

WILEY

INTERNATIONAL  
TRANSACTIONS  
IN OPERATIONAL  
RESEARCHIntl. Trans. in Op. Res. 0 (2023) 1–27  
DOI: 10.1111/itor.13301

## Matheuristics: survey and synthesis

Marco A. Boschetti<sup>a</sup>, Adam N. Letchford<sup>b</sup> and Vittorio Maniezzo<sup>c,\*</sup> <sup>a</sup>Department of Mathematics, University of Bologna, via dell'Università 50, Cesena 47521, Italy<sup>b</sup>Department of Management Science, Lancaster University, Lancaster LA1 4YX, UK<sup>c</sup>Department of Computer Science and Engineering, University of Bologna, via dell'Università 50, Cesena 47521, Italy

E-mail: marco.boschetti@unibo.it [Boschetti]; a.n.letchford@lancaster.ac.uk [Letchford];

vittorio.maniezzo@unibo.it [Maniezzo]

Received 3 March 2023; received in revised form 21 March 2023; accepted 22 March 2023

**Abstract**

In integer programming and combinatorial optimisation, people use the term *matheuristics* to refer to methods that are heuristic in nature but draw on concepts from the literature on exact methods. We survey the literature on this topic, with a particular emphasis on matheuristics that yield both primal and dual bounds (i.e., upper and lower bounds in the case of a minimisation problem). We also make some comments about possible future developments.

*Keywords:* integer programming; combinatorial optimisation; heuristics; matheuristics; Lagrangian relaxation; dual ascent

**1. Introduction**

Ever since the 1960s, integer programming and combinatorial optimisation problems have received much attention from mathematicians, computer scientists and operational researchers, due to the huge range of important practical applications (see, e.g., Wolsey, 1998; Conforti et al., 2014; Korte and Vygen, 2018). Unfortunately, many problems of interest are NP-hard (see Ausiello et al., 1999), which means that large-scale instances can be very challenging to solve.

In this context, a key distinction is between *exact* and *heuristic* methods. Exact methods are guaranteed to solve instances to proven optimality, given enough computing resources (i.e., time and memory). Heuristics are not guaranteed to find optimal solutions, but they tend to be faster, and they often yield solutions which are of ‘acceptable’ quality. Although significant progress has been made in exact methods (e.g., Wolsey, 1998; Conforti et al., 2014), heuristics remain extremely useful in many cases (e.g., Martí et al., 2018; Potvin and Gendreau, 2019).

\*Corresponding author.

© 2023 The Authors.

*International Transactions in Operational Research* published by John Wiley & Sons Ltd on behalf of International Federation of Operational Research Societies

This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

Many authors use the term *metaheuristics* to refer to general-purpose heuristic frameworks, such as simulated annealing and tabu search (see the textbooks: Martí et al., 2018; Potvin and Gendreau, 2019). More recently, researchers started using the term *matheuristics* to refer to (meta-)heuristics that draw on concepts from the traditional mathematical programming literature (see Maniezzo et al., 2010; Stützle and Maniezzo, 2020). Such heuristics may have subroutines that involve, for example, linear programming, integer programming, dynamic programming (DP), Lagrangian relaxation (LR) or Benders decomposition (BD).

The first international matheuristics conference took place in Bertinoro, Italy in 2006 (see Boschetti and Maniezzo, 2022). Ever since then, the field of matheuristics has been in rapid expansion. For details, we refer the reader to the surveys (Raidl and Puchinger, 2008; Ball, 2011; Raidl, 2015; Fischetti and Fischetti, 2016; Boschetti and Maniezzo, 2022) and the books (Maniezzo et al., 2010, 2021).

In this new survey, we approach the topic from a different viewpoint, placing a particular emphasis on matheuristics that yield both *primal* and *dual* bounds. For a minimisation problem, such matheuristics yield not only a feasible integer solution (with an associated upper bound) but also the solution to some kind of relaxed problem or dual problem (with an associated lower bound). In our view, matheuristics of this kind are highly desirable, since the bounds help one to evaluate the performance of the method with higher precision, at any given instance.

The paper has the following structure. Section 2 covers matheuristics that are based on *linear programming* (LP) relaxation and/or duality. Section 3 deals with ones based on *LR*, *surrogate relaxation* (SR) or closely related methods. Section 4 concerns heuristics based on *decomposition* techniques, such as Dantzig–Wolfe or BD. Then, in Section 5, we mention a few popular matheuristics that do not necessarily yield dual bounds. Finally, in Section 6, we make some remarks about the strengths and weaknesses of each approach and suggest some topics for future research.

Throughout, we write ‘ILP’ for an *integer linear program* and ‘COP’ for a *combinatorial optimisation problem*. We assume that the reader is familiar with the idea of formulating COPs as ILPs, along with the basics of integer programming (see, e.g., Wolsey, 1998; Conforti et al., 2014). For ease of notation, we assume that an ILP with  $n$  variables and  $m$  constraints takes the form

$$\min \{c^T x : Ax \geq b, x \in \mathbb{Z}_+^n\}, \quad (1)$$

where  $c \in \mathbb{Q}^n$ ,  $A \in \mathbb{Q}^{m \times n}$  and  $b \in \mathbb{Q}^m$ . We use the convention that all vectors are column vectors. We sometimes also speak of *mixed-integer programs* (MIPs), by which we mean problems that are similar to ILPs, except that not all variables are required to take integer values.

## 2. Methods based on LP relaxation and duality

In this section, we survey matheuristics that are based on LP relaxation and/or duality. Subsection 2.1 recalls the key concepts needed, such as relaxation, primal and dual pairs, and reduced costs. Subsections 2.2 and 2.3 concern matheuristics that use only primal or reduced-cost information, respectively. Subsection 2.4 covers matheuristics that explicitly use dual information.

## 2.1. Recap on LP relaxation and duality

If we relax the integrality constraint in the ILP (1), we obtain the following problem:

$$\min \{c^T x : Ax \geq b, x \in \mathbb{R}_+^n\}, \quad (2)$$

This is an LP, which is typically much easier to solve. Trivially, the solution to the LP gives a *lower bound* for the original COP.

The following facts can be found in any textbook on LP (e.g., Vanderbei, 2020). The *dual* of the above LP is

$$\max \{b^T y : A^T y \leq c, y \in \mathbb{R}_+^m\}. \quad (3)$$

The original LP is called the *primal*. The strong duality theorem states that if  $x^*$  and  $y^*$  are optimal primal and dual solutions, respectively, then  $c^T x^* = b^T y^*$ . Moreover, the components of  $y^*$  are the *dual prices* for the primal constraints. The vectors  $s^* = Ax^* - b \in \mathbb{R}_+^m$  and  $\rho^* = c - A^T y^* \in \mathbb{R}_+^n$  are called the *surplus* vector and the *reduced cost* vector, respectively. The conditions  $(\rho^*)^T x^* = 0$  and  $(s^*)^T y^* = 0$ , called *complementary slackness*, always hold when  $x^*$  and  $y^*$  are optimal.

## 2.2. Heuristics that use primal information only

Among the many LP-based matheuristics, the easiest ones to understand are those that use primal information only. The intuition behind these methods is that an optimal (or near-optimal) LP solution  $x^*$  is likely to contain some information that could be exploited by a heuristic. For example, if all variables in the ILP are binary, one might hope that variables with  $x^*$ -value close to 1 will have a high probability of taking the value 1 in optimal (or near-optimal) solutions to the ILP.

One strand of the literature is concerned with the use of LP-based heuristics for specific COPs. Two good early examples are the ‘LP-rounding’ heuristic for the set covering problem, due to Hochbaum (1982), and the ‘randomised rounding’ heuristic for network design problems, due to Raghavan and Thompson (1987). These two heuristics are also interesting because they are *approximation algorithms*, which means that they are guaranteed to produce solutions whose cost is within a known factor of the optimum. Other notable examples include an ‘iterative rounding’ heuristic for shift scheduling, given in Thompson (1990), and a similar heuristic for lot-sizing with setups, in Maes et al. (1991). More recently, LP solutions have been used as seeds for standard metaheuristic search (Cacchiani et al., 2023).

A parallel strand of the literature is concerned with LP-based matheuristics for ILPs in general. A good early example, from 1969, is the ‘interior-path’ method of Hillier (1969). It starts at the LP optimum  $x^*$ , and then follows a path from  $x^*$  to the interior of the feasible region. As it goes along, an attempt is made to find ‘nearby’ integer solutions. For extensions to this method, see Faaland and Hillier (1979).

In 1980, Balas and Martin (1980) devised an LP-based heuristic for 0–1 LPs, that they called ‘pivot-and-complement’. The method starts at  $x^*$  and then performs a sequence of primal simplex pivots that attempt to ‘push’ fractional variables out of the basis. Later on, several researchers

suggested improving pivot-and-complement by adding a tabu search phase at the end (e.g., Aboudi and Jörnsten, 1994; Løkketangen and Glover, 1998).

In the coming paragraphs, we mention some of the matheuristics based on primal LP solutions that came out more recently.

*OCTANE.* ‘OCTANE’ is a matheuristic for pure 0–1 LPs, developed in 2001 by Balas et al. (2001). The basic idea is as follows. First, we define a polyhedron, called the  $n$ -dimensional octahedron, that circumscribes the unit hypercube and has one facet for every vertex of the hypercube. We start at a basic optimal LP solution  $x^*$ , and then move from it in some chosen direction. Eventually, we ‘hit’ a facet of the  $n$ -dimensional octahedron. We then check the corresponding vertex of the hypercube. If it is feasible for the original 0–1 LP, we have our desired heuristic solution. This procedure is repeated for several ‘promising’ directions.

Although OCTANE performed reasonably well, it does not seem to have received much further attention. This may be because some of the alternative matheuristics mentioned below have tended to perform even better.

*Relax-and-fix.* ‘Relax-and-fix’ is a primal heuristic for general MIPs that works by solving a sequence of simpler MIPs. To our knowledge, it was first defined explicitly in 1998 by Wolsey (1998). In its original format (Wolsey, 1998; Belvaux and Wolsey, 2000), it worked as follows. First, the set of integer variables is partitioned into subsets, say  $S_1, \dots, S_k$ . A simpler MIP is then solved, in which only the variables in  $S_1$  are the declared integer. The variables in  $S_1$  are then permanently fixed at the values that they take in the solution to the simpler MIP. The variables in  $S_2$  are then the declared integer, and the resulting MIP is solved, and so on.

This approach, while at times effective, limits the search to one pass through the list of predefined subsets. A simple generalization is to make the subset choice dynamically adaptive, but other extensions are possible.

*Diving heuristics.* Diving heuristics (Bixby et al., 2000) are a family of MIP heuristics that iteratively fix variables to integer values until a feasible MIP solution is obtained. They can be thought of as a heuristic for rapidly moving from a given node of a branch-and-bound tree to a ‘leaf’ node. The name comes from the fact that they ‘dive’ to a leaf node without any possibility of backtracking.

A well-known and highly effective diving heuristic is *relaxation-induced neighbourhood search* (RINS) (Danna et al., 2005). When applied at a given branch-and-bound node, RINS compares the fractional LP solution at the node with the incumbent MIP solution. Typically, the two solutions will differ in the values of some variables. RINS tries to force the two solutions to agree on all variables, by fixing all variables that have the same values and letting the solver try to solve optimally the residual MIP problem, called the *sub-MIP*. The sub-MIP can in fact be quite large if too few variables were fixed, so its solution could potentially take a time comparable to that of the original problem. To remedy this, a limit on the computational resources available for optimization is usually imposed.

