge networks. Struct. Infrastruct. Eng. **13**, 1404–1413 (2017)

Zhang, W., Yao, Z.: A reformed lattice gas model and its application in the simulation of evacuation in hospital fire. In: Industrial Engineering and Engineering Management (IEEM), 2010 IEEE International Conference on. IEEE, pp. 1543–1547 (2010)

Zhang, W., Zhao, L.: Lattice gas model for simulating pedestrian evacuation in the dormitory fire. J. Saf. Environ. **1**, 045 (2010)