Diving heuristics are also often used within *branch-and-price* algorithms (Sadykov et al., 2019). For more on branch-and-price, see Section 4.

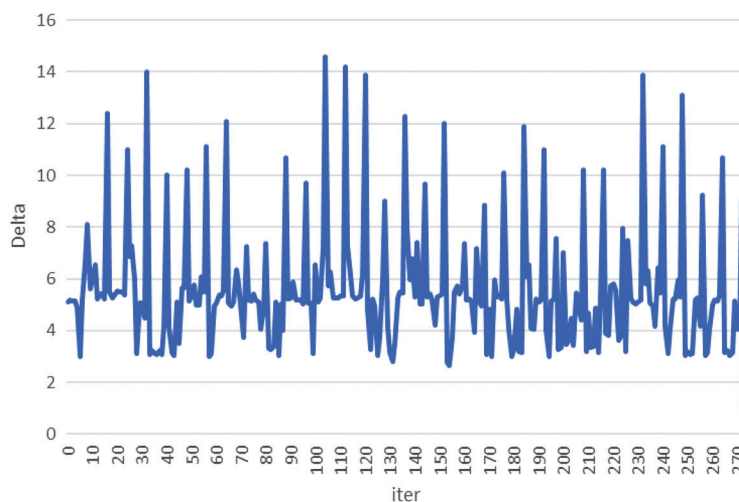


Fig. 1. Feasibility pump: delta evolution.

*The feasibility pump.* The feasibility pump (Fischetti et al., 2005) was initially conceived as a tool for finding initial feasible solutions to very challenging MIPs, and in this capacity it is often included in general-purpose MIP solvers. In the context of matheuristics, however, the feasibility pump can also be used to bring to integer feasibility a fractional and/or LP-infeasible solution. Such solutions arise not only in branch-and-bound but also in some other approaches, such as Lagrangian heuristics or destroy-and-fix approaches.

The feasibility pump builds on the observation that an integer feasible solution is coincident with its rounding. Formally, if  $P$  is the LP polytope of the problem under study, an integer feasible solution  $x^*$  corresponds to a point in  $P$  such that  $x^* = \tilde{x}$ , where  $\tilde{x}$  denotes the rounding of  $x^*$ , i.e., the possibly infeasible solution where each variable that must be integer is brought to the corresponding nearest integer.

The search for feasibility is based on the minimisation of a function measuring the distance between the two solutions,  $\Delta(x^*, \tilde{x})$ . The search starts from a point  $x^* \in P$  and its rounding  $\tilde{x}$ . If  $\tilde{x}$  is feasible, we have found a feasible integer solution and we stop the search. Otherwise, we start a *pumping cycle*. This means that we solve the linear problem  $\min\{\Delta(x, \tilde{x}) : x \in P\}$ . This yields a new point  $x^*$ , which we round to obtain a new point  $\tilde{x}$ . If  $\tilde{x}$  is feasible, we stop. Otherwise, we perform another pumping cycle, and so on.

Figure 1 shows the evolution of  $\Delta$  in the case of a simple ILP, which started from an optimal LP value of 231.45 to eventually converge to a final feasible solution of cost 333.

### 2.3. Heuristics that use reduced-cost information only

Now, recall from Subsection 2.1 the definition of the reduced-cost vector  $\rho^*$ . There are also matheuristics that rely only on  $\rho^*$ . The idea is that, if  $\rho_j^*$  is very large, the variable  $x_j$  is unlikely

to appear in an optimal solution to the ILP. Conversely, if  $\rho_j^*$  is zero or near-zero, there is a good chance that there is a (near-)optimal solution such that  $x_j$  takes a positive integer value.

The first paper we found which mentions this idea explicitly is Mansini and Speranza (1999), which is concerned with a portfolio selection problem. The problem is formulated as a Mixed Integer Linear Programming (MILP), and the LP relaxation is solved. In an attempt to obtain a good integer solution, a smaller MILP is solved, which contains all variables that have a positive value in the LP solution, plus the variables whose reduced cost is below some threshold. If time permits, a series of modified MILPs is solved, in which variables are added and dropped according to a heuristic rule.

The above heuristic inspired a general-purpose matheuristic for 0–1 LPs, called *kernel search* (Angelelli et al., 2010, 2012). Here is a brief overview of the approach. In the first phase, called the *initialisation* phase, the LP relaxation is solved, and a decision variable is called ‘promising’ if it has a zero (or near-zero) reduced cost. The set of all promising variables forms the *kernel*. The remaining variables are partitioned into sets, called *buckets*, according to the value of their reduced cost. One then solves a simplified version of the 0–1 LP, in which only the variables in the kernel are present. In most cases, this yields a good feasible solution, with an accompanying upper bound.

If the gap between the lower and upper bounds is acceptable, the process terminates. Otherwise, the variables in the first bucket are added to the 0–1 LP, along with a constraint stating that at least one of the variables in the bucket must take the value 1. The simplified ILP is then re-solved. If the resulting solution is better than the previous one, the variables that are in the bucket and take the value of 1 in the solution are added to the kernel. If the gap between the lower and upper bounds is now acceptable, the process terminates. Otherwise, the variables in the second bucket are added to the 0–1 LP, and so on.

Kernel search has been applied with great success to the multi-dimensional knapsack problem (Angelelli et al., 2010), a portfolio selection problem (Angelelli et al., 2012), the capacitated facility location problem (Guastaroba and Speranza, 2012) and an inventory routing problem (Archetti et al., 2021). It has also been extended to general MILPs (Guastaroba et al., 2017).

A variant of Kernel Search is the *incremental core* approach proposed in Boschetti et al. (2004, 2008). The idea is to select a parameter  $\rho^{\max}$  and include in the 0–1 LP only the variables with  $\rho_j^* \leq \rho^{\max}$ . If  $\rho^{\max}$  is equal to the gap between the upper and lower bounds, the 0–1 LP solution is optimal. Otherwise, it is a heuristic solution. In the latter case, one can increase  $\rho^{\max}$  in the hope of obtaining an improved heuristic solution (or even an optimal one).

Figure 2 shows a trace of an incremental core search on a resource-constrained project scheduling problem (from Mingozzi et al., 1998) based on an ‘additive’ lower bound (see Subsection 3.4). In this case, the problem could be solved to proven optimality. The plot shows the incremental contributions of the four successively computed bounds, together with the contribution of each new bound to the computation of improved feasible solutions.

## 2.4. Dual-based heuristics

When dealing with large-scale COP instances, even solving the initial LP relaxation (2) can be time consuming. Fortunately, with the help of LP duality, one can compute valid lower bounds without explicitly solving the initial LP. Indeed, if  $\bar{y}$  is any feasible solution to the dual LP (3), then  $b^T \bar{y}$  is

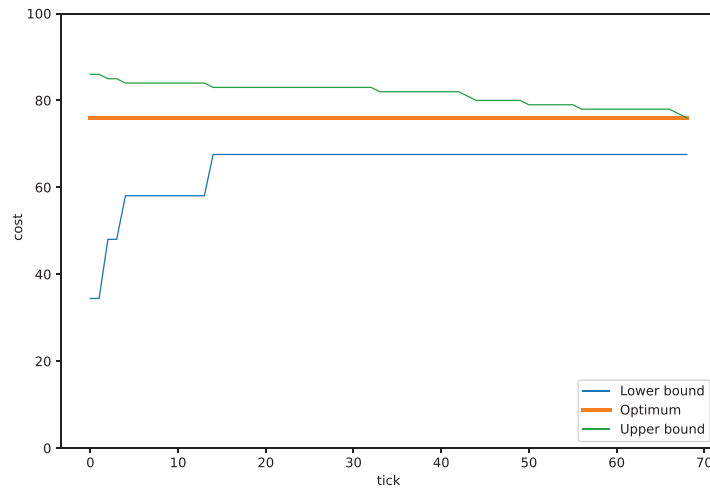


Fig. 2. Incremental core, upper and lower bound evolution.

a lower bound for the original LP, and therefore for the original COP. Thus, if we wish to obtain a lower bound quickly, we can solve the dual approximately using some kind of heuristic. Moreover, as we will see, the dual solution can then be used to drive a matheuristic.

This idea first appeared in Bilde and Krarup (1977) and Erlenkotter (1978), in the context of the *uncapacitated facility location problem* or UFLP. The authors of those papers proposed to start with all dual variables equal to zero and then iteratively increase individual dual values until no further increases are possible. Perhaps surprisingly, this approach gives quite good lower bounds for many UFLP instances.

Erlenkotter (1978) called this procedure *dual ascent*. He also proposed a simple local search procedure, called *dual adjustment*, which attempts to modify the current dual solution  $\bar{y}$  in order to improve the lower bound.

Now, for a given primal variable  $x_j$  and a given dual solution  $\bar{y}$ , consider the following quantity:

$$\bar{c}_j(\bar{y}) = c_j - \sum_{i=1}^m \bar{y}_i A_{ij}.$$

If  $\bar{y}$  is an optimal dual solution, then  $\bar{c}_j(\bar{y})$  is the standard reduced cost, denoted by  $\rho_j^*$  above. Even if  $\bar{y}$  is not an optimal dual solution, however, it might still contain some useful information. In particular, one might expect variables with a small  $\bar{c}$  value to have a high probability of belonging to an optimal solution of the original ILP. Thus, the  $\bar{c}$  values can be used within a heuristic.

The above observations led Erlenkotter to propose a primal heuristic for the UFLP, in which one iteratively opens facilities with zero  $\bar{c}$  values until certain ‘complementary slackness’ conditions are met. The results were very encouraging, and the method was subsequently improved in Körkel (1989), Janáček and Buzna (2008), and Letchford and Miller (2012).

Later on, dual ascent and dual adjustment were successfully applied to many other COPs, including (in roughly chronological order) the Steiner tree problem (Wong, 1984; Polzin and Daneshmand, 2001), the multi-dimensional knapsack problem (Magazine and Oguz, 1984; Angelelli et al.,

2010), the set covering problem (Balas and Ho, 1980; Beasley, 1987; Balas and Carrera, 1996), the uncapacitated network design problem (Balakrishnan et al., 1989), the set partitioning problem (Boschetti et al., 2008; Fisher and Kedia, 1990), the  $p$ -median problem (Captivo, 1991), the design of railway networks (Keaton, 1992), the multi-period assignment problem (Murthy, 1993), the capacitated multi-commodity flow problem (Barnhart, 1993), the hub location problem (Klincewicz, 1996), the quadratic assignment problem (Hahn and Grant, 1998) and the fixed-charge transportation problem (Buson et al., 2014).

Some authors have considered more sophisticated matheuristics, which attempt to exploit primal and dual information in a more intelligent way. For brevity, we mention just a few examples. Fisher and Kedia (1990) improved the approach to the set covering problem by using a local search phase to improve the dual solution, along with an improved primal heuristic. Wedelin (1995) presented a primal–dual heuristic for a certain generalisation of the set partitioning problem arising in crew scheduling. The dual is solved heuristically via coordinate ascent. Later on, Hansen et al. (2007) and Posta et al. (2014) improved the dual ascent approach to the UFLP by incorporating sophisticated local search procedures for improving both primal and dual solutions.

To close this section, we mention that there is significant literature on *primal–dual approximation algorithms* for NP-hard COPs. These algorithms iteratively build heuristic solutions to the primal and dual in parallel, in such a way that the gap between the corresponding upper and lower bounds is bounded in a specified way. For details, see the textbooks (Vazirani, 2001; Williamson and Shmoys, 2011).

### 3. Lagrangian and surrogate relaxation

In this section, we consider matheuristics that are based on LR, SR or closely related methods. Subsection 3.1 recalls the basics of LR and SR. Subsections 3.2 and 3.3 review some LR-based and SR-based matheuristics. Subsection 3.4 mentions some related methods, such as Lagrangian dual ascent, semi-LR and additive bounding.

#### 3.1. Recap on Lagrangian and surrogate relaxation

Some COPs of interest can be formulated as ILPs of the form

$$\min \{c^T x : Ax \geq b, Cx \geq d, x \in \mathbb{Z}_+^n\}, \quad (4)$$

where the constraints  $Ax \geq b$  are ‘easy’ and the constraints  $Cx \geq d$  are ‘hard’. By this, we mean that, if the ‘hard’ constraints are dropped, the problem becomes significantly easier to solve.

Let us suppose that the number of ‘hard’ constraints is  $t$ . In LR, we pick a vector  $\lambda \in \mathbb{R}_+^t$  of *Lagrangian multipliers* and then solve the following simpler ILP:

$$\min \{c^T x + \lambda^T (d - Cx) : Ax \geq b, x \in \mathbb{Z}_+^n\}. \quad (5)$$

Geoffrion (1974) showed that, for any choice of  $\lambda$ , LR yields a lower bound for the original ILP. We will call this bound  $L(\lambda)$ .



The problem of finding the vector  $\lambda$  which maximises  $L(\lambda)$  is called the *Lagrangian dual* (LD). The LD is a piecewise-linear concave maximisation problem, and there exist several algorithms for solving it either exactly or approximately (see, e.g., Lemaréchal, 2001; Guignard, 2003).

*Surrogate relaxation*, proposed in Glover (1975) and Greenberg and Pierskalla (1970), is similar to LR. The difference is that, instead of modifying the objective function in the ILP (4), we replace the complicating constraints  $Cx \geq d$  with a single linear constraint. More precisely, for a given multiplier vector  $\lambda$ , we solve the following ILP:

$$\min \{c^T x : Ax \geq b, (\lambda^T C)x \geq \lambda^T d, x \in \mathbb{Z}_+^n\}. \quad (6)$$

In most applications of SR, the system  $Ax \geq b$  takes a very simple form. As a result, the ILP (6) is usually some kind of knapsack problem and solved in pseudo-polynomial time by DP.

Let us denote by  $S(\lambda)$  the lower bound obtained with SR. It is proved in Greenberg and Pierskalla (1970) that  $S(\lambda) \geq L(\lambda)$  for any given  $\lambda$ . The problem of finding the vector  $\lambda$  which maximises  $S(\lambda)$  is called the *surrogate dual* (SD). The SD is a quasi-concave maximisation problem and is typically somewhat harder to solve than the LD (see, e.g., Karwan and Rardin, 1984; Kim and Kim, 1998; Dokka et al., 2021).

### 3.2. Lagrangian heuristics

For a given multiplier vector  $\lambda$ , let  $\bar{x}(\lambda)$  be the solution to the relaxed problem (5). By definition,  $\bar{x}(\lambda)$  is an integer and satisfies the constraints  $Ax \geq b$ , but it might fail to satisfy the constraints  $Cx \geq d$ . For some specific families of ILPs, it is possible to ‘repair’  $\bar{x}(\lambda)$ , with not much effort, in order to obtain a heuristic solution for the original ILP. (For instance, in the generalised assignment problem, each job must be assigned to exactly one machine. If  $\bar{x}(\lambda)$  does not satisfy this condition, one can attempt to convert it into a feasible solution by eliminating multiple assignments and then trying to assign any unassigned jobs to machines having enough free capacity (Jörnsten and Näsberg, 1986).) This can be repeated at each iteration with the corresponding multiplier vector, and one can select the best heuristic solution found.

Fisher (1985) called heuristics of this kind ‘Lagrangian heuristics’, but one can view them as a particular kind of matheuristic. They have been applied with great success to several classical combinatorial optimisation problems, such as single machine scheduling (Fisher, 1976), the set covering problem (Balas and Ho, 1980; Beasley, 1990), the capacitated vehicle routing problem (Fisher et al., 1982), the generalised assignment problem (GAP) (Jörnsten and Näsberg, 1986), the many-to-many assignment problem (Litvinchev et al., 2010), and various facility location problems (Barceló and Casanovas, 1984; Pirkul, 1987; Galvão and Raggi, 1989; Beasley, 1993).

Lagrangian matheuristics have also been developed for a huge array of more realistic practical problems, such as manpower planning (Glover et al., 1979), scheduling of energy generators (Lauer et al., 1982; Bard, 1988), product distribution (Van Roy and Gelders, 1981; Bell et al., 1983), lot sizing (Billington et al., 1986; Trigeiro, 1987), school timetabling (Carter, 1989), aircraft assignment (Daskin and Panayotopoulos, 1989), railway network design (Keaton, 1989), fixed-charge problems (Wright and von Lanzener, 1989), hybrid flowshop scheduling (Mao et al., 2014), capacitated

network design (Holmberg and Yuan, 2000) and the closest string problem in computational biology (Tanaka, 2012).

We remark that, for a given primal variable  $x_j$  and a given multiplier vector  $\lambda$ , the quantity

$$c_j - \sum_{i=1}^t \lambda_i C_{ij}$$

can be viewed as a ‘Lagrangian reduced cost’. These values can be used to guide heuristics, just as we saw for the LP-reduced costs in Subsection 2.3 and the ‘approximate’-reduced costs in Subsection 2.4. This idea has been used to particularly good effect in Lagrangian matheuristics for the set covering problem (Balas and Ho, 1980; Beasley, 1990; Balas and Carrera, 1996; Caprara et al., 1999; Ceria et al., 1998) and the capacitated facility location problem (Avella et al., 2009).

### 3.3. Heuristics based on surrogate relaxation

As far back as 1977, Glover (1977) suggested using SR to drive heuristics. As in the case of LR, the idea is to take the solution to the relaxed problem and ‘repair’ it, to make it feasible for the original problem. Although this idea has received less attention than the Lagrangian approach, it has been applied to several problems, such as loading problems (Fisk and Hung, 1979), manpower planning (Glover et al., 1979), single-machine scheduling (Fisher et al., 1983), the multi-dimensional knapsack problem (Fréville and Plateau, 1986; Pirkul, 1987; Dokka et al., 2022), resource-constrained scheduling (Dobson and Khosla, 1995) and a variant of the quadratic knapsack problem (Létocart et al., 2014).

More recently, Dokka et al. (2021) proposed a general framework for designing matheuristics based on SR. The idea is to exploit the fact that the relaxed problem (6) is usually solved via DP. The nature of DP is that it constructs and stores a large number of ‘intermediate’  $x$  vectors as it goes along. These  $x$  vectors are integral but unlikely to be feasible for the original ILP. Accordingly, Dokka et al. propose to take some or all of the  $x$  vectors and ‘repair’ them in the usual way.

Some authors have also experimented with ‘hybrids’ of Lagrangian and surrogate relaxation. See Lorena and Lopes (1994), Lorena and Narciso (1996) and Galvão et al. (2000) for applications to the set covering problem, the GAP and the maximal covering location problem, respectively.

### 3.4. Variants

We now mention some variations of LR that have also been used to drive heuristics. These variations are presented in more-or-less chronological order.

*Lagrangian dual ascent.* *Lagrangian dual ascent* (also known as multiplier adjustment) is a hybrid of LR and dual ascent (Fisher, 1981; Guignard and Rosenwein, 1989). It is basically a greedy constructive heuristic for the Lagrangian dual, just as the dual ascent is a greedy constructive heuristic for the LP dual. The idea is to set the Lagrangian multipliers to some simple initial

values (e.g., zero) and then increase them one at a time, in such a way that the lower bound  $L(\lambda)$  is guaranteed to increase monotonically. This approach to solving the Lagrangian dual tends to be much faster than the subgradient method, though it may come at the expense of a weaker lower bound. The method has been used to derive matheuristics for the capacitated vehicle routing problem (Fisher and Jaikumar, 1981), the GAP (Fisher et al., 1986; Guignard and Rosenwein, 1989), the uncapacitated facility location problem (Guignard, 1998), the segregated storage problem (Neebe, 1987), capacity planning in manufacturing (Lim and Kim, 1998), the management of cross-docking terminals (Monaco and Sammarra, 2022) and various problems in telecommunications (Lin and Yee, 1992; Chang and Gavish, 1995).

*The restricted Lagrangian approach.* The *restricted Lagrangian* approach was introduced by Balas and Christofides (1981), in the context of the asymmetric travelling salesman problem (ATSP). Suppose once more that we have an ILP in the form (9), where the problem becomes much easier to solve when the constraints  $Cx \geq d$  are dropped. The first step in the approach is to solve the relaxed problem, obtaining a primal vector  $x^* \in \mathbb{Z}_+^n$  and a lower bound. After that, we apply a ‘restricted’ form of LR, in an attempt to increase the lower bound. We are permitted to assign positive Lagrangian multipliers to one or more of the constraints in the system  $Cx \geq d$ , but only under the condition that  $x^*$  remains optimal for the relaxed problem (or, equivalently, that the Lagrangian reduced cost remains at zero for any variable  $x_j$  taking a positive value at  $x^*$ ). Balas and Christofides provide two fast procedures for determining the multipliers.

An interesting feature of the restricted Lagrangian approach is that the set of  $x$  variables having zero Lagrangian reduced cost grows during the course of the algorithm. At the end of the procedure, Balas and Christofides use an enumerative procedure to search for an ATSP solution that uses only arcs of zero-reduced cost. This last step can be viewed as yet another early example of a matheuristic.

*Additive bounding.* A generalisation of the restricted Lagrangian approach, called *additive bounding*, was proposed by Fischetti and Toth (1989). Let us assume for simplicity that our COP has been formulated as an ILP of the form (1). We suppose that there are several fast lower-bounding procedures for our COP, each of which exploits a different substructure of the problem. We also assume that each such procedure returns a dual vector. The procedures are then applied in some chosen sequence. Suppose the first procedure terminates with dual vector  $y^1$ . We store the corresponding lower bound  $b^T y^1$  and replace the original cost vector  $c$  with the reduced cost vector  $c^1 = c - A^T y^1$ . We then feed the reduced cost vector into the second procedure, yielding a new dual vector  $y^2$ . We then increase our lower bound by  $b^T y^2$  and replace the vector  $c^1$  with the new vector  $c^2 = c^1 - A^T y^2$ . This process is repeated until all procedures have been terminated.

The additive bounding procedure has been mainly used to fathom nodes within branch-and-bound algorithms. It has however also been used to develop matheuristics. Fischetti and Toth (1992) proposed an additive bounding procedure for the ATSP that generates heuristic solutions using an approach similar to the one proposed by Balas and Christofides (1981). Later on, Vigo (1996) proposed a heuristic for the asymmetric capacitated vehicle routing problem that makes use of an additive bounding procedure for generating an initial solution. Later still, Caprara (2002) used additive bounding to devise an approximation algorithm for the *breakpoint median problem*, a

well-known problem in computational biology. We remark that additive bounding was also been used for generating good dual solutions within the incremental core approach described at the end of Subsection 2.3.

*Relaxation adaptive memory programming.* Relaxation adaptive memory programming (RAMP) was proposed by Rego (2005). It is similar to the hybrid Lagrangian/surrogate approaches mentioned above, but one is permitted to improve both primal and dual solutions along the way, using previously existing metaheuristics (such as scatter search and path relinking (Glover et al., 2000)). The method has been applied to the capacitated minimum spanning tree problem (Rego et al., 2010), resource constrained project scheduling (Riley and Rego, 2019) and capacitated facility location problems (Oliveira et al., 2021).

*Semi-Lagrangian relaxation.* *Semi-Lagrangian* relaxation (Beltran-Royo et al., 2006) is designed for COPs that have a natural ILP formulation of the form

$$\min \{c^T x : Ax = b, x \in \mathcal{X}\},$$

where  $A$ ,  $b$  and  $c$  are non-negative and  $\mathcal{X}$  is a ‘reasonably simple’ subset of  $\mathbb{Z}_+^n$ . The idea is that we split the equation system  $Ax = b$  into two inequality systems,  $Ax \leq b$  and  $Ax \geq b$ . We then relax the latter in Lagrangian fashion. The relaxed problem takes the form:

$$\min \{c^T x + \lambda^T (b - Ax) : Ax \leq b, x \in \mathcal{X}\}. \quad (7)$$

For some COPs, the relaxed ILP can be solved much more easily than the original. In particular, this happens if variables with a positive objective coefficient in (7) must take the value zero in an optimal solution to (7). In this case, it often happens that a large proportion of the variables can be eliminated.

As usual, one can often repair the solution of the relaxed problem to obtain heuristic solutions for the original COP. Matheuristics of this kind have been developed for the  $p$ -median problem (Beltran-Royo et al., 2006), facility location problems (Beltran-Royo et al., 2012; Monabbati, 2014; Jörnsten and Klöse, 2016; Cabezas and García, 2022), the quadratic assignment problem (Zhang et al., 2016), and the design of multi-commodity distribution centres (Zhang et al., 2019).

#### 4. Methods based on decomposition

Another important family of matheuristics is composed of those that are based on *Dantzig–Wolfe decomposition* (Dantzig and Wolfe, 1960) or *BD* (Benders, 1962). We recall the basic ideas of these decomposition schemes in Subsections 4.1 and 4.2. Matheuristics based on the schemes are reviewed in Subsections 4.3 and 4.4. The reader interested in more details is referred to Boschetti et al. (2010), Maniezzo et al. (2021) and Raidl and Puchinger (2008).

#### 4.1. Recap on Dantzig–Wolfe decomposition

Consider a MIP of the form

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & Ax \geq b \\ & x^s \in X^s \quad (s = 1, \dots, t) \\ & x \in \mathbb{R}_+^n, \end{aligned} \tag{8}$$

where the vector  $x$  has been partitioned into sub-vectors  $x^1, \dots, x^s$ .

Dantzig–Wolfe decomposition works as follows. For  $s = 1, \dots, t$ , let  $P^s$  be the convex hull of  $X^s$  and let  $p^s(1), \dots, p^s(n_s)$  denote the extreme points of  $P^s$ . We add a new continuous variable, say  $\lambda_k^s$ , for  $s = 1, \dots, t$  and  $k = 1, \dots, n_s$ . We then reformulate the MIP by replacing the constraints (8) with

$$\begin{aligned} x^s &= \sum_{k=1}^{n_s} p^s(k)^T \lambda_k^s \quad (s = 1, \dots, t) \\ \sum_{k=1}^{n_s} \lambda_k^s &= 1 \quad (s = 1, \dots, t) \\ \lambda_k^s &\in \mathbb{R}_+^{n_s} \quad (s = 1, \dots, t). \end{aligned}$$

The  $x$  variables can then be eliminated if desired.

It can be shown (using a similar argument to the one that Geoffrion (1974) used for LR) that the LP relaxation of the new MIP gives a lower bound that is at least as strong as the one from the LP relaxation of the original MIP. Moreover, the LP relaxation of the new MIP can be solved by an iterative procedure that starts with a subset of the  $\lambda$  variables and uses dual prices to generate other  $\lambda$  variables as needed. This procedure is called *column generation*.

If desired, the whole procedure can be embedded within a branch-and-bound framework. This overall approach is called *branch-and-price* (Barnhart et al., 1998).

#### 4.2. Recap on Benders decomposition

Now consider a MIP of the form

$$\min \{c_1^T x + c_2^T y : Ax + By \geq b, x \in X, y \geq 0\}. \tag{9}$$

We assume for simplicity that this MIP is feasible and bounded.

Benders (1962) noted that one can write the MIP as

$$\min \{c_1^T x + f(x) : x \in X\},$$

where  $f(x)$  is the optimal solution to the following subproblem:

$$\min \{c_2^T y : By \geq b - Ax, y \geq 0\}.$$

By LP duality, we have

$$f(x) = \max \{(b - Ax)^T w : B^T w \leq c_2, w \geq 0\}. \quad (10)$$

Now, let  $p^1, \dots, p^t$  be the extreme points of the feasible region of (10). By definition, we have

$$f(x) = \max_{1 \leq s \leq t} \{(b - Ax)^T p^s\}.$$

Thus, the original MIP can be reformulated as

$$\min c_1^T x + z, \quad (11)$$

$$\text{s.t. } z \geq b^T p^s - (B^T p^s)^T x \quad (s = 1, \dots, t), \quad (12)$$

$$x \in X, \quad (13)$$

where (11)–(13) is the *master problem* and constraints (12) are the so-called *Benders' cuts*. Since  $t$  is usually huge, the master problem is initially solved with only a small number of Benders' cuts, and the others are generated as needed with an iterative procedure.

### 4.3. Dantzig–Wolfe decomposition heuristics

There are several ways in which one can use Dantzig–Wolfe decomposition within a matheuristic scheme. One way is to use diving heuristics instead of branch-and-price, as we mentioned in Subsection 2.2. Another way is to stop when a desired number of columns has been generated and then feed the current master MIP into a branch-and-bound solver. An early example of this approach is Agarwal et al. (1989), who applied it to the capacitated vehicle routing problem.

A third option is to use a fast heuristic for the pricing subproblem, instead of an exact method. One of the first proposals along these lines is presented in Caserta and Voß (2012), where the authors deal with a lot of sizing problem with setup times and costs and where columns correspond to suboptimal production plans. It turns out that even the pricing subproblem is *NP*-hard for this problem. Given that optimal solutions to the subproblem are not necessarily needed to produce negative-cost columns, a corridor method (see Section 5.5) is used to generate them. The overall method is heuristic in nature, given the fact that the pricing subproblem is solved heuristically.

Another work (Cunha et al., 2019), related to lot sizing, proposes to include in the heuristic column generation scheme outlined above a ‘fix-and-optimize’ procedure (related to relax-and-fix, Section 2.2), which fixes the non-integer variables produced by the column generation component.

Several other works apply column generation for heuristic search without explicitly mentioning Dantzig–Wolfe (DW) decomposition. The addressed range of problems is wide, including technician routing (Dupin et al., 2021), train timetabling (Martin-Irardi and Ropke, 2022), electric vehicle

routing (Echeverri et al., 2019), city logistics (Boschetti and Maniezzo, 2015) and planning in bulk ports (de Andrade and Menezes, 2023) among others.

#### 4.4. Benders decomposition heuristics

Recently, there has been a surge of interest in heuristics based on BD, which is well suited to modelling problems that can be decomposed into two interacting subproblems. Until recently, the possibility of using BD to design efficient heuristics was limited by the need to solve the master problem to optimality in order to obtain at least a valid bound, a task that becomes increasingly complex as newly generated Benders' cuts are added. The increased efficiency of MIP solvers now allows BD to be used as a valuable technique for exploiting problem separability in the design of primal heuristics.

Several real-world applications based on Benders' matheuristics have been described. A first approach attempts to address the inability to produce a feasible solution before the end of cut generation. For example, Kergosien et al. (2017) describe an application to chemotherapy production and delivery, a problem that can be decomposed into two stages, a parallel machine scheduling problem combined with a multi-trip travelling salesman problem. The master problem consists in finding the sequence of trips, and the slave problem is a parallel machine scheduling problem. A solution of the master problem is given to a tabu search heuristic, which derives a heuristic solution.

Another example deals with the identification of the economic operating point of power systems (Saberri et al., 2020). Here, BD verifies a security criterion through an LP-based master problem, and the subproblem involves the solution of a non-linear program to compute the system energy for transient stability evaluation.

Taking a less problem-dependent approach, Maher (2021) describes how the general BD framework of the MIP solver SCIP was extended with two heuristics, a trust region-based heuristic and a large neighbourhood search, to improve the BD algorithm and the ability to find primal feasible solutions.

Another notable contribution related to BD was the introduction of the so-called *combinatorial Benders' cuts* (Codato and Fischetti, 2006). These follow the same structure as classical BD, but the subproblem is an ILP rather than an LP. In case of an infeasible master solution, the subproblem returns combinatorial inequalities to be added as cuts to the master. The method was originally intended for exact MIP solutions, but it has been adapted, possibly more easily than classical BD, to design matheuristics. For example, Bai and Rubin (2009) describe a combinatorial Benders' cut approach to minimising the number of required toll facilities in a transportation network, and Côté et al. (2014) deal with the strip packing problem, where the master problem cuts items into unit-width slices and packs them into the strip, while the subproblem tries to reconstruct the rectangular items by fixing the vertical positions of their unit-width slices.

## 5. Some other matheuristics

In this section, we consider some matheuristics that do not necessarily yield dual bounds, yet have proven to be very successful in some applications.

### 5.1. Local branching

*Local branching* (Fischetti and Lodi, 2003) is a method that solves MIPs to perform a local search. For brevity, we explain how it works only for pure 0–1 LPs. Let  $x^h \in \{0, 1\}^n$  be the current incumbent heuristic solution, and let  $k$  be a given positive integer parameter. We wish to explore the  $k$ -opt neighbourhood of  $x^h$ , by which we mean the set of all feasible solutions that can be obtained by changing the value of no more than  $k$  variables. To do this, we define the set  $S = \{j \in \{1, \dots, n\} : x_j^h = 1\}$  and add the following constraint to the 0–1 LP:

$$\sum_{j \notin S} x_j + \sum_{j \in S} (1 - x_j) \leq k.$$

Provided  $k$  is small enough, it is likely that the modified 0–1 LP can be solved much more quickly than the original. For this purpose, one can use any decent ILP solver. If the solution to the modified 0–1 LP is cheaper than  $x^h$ , it takes the place of  $x^h$ , and the process is repeated.

### 5.2. Very large-scale neighbourhood search

Local branching can be viewed as a special case of the *very large-scale neighbourhood* (VLSN) search, which means local search based on neighbourhoods whose cardinality is permitted to be huge and possibly even exponentially large in the number of variables (Ahuja et al., 2000, 2002).

Broadly speaking, VLSN heuristics can be put into three categories: (i) those that use a heuristic to perform a partial search of the neighbourhood; (ii) those that solve a classic COP (such as a matching problem or network flow problem) to explore the neighbourhood fully in polynomial time; and (iii) those that explore the neighbourhood by solving an auxiliary MIP, as we saw above in the case of local branching.

A good example of the first type is the approach in Thompson and Psaraftis (1993), which is based on a one-to-one correspondence between ‘improving cyclic exchanges’ and negative-cost cycles in a certain auxiliary graph. This idea has been applied to vehicle routing problems (Thompson and Psaraftis, 1993), parallel machine scheduling (Frangioni et al., 2004), the graph colouring problem (Chiarandini et al., 2008) and timetabling problems (Meyers and Orlin, 2007), among others. Other influential approaches of the first type are ‘large neighbourhood search’ (Shaw, 1998) and ‘adaptive large neighbourhood search’ (Ropke and Pisinger, 2006).

A good example of the second type of VLSN approach is the ‘matching’ neighbourhood for the TSP (Sarvanov and Doroshko, 1981), which was later adapted to several other problems, such as scheduling problems (Brueggemann and Hurink, 2007, 2011) and the GAP (Mitrović-Minić and Punnen, 2008, 2009).

As for the third type, we already mentioned local branching above. Another excellent example of the third type is the heuristic for the capacitated vehicle routing problem in De Franceschi et al. (2006). It iteratively removes edges (and short paths) from the incumbent solution and then solves a small ILP to find the cheapest way to add edges and restore feasibility.

We remark that MIP solvers have become increasingly efficient over the past couple of decades and now incorporate a variety of advanced techniques to tackle hard problems. Thus, using a MIP



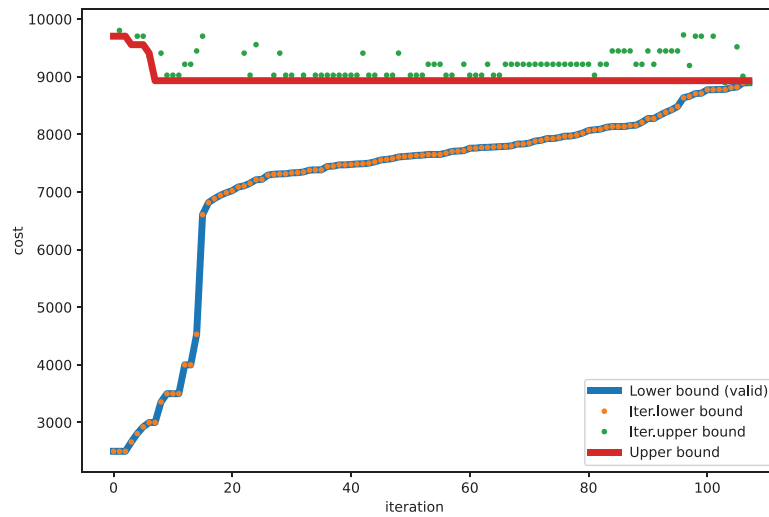


Fig. 3. Benders' heuristic, upper and lower bound per iteration.

solver to explore a neighbourhood is now a perfectly reasonable approach. Using complete MIP models for solving subproblems has been called ‘MIPping’ (Fischetti et al., 2009). The current literature contains several other examples of this. For instance, one can use a MIP solver to solve to optimality a Benders' master problem, in order to obtain a valid lower bound. Figure 3 shows some results that we obtained with such an approach.

To close this subsection, we remark that the ‘ejection chain’, a well-known meta-heuristic proposed by Glover (1992) sharing similarities with relax-and-fix (Section 2.2), with ‘construct, merge, solve and adapt’ (CMSA; Blum et al., 2016) and with ‘adaptive large neighbourhood search’ (ALNS; Ropke and Pisinger, 2006), has also been extended with MP modules (Burke and Curtois, 2014; Adamo et al., 2020), making it convergent with VLSN and with other similar approaches based on decomposition into subproblems (Chirayil Chandrasekharan et al., 2021).

### 5.3. Dynamic programming heuristics

*DP*, created by Bellman (1957), is a classical technique for solving optimisation problems that have a certain ‘sequential’ structure. Although *DP* was originally intended to be an exact method, it has been used to devise heuristics for a range of COPs, including machine grouping in cellular manufacturing (Steudel and Ballakur, 1987), the capacitated minimum spanning tree problem (Gouveia and Paixao, 1991), the TSP with precedence constraints (Bianco et al., 1994), the time-dependent TSP (Malandraki and Dial, 1996), the  $p$ -median problem (Hribar and Daskin, 1997), the pallet loading problem (Scheithauer and Terno, 1996), the multi-dimensional knapsack problem (Bertsimas and Demir, 2002; Masmoudi et al., 2023), assembly line balancing (Bautista and Pereira, 2009) and the quadratic knapsack problem (Djeumou Fomeni and Letchford, 2014). Moreover, as already mentioned in Subsection 3.3, matheuristics have been devised by exploiting the fact that *DP* is often used when applying surrogate relaxation.

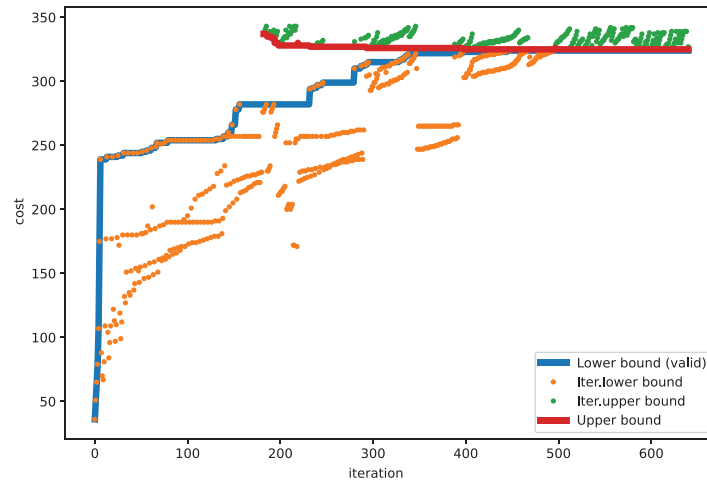


Fig. 4. Fore-and-back, upper and lower bound evolution.

#### 5.4. Fore-and-back

*Fore-and-Back (FB)* was devised by Bartolini et al. (2008). It is suitable for NP-hard problems that have a ‘natural’ DP formulation. (For instance, we can formulate the GAP as a DP, by ordering the jobs and then having one stage for each job, in which we allocate the given job to one of the machines (Maniezzo et al., 2021).) If FB is given enough computational resources (memory and time), FB is actually an exact method. It was however intended to be a matheuristic, having been designed for quickly finding high-quality integer solutions.

FB is an iterated primal-only constructive method, but it computes bounds on the cost of completing partial solutions. This permits it to discard partial solutions from consideration, and sometimes even to compute lower bounds for the entire problem. The basic idea is to perform a series of DP computations while imposing a restriction on the amount of memory used. The first DP is performed in a ‘forward’ direction, whereas the second is in the ‘reverse’ direction. (In the GAP example, this could correspond to considering jobs in the order  $1, 2, \dots, n$  in the forward phase, but in the order  $n, n - 1, \dots, 1$  in the backward phase.)

Following this, a third DP is performed in the forward direction again, and so on until some termination condition is met. Along the way, ‘promising’ partial solutions are stored in memory. In either direction, partial solutions can be tentatively extended to feasible ones using the information stored in the previous DP computation.

FB has been successfully applied to network design problems (Bartolini and Mingozzi, 2009) and GAPs (Maniezzo et al., 2021). Figure 4 shows a trace of an FB run, where at each DP expansion stage we plot the upper and lower bounds computed with the data available at that stage (green and orange dots, respectively). The figure also shows the evolution of the overall best upper and lower bounds as solid lines (red and blue lines, respectively). Note how the lower bounds computed during the forward and backward passes are clearly distinguishable, forming a sort of vertical stripes in the plot. For this particular instance, the first feasible solution is obtained during the first backward pass, and most of the improving feasible solutions are obtained during subsequent backward passes.

### 5.5. The corridor method

The *corridor method* (CM) (Sniedovich and Voß, 2006) is a well-established matheuristic paradigm that originated as a DP adaptation, though it was later extended to other exact approaches such as branch-and-bound. The name comes from the idea of directing the dynamics of expansions in DP, channelising them as in a corridor, where the walls are represented by additional constraints imposed on the instance. CM can thus be used to solve problems for which we know an exact method (DP, also branch-and-bound, branch-and-cut, etc.) that can solve them effectively, but only on smaller instances than the one we are interested in. CM then applies the exact method over successive restricted parts of the solution space, enforcing bounds on the possible expansions of the current state or partial solution, i.e., defining reduced neighbourhoods of the current partial solution. Extensions of other search techniques, such as branch-and-bound, may involve searching in the space of feasible solutions, where a first solution is somehow generated and then constraints on its possible adaptations are added. These constraints can be diving like on the absolute number of variables that can be changed, or they can be more problem related, such as the number of facilities that can be opened/closed in facility location problems, or the number of jobs that can be rescheduled in a scheduling problem. In this case, the corridor constraints define local search neighbourhoods. A procedure such as that proposed by relax-and-fix (see Section 2.2) can be seen as working on the linear relaxation of the problem, with the integrality requirements acting as corridor constraints.

Search often results in having to deal with neighbourhoods that are exponentially large, but that correspond to instances that can be efficiently solved by the core exact method. The execution is heavily influenced by a control parameter,  $\delta_{max}$ , which specifies the maximum ‘width of the corridor’, which is a measure somehow quantifying the maximum size of the subproblems passed to the exact method. It is commonly implemented as a simple extension that supports dynamic corridor widths, adjusting the width of the corridor depending on whether or not improving solutions have been found in the current neighbourhood. If an improving solution is found in a small neighbourhood, the incumbent solution is updated and a new corridor is defined around this new solution. Otherwise, the width of the corridor is increased in the hope of helping to find feasible solutions. Figure 5 shows a trace of a run, showing the iteration upper and lower bounds.

## 6. Discussion

The field of matheuristics has only recently emerged as an independent area of research, but it draws on a rich and extensive body of contributions. In this survey, we have argued that matheuristics offer a key advantage over other meta-heuristics; namely, the use of dual bounds to enable one to terminate early and/or to evaluate the quality of the best-found primal solution. Another nice feature of matheuristics is that they enable one to exploit the relevant progress that has been made in theory, algorithms and software for integer programming.

Generally speaking, we believe that there is no dominance relation between matheuristics and the more widely used, ‘mathematics-free’ meta-heuristics. Nevertheless, relative superiorities can be found on a per-problem basis. In particular, greater effectiveness of matheuristics has been reported

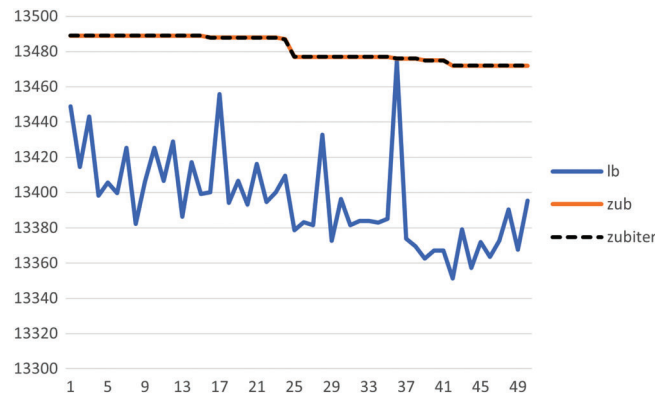


Fig. 5. Corridor, upper and lower bound per iteration.

for highly constrained problems, where the feasible solutions are very sparse and thus simple local search becomes expensive and ineffective (Maniezzo et al., 2021).

We remark that many of the approaches mentioned, such as LR, dual ascent and additive bounding, do not rely on the solution of large-scale integer programs at all. For this reason, they are particularly attractive when one is dealing with large-scale problem instances. Moreover, methods such as LR and Dantzig–Wolfe decomposition often involve the solution of several independent subproblems, which means that they can exploit parallel processors, and possibly even lead to fully distributed algorithms. Finally, heuristics based on surrogate and semi-LR have received relatively little attention so far, and we believe that they merit further study.

## Acknowledgments

Open Access Funding provided by Universita degli Studi di Bologna within the CRUI-CARE Agreement.

## References

- Aboudi, R., Jörnsten, K., 1994. Tabu search for general zero-one integer programs using the pivot and complement heuristic. *ORSA Journal of Computing* 6, 82–93.
- Adamo, T., Ghiani, G., Guerriero, E., Manni, E., 2020. Ejection chain moves for automatic neighborhood synthesis in constrained cardinality-minimization problems. *International Transactions in Operational Research* 27, 1210–1235.
- Agarwal, Y., Mathur, K., Salkin, H.M., 1989. A set-partitioning-based exact algorithm for the vehicle routing problem. *Networks* 19, 731–749.
- Ahuja, R.K., Ergun, Ö., Orlin, J.B., Punnen, A.P., 2002. A survey of very large-scale neighbourhood search techniques. *Discrete Applied Mathematics* 123, 75–102.
- Ahuja, R.K., Orlin, J.B., Sharma, D., 2000. Very large-scale neighborhood search. *International Transactions in Operational Research* 7, 301–317.
- Angelelli, E., Mansini, R., Speranza, M.G., 2010. Kernel search: a general heuristic for the multi-dimensional knapsack problem. *Computers & Operations Research* 37, 2017–2026.

© 2023 The Authors.

*International Transactions in Operational Research* published by John Wiley & Sons Ltd on behalf of International Federation of Operational Research Societies

- Angelelli, E., Mansini, R., Speranza, M.G., 2012. Kernel search: a new heuristic framework for portfolio selection. *Computational Optimization and Applications* 51, 345–361.
- Archetti, C., Guastaroba, G., Huerta-Muñoz, D.L., Speranza, M.G., 2021. A kernel search heuristic for the multivehicle inventory routing problem. *International Transactions in Operational Research* 28, 2984–3013.
- Ausiello, G., Crescenzi, P., Gambosi, G., Kann, V., Marchetti-Spaccamela, A., Protasi, M., 1999. *Complexity and Approximation*. Springer, Heidelberg.
- Avella, P., Boccia, M., Sforza, A., Vasil'ev, I., 2009. An effective heuristic for large-scale capacitated facility location problems. *Journal of Heuristics* 15, 597–615.
- Bai, L., Rubin, P.A., 2009. Combinatorial Benders cuts for the minimum tollbooth problem. *Operations Research* 57, 1510–1522.
- Balakrishnan, A., Magnanti, T.L., Wong, R.T., 1989. A dual-ascent procedure for large-scale uncapacitated network design. *Operations Research* 37, 716–740.
- Balas, E., Carrera, M.C., 1996. A dynamic subgradient-based branch-and-bound procedure for set covering. *Operations Research* 44, 875–890.
- Balas, E., Ceria, S., Dawande, M., Margot, F., Pataki, G., 2001. OCTANE: a new heuristic for pure 0–1 programs. *Operations Research* 49, 207–225.
- Balas, E., Christofides, N., 1981. A restricted Lagrangian approach to the traveling salesman problem. *Mathematical Programming* 21, 19–46.
- Balas, E., Ho, A., 1980. Set covering algorithms using cutting planes, heuristics, and subgradient optimization: a computational study. *Mathematical Programming Studies* 12, 37–60.
- Balas, E., Martin, C.H., 1980. Pivot and complement—a heuristic for 0-1 programming. *Management Science* 26, 86–96.
- Ball, M.O., 2011. Heuristics based on mathematical programming. *Surveys in Operations Research and Management Science* 16, 21–38.
- Barceló, J., Casanovas, J., 1984. A heuristic Lagrangian algorithm for the capacitated plant location problem. *European Journal of Operational Research* 15, 212–226.
- Bard, J.F., 1988. Short-term scheduling of thermal-electric generators using Lagrangian relaxation. *Operations Research* 36, 756–766.
- Barnhart, C., 1993. Dual-ascent methods for large-scale multicommodity flow problems. *Naval Research Logistics* 40, 305–324.
- Barnhart, C., Johnson, E.L., Nemhauser, G.L., Savelsbergh, M.W.P., Vance, P.H., 1998. Branch-and-price: column generation for solving huge integer programs. *Operations Research* 46, 316–329.
- Bartolini, E., Maniezzo, V., Mingozzi, A., 2008. An adaptive memory-based approach based on partial enumeration. In Maniezzo, V., Battiti, R., Watson, J.-P. (eds) *Proceedings of the LION '07*. Springer, Heidelberg, pp. 12–24.
- Bartolini, E., Mingozzi, A., 2009. Algorithms for the non-bifurcated network design problem. *Journal of Heuristics* 15, 259–281.
- Bautista, J., Pereira, J., 2009. A dynamic programming based heuristic for the assembly line balancing problem. *European Journal of Operational Research* 194, 787–794.
- Beasley, J.E., 1987. An algorithm for set covering problem. *European Journal of Operational Research* 31, 85–93.
- Beasley, J.E., 1990. A Lagrangian heuristic for set covering problems. *Naval Research Logistics* 37, 151–164.
- Beasley, J.E., 1993. Lagrangian heuristics for location problems. *European Journal of Operational Research* 65, 383–399.
- Bell, W.J., Dalberto, L.M., Fisher, M.L., Greenfield, A.J., Jaikumar, R., Kedia, P., Mack, R.G., Prutzman, P.J., 1983. Improving the distribution of industrial gases with an on-line computerized routing and scheduling optimizer. *Interfaces* 13, 4–23.
- Bellman, R.E., 1957. *Dynamic Programming*. Princeton University Press, Princeton, NJ.
- Beltran-Royo, C., Tadonki, C., Vial, J.-P., 2006. Solving the  $p$ -median problem with a semi-Lagrangian relaxation. *Computational Optimization and Applications* 35, 239–260.
- Beltran-Royo, C., Vial, J.-P., Alonso-Ayuso, A., 2012. Semi-Lagrangian relaxation applied to the uncapacitated facility location problem. *Computational Optimization and Applications* 51, 387–409.
- Belvaux, G., Wolsey, L.A., 2000. bc-prod: a specialized branch-and-cut system for lot-sizing problems. *Management Science* 46, 724–738.

- Benders, J.F., 1962. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik* 4, 238–252.
- Bertsimas, D., Demir, R., 2002. An approximate dynamic programming approach to multidimensional knapsack problems. *Management Science* 48, 550–565.
- Bianco, L., Mingozzi, A., Ricciardelli, S., Spadoni, M., 1994. Exact and heuristic procedures for the traveling salesman problem with precedence constraints, based on dynamic programming. *INFOR: Information Systems and Operational Research* 32, 19–32.
- Bilde, O., Krarup, J., 1977. Sharp lower bounds and efficient algorithms for the simple plant location problem. *Annals of Discrete Mathematics* 1, 79–97.
- Billington, P.J., McClain, J.O., Thomas, L.J., 1986. Heuristics for multilevel lot-sizing with a bottleneck. *Management Science* 32, 989–1006.
- Bixby, R.E., Felton, M., Gu, Z., Rothberg, E., Wunderling, R., 2000. MIP: theory and practice—closing the gap. In Powell, M.J.D., Scholtes, S. (eds) *System Modelling and Optimization*. Springer, Boston, pp. 19–49.
- Blum, C., Pinacho, P., López-Ibáñez, M., Lozano, J.A., 2016. Construct, merge, solve & adapt: a new general algorithm for combinatorial optimization. *Computers & Operations Research* 68, 75–88.
- Boschetti, M.A., Maniezzo, V., 2015. A set covering based matheuristic for a real-world city logistics problem. *International Transactions in Operational Research* 22, 169–195.
- Boschetti, M.A., Maniezzo, V., 2022. Matheuristics: using mathematics for heuristic design. *4OR* 20, 173–208.
- Boschetti, M.A., Maniezzo, V., Roffilli, M., 2010. Decomposition techniques as metaheuristic frameworks. In Maniezzo, V., Stützle, T., Voß, S. (eds) *Matheuristics*. Springer, Boston, pp. 135–158.
- Boschetti, M.A., Mingozzi, A., Ricciardelli, S., 2004. An exact algorithm for the simplified multiple depot crew scheduling problem. *Annals of Operations Research* 127, 177–201.
- Boschetti, M.A., Mingozzi, A., Ricciardelli, S., 2008. A dual ascent procedure for the set partitioning problem. *Discrete Optimization* 5, 735–747.
- Brueggemann, T., Hurink, J.L., 2007. Two very large-scale neighborhoods for single machine scheduling. *OR Spectrum* 29, 513–533.
- Brueggemann, T., Hurink, J.L., 2011. Matching based very large-scale neighborhoods for parallel machine scheduling. *Journal of Heuristics* 17, 637–658.
- Burke, E.K., Curtois, T., 2014. New approaches to nurse rostering benchmark instances. *European Journal of Operational Research* 237, 71–81.
- Buson, E., Roberti, R., Toth, P., 2014. A reduced-cost iterated local search heuristic for the fixed-charge transportation problem. *Operations Research* 62, 1095–1106.
- Cabezas, X., García, S., 2022. A semi-Lagrangian relaxation heuristic algorithm for the simple plant location problem with order. *Journal of the Operational Research Society*. <https://doi.org/10.1080/01605682.2022.2150573>
- Cacchiani, V., Contreras-Bolton, C., Escobar-Falcón, L.M., Toth, P., 2023. A matheuristic algorithm for the pollution and energy minimization traveling salesman problems. *International Transactions in Operational Research* 30, 655–687.
- Caprara, A., 2002. Additive bounding, worst-case analysis, and the breakpoint median problem. *Siam Journal of Optimization* 13, 508–519.
- Caprara, A., Fischetti, M., Toth, P., 1999. A heuristic method for the set covering problem. *Operations Research* 47, 730–743.
- Captivo, M.E., 1991. Fast primal and dual heuristics for the  $p$ -median location problem. *European Journal of Operational Research* 52, 65–74.
- Carter, M.W., 1989. A Lagrangian relaxation approach to the classroom assignment problem. *INFOR: Information Systems and Operational Research* 27, 230–246.
- Caserta, M., Voß, S., 2012. A math-heuristic Dantzig-Wolfe algorithm for the capacitated lot sizing problem. In Hamadi, Y., Schoenauer, M. (eds) *Proceedings of the LION '12*. Springer, Heidelberg, pp. 31–41.
- Ceria, S., Nobili, P., Sassano, A., 1998. A Lagrangian-based heuristic for large-scale set covering problems. *Mathematical Programming* 81, 215–228.
- Chang, S.-G., Gavish, B., 1995. Lower bounding procedures for multiperiod telecommunications network expansion problems. *Operations Research* 43, 43–57.

- Chiarandini, M., Dumitrescu, I., Stützle, T., 2008. Very large-scale neighborhood search: overview and case studies on coloring problems. In Blum, C., Blesa Aguilera, M.J., Roli, A., Sampels, M. (eds) *Hybrid Metaheuristics: An Emerging Approach to Optimization*. Springer, Heidelberg, pp. 117–150.
- Chirayil Chandrasekharan, R., Smet, P., Wauters, T., 2021. An automatic constructive matheuristic for the shift minimization personnel task scheduling problem. *Journal of Heuristics* 27, 1, 205–227.
- Codato, G., Fischetti, M., 2006. Combinatorial Benders' cuts for mixed-integer linear programming. *Operations Research* 54, 756–766.
- Conforti, M., Cornuéjols, G., Zambelli, G., 2014. *Integer Programming*. Springer, Cham.
- Côté, J.-F., Dell'Amico, M., Iori, M., 2014. Combinatorial Benders' cuts for the strip packing problem. *Operations Research* 62, 643–661.
- Cunha, J.O., Kramer, H.H., Melo, R.A., 2019. Effective matheuristics for the multi-item capacitated lot-sizing problem with remanufacturing. *Computers & Operations Research* 104, 149–158.
- Danna, E., Rothberg, E., Pape, C., 2005. Exploring relaxation induced neighborhoods to improve MIP solutions. *Mathematical Programming* 102, 71–90.
- Dantzig, G.B., Wolfe, P., 1960. Decomposition principle for linear programs. *Operations Research* 8, 101–111.
- Daskin, M.S., Panayotopoulos, N.D., 1989. A Lagrangian relaxation approach to assigning aircraft to routes in hub and spoke networks. *Transportation Science* 23, 91–99.
- de Andrade, J.L.M., Menezes, G.C., 2023. A column generation-based heuristic to solve the integrated planning, scheduling, yard allocation and berth allocation problem in bulk ports. *Journal of Heuristics* 29, 39–76.
- De Franceschi, R., Fischetti, M., Toth, P., 2006. A new ILP-based refinement heuristic for vehicle routing problems. *Mathematical Programming* 105, 471–499.
- Djeumou Fomeni, F., Letchford, A.N., 2014. A dynamic programming heuristic for the quadratic knapsack problem. *INFORMS Journal on Computing* 26, 173–182.
- Dobson, G., Khosla, I., 1995. Simultaneous resource scheduling with batching to minimize weighted flow times. *IIE Transactions* 27, 587–598.
- Dokka, T., Letchford, A.N., Mansoor, M.H., 2021. On the complexity of surrogate and group relaxation for integer linear programs. *Operations Research Letters* 49, 530–534.
- Dokka, T., Letchford, A.N., Mansoor, M.H., 2021. Surrogate relaxation as a matheuristic. Technical report, Department of Management Science, Lancaster University.
- Dokka, T., Letchford, A.N., Mansoor, M.H., 2022. Revisiting surrogate relaxation for the multidimensional knapsack problem. *Operations Research Letters* 50, 674–678.
- Dupin, N., Parize, R., Talbi, E.-G., 2021. Matheuristics and column generation for a basic technician routing problem. *Algorithms* 14, 313–352.
- Echeverri, L.C., Froger, A., Mendoza, J.E., Neron, E., 2019. A matheuristic for the multi-period electric vehicle routing problem. In *13th Metaheuristics International Conference*. Cartagena de Indias, Colombia.
- Erlenkotter, D., 1978. A dual-based procedure for uncapacitated facility location. *Operations Research* 26, 992–1009.
- Faaland, B.H., Hillier, F.S., 1979. Interior path methods for heuristic integer programming procedures. *Operations Research* 27, 1069–1087.
- Fischetti, M., Fischetti, M., 2016. Matheuristics. In Martí, R., Panos, P., Resende, M.G.C. (eds) *Handbook of Heuristics*. Springer, Cham, pp. 1–33.
- Fischetti, M., Glover, F., Lodi, A., 2005. The feasibility pump. *Mathematical Programming* 104, 91–104.
- Fischetti, M., Lodi, A., 2003. Local branching. *Mathematical Programming* 98, 23–47.
- Fischetti, M., Lodi, A., Salvagnin, D., 2009. Just MIP it! In Maniezzo, V., Stützle, T., Voß, S. (eds) *Matheuristics, Hybridizing Metaheuristics and Mathematical Programming*. Springer, Boston, pp 39–70.
- Fischetti, M., Toth, P., 1989. An additive bounding procedure for combinatorial optimization problems. *Operations Research* 37, 319–328.
- Fischetti, M., Toth, P., 1992. An additive bounding procedure for the asymmetric travelling salesman problem. *Mathematical Programming* 53, 173–197.
- Fisher, M.L., 1976. A dual algorithm for the one-machine scheduling problem. *Mathematical Programming* 11, 229–251.
- Fisher, M.L., 1981. The Lagrangian relaxation method for solving integer programming problems. *Management Science* 27, 1–18.

- Fisher, M.L., 1985. An applications oriented guide to Lagrangian relaxation. *Interfaces* 15, 10–21.
- Fisher, M.L., Greenfield, A.J., Jaikumar, R., Lester, J.T., 1982. A computerized vehicle routing application. *Interfaces* 12, 42–52.
- Fisher, M.L., Jaikumar, R., 1981. A generalized assignment heuristic for vehicle routing. *Networks* 11, 109–124.
- Fisher, M.L., Jaikumar, R., Van Wassenhove, L.N., 1986. A multiplier adjustment method for the generalized assignment problem. *Management Science* 32, 1095–1103.
- Fisher, M.L., Kedia, P., 1990. Optimal solution of set covering/partitioning problems using dual heuristics. *Management Science* 36, 674–688.
- Fisher, M.L., Lageweg, B.J., Lenstra, J.K., Rinnooy Kan, A.H.G., 1983. Surrogate duality relaxation for job shop scheduling. *Discrete Applied Mathematics* 5, 65–75.
- Fisk, J.C., Hung, M.S., 1979. A heuristic routine for solving large loading problems. *Naval Research Logistics Quarterly* 26, 643–650.
- Frangioni, A., Necciarì, E., Scutellà, M.G., 2004. A multi-exchange neighborhood for minimum makespan parallel machine scheduling problems. *Journal of Combinatorial Optimization* 8, 195–220.
- Fréville, A., Plateau, G., 1986. Heuristics and reduction methods for multiple constraints 0–1 linear programming problems. *European Journal of Operational Research* 24, 206–215.
- Galvão, R.D., Acosta Espejo, L.G., Boffey, B., 2000. A comparison of Lagrangian and surrogate relaxations for the maximal covering location problem. *European Journal of Operational Research* 124, 377–389.
- Galvão, R.D., Raggi, L.A., 1989. A method for solving to optimality uncapacitated location problems. *Annals of Operations Research* 18, 225–244.
- Geoffrion, A.M., 1974. Lagrangian relaxation for integer programming. *Mathematical Programming Studies* 2, 82–114.
- Glover, F., 1975. Surrogate constraint duality in mathematical programming. *Operations Research* 23, 434–451.
- Glover, F., 1977. Heuristics for integer programming using surrogate constraints. *Decision Sciences* 8, 156–166.
- Glover, F., 1992. Ejection chains and combinatorial leverage for traveling salesman problems. Technical report, School of Business, University of Colorado, Boulder, CO.
- Glover, F., Karney, D., Klingman, D., 1979. A study of alternative relaxation approaches for a manpower planning problem. In Ijiri, Y., Whinston, A.B., Naddef, D. (eds) *Quantitative Planning and Control*. Academic Press, New York, pp. 141–164.
- Glover, F., Laguna, M., Martí, R., 2000. Fundamentals of scatter search and path relinking. *Control and Cybernetics* 29, 653–684.
- Gouveia, L., Paixao, J., 1991. Dynamic programming based heuristics for the topological design of local access networks. *Annals of Operations Research* 33, 305–327.
- Greenberg, H.J., Pierskalla, W.P., 1970. Surrogate mathematical programming. *Operations Research* 18, 924–939.
- Guastaroba, G., Savelsbergh, M.W.P., Speranza, M.G., 2017. Adaptive kernel search: a heuristic for solving mixed integer linear programs. *European Journal of Operational Research* 263, 789–804.
- Guastaroba, G., Speranza, M.G., 2012. Kernel search for the capacitated facility location problem. *Journal of Heuristics* 18, 877–917.
- Guignard, M., 1998. A Lagrangian dual ascent algorithm for simple plant location problems. *European Journal of Operational Research* 35, 193–200.
- Guignard, M., 2003. Lagrangian relaxation. *Trabajos de Operativa (TOP)* 11, 151–228.
- Guignard, M., Rosenwein, M.B., 1989. An application-oriented guide for designing Lagrangian dual ascent algorithms. *European Journal of Operational Research* 43, 197–205.
- Guignard, M., Rosenwein, M.B., 1989. An improved dual based algorithm for the generalized assignment problem. *Operations Research* 37, 658–663.
- Hahn, P., Grant, T., 1998. Lower bounds for the quadratic assignment problem based upon a dual formulation. *Operations Research* 46, 912–922.
- Hansen, P., Brimberg, J., Urošević, D., Mladenović, N., 2007. Primal-dual variable neighborhood search for the simple plant-location problem. *INFORMS Journal on Computing* 19, 552–564.
- Hillier, F.S., 1969. Efficient heuristic procedures for integer linear programming with an interior. *Operations Research* 17, 600–637.



- Hochbaum, D.S., 1982. Approximation algorithms for the set covering and vertex cover problems. *Siam Journal on Computing* 11, 555–556.
- Holmberg, K., Yuan, D., 2000. A Lagrangian heuristic based branch-and-bound approach for the capacitated network design problem. *Operations Research* 48, 461–481.
- Hribar, M., Daskin, M.S., 1997. A dynamic programming heuristic for the  $p$ -median problem. *European Journal of Operational Research* 101, 499–508.
- Janáček, J., Buzna, L., 2008. An acceleration of Erlenkotter–Körkel’s algorithms for the uncapacitated facility location problem. *Annals of Operations Research* 164, 97–109.
- Jörnsten, K., Klose, A., 2016. An improved Lagrangian relaxation and dual ascent approach to facility location problems. *Computational Management Science* 13, 317–348.
- Jörnsten, K., Näsberg, M., 1986. A new Lagrangian relaxation approach to the generalized assignment problem. *European Journal of Operational Research* 27, 313–323.
- Karwan, M.H., Rardin, R.L., 1984. Surrogate dual multiplier search procedures in integer programming. *Operations Research* 32, 52–69.
- Keaton, M.H., 1989. Designing optimal railroad operating plans: Lagrangian relaxation and heuristic approaches. *Transportation Research Part B: Methodological* 23, 415–431.
- Keaton, M.H., 1992. Designing railroad operating plans: a dual adjustment method for implementing Lagrangian relaxation. *Transportation Science* 26, 263–279.
- Kergosien, Y., Gendreau, M., Billaut, J.-C., 2017. A Benders decomposition-based heuristic for a production and out-bound distribution scheduling problem with strict delivery constraints. *European Journal of Operational Research* 262, 287–298.
- Kim, S.-L., Kim, S., 1998. Exact algorithm for the surrogate dual of an integer programming problem: subgradient method approach. *Journal of Optimization Theory and Applications* 96, 363–375.
- Klincwicz, J.G., 1996. A dual algorithm for the uncapacitated hub location problem. *Location Science* 4, 173–184.
- Körkel, M., 1989. On the exact solution of large-scale simple plant location problems. *European Journal of Operational Research* 39, 157–173.
- Korte, B., Vygen, J., 2018. *Combinatorial Optimization: Theory and Algorithms* (6th edn). Springer, Heidelberg.
- Lauer, G.S., Sandell, N.R., Bertsekas, D.P., Posbergh, T.A., 1982. Solution of large-scale optimal unit commitment problems. *IEEE Transactions on Power Apparatus and Systems* 1, 1, 79–86.
- Lemaréchal, C., 2001. Lagrangian relaxation. In Jünger, M., Naddef, D. (eds) *Computational Combinatorial Optimization: Optimal or Provably Near-Optimal Solutions*. Springer, Heidelberg, pp. 112–156.
- Letchford, A.N., Miller, S.J., 2012. Fast bounding procedures for large instances of the simple plant location problem. *Computers & Operations Research* 39, 985–990.
- Létocart, L., Plateau, M.-C., Plateau, G., 2014. An efficient hybrid heuristic method for the 0-1 exact  $k$ -item quadratic knapsack problem. *Pesquisa Operacional* 34, 49–72.
- Lim, S.-K., Kim, Y.-D., 1998. Capacity planning for phased implementation of flexible manufacturing systems under budget restrictions. *European Journal of Operational Research* 104, 175–186.
- Lin, F.Y.S., Yee, J.R., 1992. A new multiplier adjustment procedure for the distributed computation of routing assignments in virtual circuit data networks. *ORSA Journal of Computing* 4, 250–266.
- Litvinchev, I., Mata, M., Rangel, S., Saucedo, J., 2010. Lagrangian heuristic for a class of the generalized assignment problems. *Computers & Mathematics with Applications* 60, 1115–1123.
- Løkketangen, A., Glover, F., 1998. Solving zero-one mixed integer programming problems using tabu search. *European Journal of Operational Research* 106, 624–658.
- Lorena, L.A.N., Lopes, F.B., 1994. A surrogate heuristic for set covering problems. *European Journal of Operational Research* 79, 138–150.
- Lorena, L.A.N., Narciso, M.G., 1996. Relaxation heuristics for a generalized assignment problem. *European Journal of Operational Research* 91, 600–610.
- Maes, J., McClain, J.O., Van Wassenhove, L.N., 1991. Multilevel capacitated lotsizing complexity and LP-based heuristics. *European Journal of Operational Research* 53, 131–148.
- Magazine, M.J., Oguz, O., 1984. A heuristic algorithm for the multidimensional zero-one knapsack problem. *European Journal of Operational Research* 16, 319–326.

- Maher, S.J., 2021. Enhancing large neighbourhood search heuristics for Benders' decomposition. *Journal of Heuristics* 27, 615–648.
- Malandraki, C., Dial, R.B., 1996. A restricted dynamic programming heuristic algorithm for the time dependent traveling salesman problem. *European Journal of Operational Research* 90, 45–55.
- Maniezzo, V., Boschetti, M.A., Stützle, T., 2021. *Matheuristics: Algorithms and Implementations*. Springer, Cham.
- Maniezzo, V., Stützle, T., Voß, S. (eds), 2010. *Matheuristics: Hybridizing Metaheuristics and Mathematical Programming*. Springer, Boston.
- Mansini, R., Speranza, M.G., 1999. Heuristic algorithms for the portfolio selection problem with minimum transaction lots. *European Journal of Operational Research* 114, 219–233.
- Mao, K., Pan, Q.-K., Pang, X., Chai, T., 2014. A novel Lagrangian relaxation approach for a hybrid flowshop scheduling problem in the steelmaking-continuous casting process. *European Journal of Operational Research* 236, 51–60.
- Martí, R., Pardalos, P.M., Resende, M.G. (eds), 2018. *Handbook of Heuristics*. Springer, Cham.
- Martin-Iradi, B., Ropke, S., 2022. A column-generation-based matheuristic for periodic and symmetric train timetabling with integrated passenger routing. *European Journal of Operational Research* 297, 511–531.
- Masmoudi, M., Adouani, Y., Jarboui, B., 2023. LP relaxation and dynamic programming enhancing VNS for the multiple knapsack problem with setup. *International Transactions in Operational Research*. <https://doi.org/10.1111/itor.13213>
- Meyers, C., Orlin, J.B., 2007. Very large-scale neighborhood search techniques in timetabling problems. In Burke, E.K., Rudová, H. (eds) *Proceedings of the PATAT '06*. Springer, Heidelberg, pp. 24–39.
- Mingozzi, A., Maniezzo, V., Ricciardelli, S., Bianco, L., 1998. An exact algorithm for the resource-constrained project scheduling problem based on a new mathematical formulation. *Management Science* 44, 714–729.
- Mitrović-Minić, S., Punnen, A.P., 2008. Very large-scale variable neighborhood search for the generalized assignment problem. *Journal of Interdisciplinary Mathematics* 11, 653–670.
- Mitrović-Minić, S., Punnen, A.P., 2009. Local search intensified: very large-scale variable neighborhood search for the multi-resource generalized assignment problem. *Discrete Optimization* 6, 370–377.
- Monabbati, E., 2014. An application of a Lagrangian-type relaxation for the uncapacitated facility location problem. *Japan Journal of Industrial and Applied Mathematics* 31, 483–499.
- Monaco, M.F., Sammarra, M., 2022. A multiplier adjustment algorithm for a truck scheduling and transshipment problem at a cross-docking terminal. *Soft Computing* 27, 4169–4193.
- Murthy, I., 1993. Solving the multiperiod assignment problem with start-up costs using dual ascent. *Naval Research Logistics* 40, 325–344.
- Neebe, A.W., 1987. An improved, multiplier adjustment procedure for the segregated storage problem. *Journal of the Operational Research Society* 38, 815–825.
- Oliveira, Ó., Matos, T., Gamboa, D., 2021. A dual RAMP algorithm for single source capacitated facility location problems. *Annals of Mathematics and Artificial Intelligence* 89, 815–834.
- Pirkul, H., 1987. Efficient algorithms for the capacitated concentrator location problem. *Computers & Operations Research* 14, 197–208.
- Pirkul, H., 1987. A heuristic solution procedure for the multiconstraint zero-one knapsack problem. *Naval Research Logistics* 34, 161–172.
- Polzin, T., Daneshmand, S.V., 2001. Improved algorithms for the Steiner problem in networks. *Discrete Applied Mathematics* 112, 263–300.
- Posta, M., Ferland, J.A., Michelon, P., 2014. An exact cooperative method for the uncapacitated facility location problem. *Mathematical Programming Computation* 6, 199–231.
- Potvin, J.-Y., Gendreau, M., 2019. *Handbook of Metaheuristics* (3rd edn). Springer, Cham.
- Raghavan, P., Thompson, C.D., 1987. Randomized rounding: a technique for provably good algorithms and algorithmic proofs. *Combinatorica* 7, 365–374.
- Raidl, G.R., 2015. Decomposition based hybrid metaheuristics. *European Journal of Operational Research* 244, 66–76.
- Raidl, G.R., Puchinger, J., 2008. Combining (integer) linear programming techniques and metaheuristics for combinatorial optimization. In Blum, C., Aguilera, M.J.B., Roli, A., Sampels, M. (eds) *Hybrid Metaheuristics: An Emerging Approach to Optimization*. Springer, Heidelberg, pp. 31–62.

- Rego, C., 2005. RAMP: a new metaheuristic framework for combinatorial optimization. In Sharda, R., Voß, S., Rego, C., Alidaee, B. (eds) *Metaheuristic Optimization via Memory and Evolution*. Springer, Boston, pp. 441–460.
- Rego, C., Mathew, F., Glover, F., 2010. RAMP for the capacitated minimum spanning tree problem. *Annals of Operations Research* 181, 661–681.
- Riley, R.C.L., Rego, C., 2019. Intensification, diversification, learning via relaxation adaptive memory programming: a case study on resource constrained project scheduling. *Journal of Heuristics* 25, 793–807.
- Ropke, S., Pisinger, D., 2006. An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science* 40, 455–472.
- Saberi, H., Amraee, T., Zhang, C., Zhao, Y.D., 2020. A heuristic Benders-decomposition-based algorithm for transient stability constrained optimal power flow. *Electric Power Systems Research* 185, 106380.
- Sadykov, R., Vanderbeck, F., Pessoa, A., Tahiri, I., Uchoa, E., 2019. Primal heuristics for branch and price: the assets of diving methods. *INFORMS Journal on Computing* 31, 251–267.
- Sarvanov, V.I., Doroshko, N.N., 1981. Approximate solution of the traveling salesman problem by a local algorithm with scanning neighbourhoods of factorial cardinality in cubic time (in Russian). *Software, Algorithms and Programs* 31, 11–13.
- Scheithauer, G., Terno, J., 1996. The G4-heuristic for the pallet loading problem. *Journal of the Operational Research Society* 47, 511–522.
- Shaw, P., 1998. Using constraint programming and local search methods to solve vehicle routing problems. In Maher, M., Puget, J.-F. (eds) *Proceedings of the CP '98*. Springer, Heidelberg, pp. 417–431.
- Sniedovich, M., Voß, S., 2006. The corridor method: a dynamic programming inspired metaheuristic. *Control and Cybernetics* 35, 551–578.
- Studel, H.J., Ballakur, A., 1987. A dynamic programming based heuristic for machine grouping in manufacturing cell formation. *Computers & Industrial Engineering* 12, 215–222.
- Stützle, T., Maniezzo, V., 2020. Preface to the special issue on matheuristics and metaheuristics. *International Transactions in Operational Research* 27, 5–8.
- Tanaka, S., 2012. A heuristic algorithm based on Lagrangian relaxation for the closest string problem. *Computers & Operations Research* 39, 709–717.
- Thompson, G.M., 1990. Shift scheduling in services when employees have limited availability: an LP approach. *Journal of Operations Management* 9, 352–370.
- Thompson, P.M., Psaraftis, H.N., 1993. Cyclic transfer algorithm for multivehicle routing and scheduling problems. *Operations Research* 41, 935–946.
- Trigeiro, W.W., 1987. A dual-cost heuristic for the capacitated lot sizing problem. *IIE Transactions* 19, 67–72.
- Van Roy, T.J., Gelders, L.F., 1981. Solving a distribution problem with side constraints. *European Journal of Operational Research* 6, 61–66.
- Vanderbei, R.J., 2020. *Linear Programming* (5th edn). Springer, Cham.
- Vazirani, V.V., 2001. *Approximation Algorithms*. Springer, Heidelberg.
- Vigo, D., 1996. A heuristic algorithm for the asymmetric capacitated vehicle routing problem. *European Journal of Operational Research* 89, 108–126.
- Wedelin, D., 1995. An algorithm for large scale 0–1 integer programming with application to airline crew scheduling. *Annals of Operations Research* 57, 283–301.
- Williamson, D.P., Shmoys, D.B., 2011. *The Design of Approximation Algorithms*. Cambridge University Press, New York.
- Wolsey, L.A., 1998. *Integer Programming*. Wiley, Chichester.
- Wong, R.T., 1984. A dual ascent approach for Steiner tree problems on a directed graph. *Mathematical Programming* 28, 271–287.
- Wright, D.D., von Lanzanauer, C.H., 1989. Solving the fixed charge problem with Lagrangian relaxation and cost allocation heuristics. *European Journal of Operational Research* 42, 305–312.
- Zhang, H., Beltran-Royo, C., Wang, B., Ma, L., Zhang, Z., 2016. Solution to the quadratic assignment problem using semi-Lagrangian relaxation. *Journal of Systems Engineering and Electronics* 27, 1063–1072.
- Zhang, H., Beltran-Royo, C., Wang, B., Zhang, Z., 2019. Two-phase semi-Lagrangian relaxation for solving the uncapacitated distribution centers location problem for B2C E-commerce. *Computational Optimization and Applications* 72, 827–848.