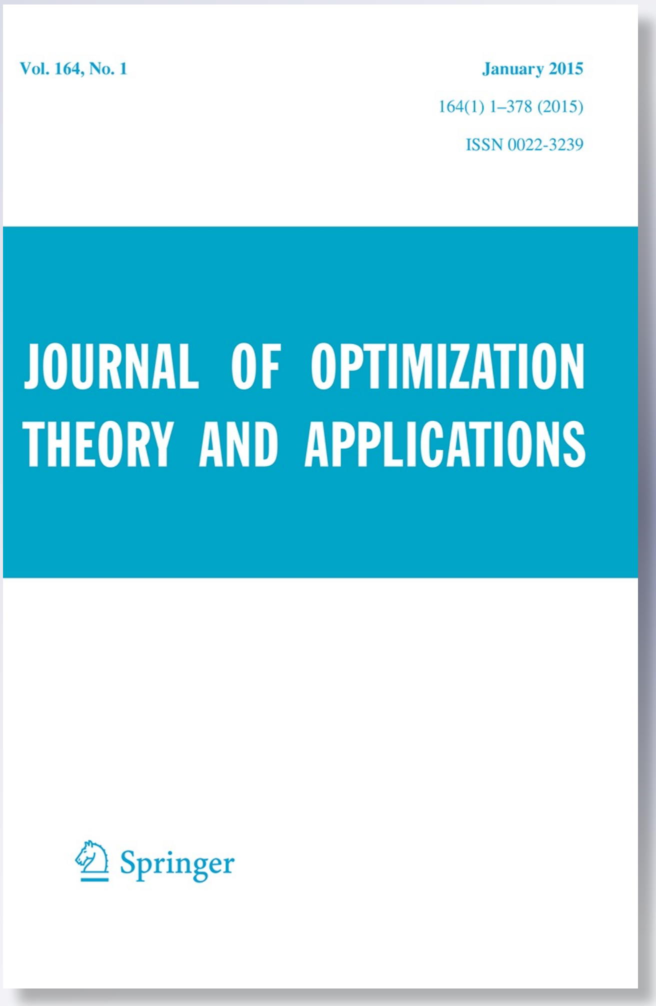
Convergence of the Surrogate Lagrangian Relaxation Method

Mikhail A. Bragin, Peter B. Luh, Joseph H. Yan, Nanpeng Yu & Gary A. Stern

Journal of Optimization Theory and Applications

ISSN 0022-3239
Volume 164
Number 1

J Optim Theory Appl (2015) 164:173-201
DOI 10.1007/s10957-014-0561-3



Springer

J Optim Theory Appl (2015) 164:173–201
DOI 10.1007/s10957-014-0561-3

Convergence of the Surrogate Lagrangian Relaxation Method

Mikhail A. Bragin · Peter B. Luh · Joseph H. Yan · Nanpeng Yu · Gary A. Stern

Received: 22 April 2013 / Accepted: 20 March 2014 / Published online: 8 April 2014

Abstract Studies have shown that the surrogate subgradient method, to optimize non-smooth dual functions within the Lagrangian relaxation framework, can lead to significant computational improvements as compared to the subgradient method. The key idea is to obtain surrogate subgradient directions that form acute angles toward the optimal multipliers without fully minimizing the relaxed problem. The major difficulty of the method is its convergence, since the convergence proof and the practical implementation require the knowledge of the optimal dual value. Adaptive estimations of the optimal dual value may lead to divergence and the loss of the lower bound property for surrogate dual values. The main contribution of this paper is on the development of the surrogate Lagrangian relaxation method and its convergence proof to the optimal multipliers, without the knowledge of the optimal dual value and without fully optimizing the relaxed problem. Moreover, for practical implementations, a stepsizing formula that guarantees convergence without requiring the optimal dual value has been constructively developed. The key idea is to select stepsizes in a way that distances between Lagrange multipliers at consecutive iterations decrease, and as a result, Lagrange multipliers converge to a unique limit. At the same time, stepsizes are kept sufficiently large so that the algorithm does not terminate prematurely. At convergence, the lower-bound property of the surrogate dual is guaranteed. Testing results demonstrate that non-smooth dual functions can be efficiently optimized, and the new method leads to faster convergence as compared to other methods available

Communicated by Fabián Flores-Bazàn.

M. A. Bragin (✉)· P. B. Luh
Department of Electrical and Computer Engineering, University of Connecticut,
Storrs, CT 06269-2157, USA
e-mail: mab08017@engr.uconn.edu

J. H. Yan · N. Yu · G. A. Stern
Southern California Edison, Rosemead, CA 91770, USA

for optimizing non-smooth dual functions, namely, the simple subgradient method, the subgradient-level method, and the incremental subgradient method.

Keywords Non-smooth optimization · Subgradient methods · Surrogate subgradient method · Lagrangian relaxation · Mixed-integer programming

Mathematics Subject Classification (2000) 90C25

1 Introduction

When solving complicated mixed-integer optimization problems, the effort needed to obtain an optimal solution increases dramatically as the problem size increases. Therefore, the goal for practical applications is often to obtain a near-optimal solution with quantifiable quality in a computationally efficient manner. Lagrangian relaxation has successfully achieved this goal by exploiting separability of a problem. In the method, the relaxed problem is fully optimized, and the dual function is obtained. Dual functions are always concave, and the feasible set of dual solutions is always convex regardless of the characteristics of the original problem such as convexity. To optimize non-smooth concave dual functions, Lagrange multipliers are adjusted based on appropriately defined stepsizes and by using subgradient directions [1–13] or surrogate subgradient directions [14–20]. At convergence of multipliers, heuristics are typically used to obtain feasible solutions.[1]

The subgradient method has been extensively studied starting with the pioneering works of Ermoliev [1], Polyak [2,3], and Shor [4,5]. The general convergence has been established in [1] and [2]. To ensure convergence with a geometric rate, convergence was proved by requiring the optimal dual value [3]. In practical implementations, adaptive rules to adjust estimates of the optimal dual value were first developed in [3], and such rules have been improved in [7–9,11,21] to guarantee convergence to the optimum. Difficulties associated with the unavailability of the optimal dual value have been overcome owing to the fact that subgradients form acute angles with directions toward the optimal multipliers and owing to the convexity of the dual function. Therefore, for properly chosen stepsizes, multipliers move closer to the optimal multipliers. For example, in the subgradient-level method [7], stepsizes are set by using estimates of the optimal dual value based on the highest dual value obtained so far, and such estimates are further adjusted when significant oscillations of multipliers are detected. However, subgradient methods require the relaxed problem to be fully optimized, which can be difficult when the relaxed problem is non-separable or NP-hard. Moreover, convergence can be slow because multipliers often zigzag across the ridges of the dual function, and the zigzagging is especially noticeable when ridges are sharp. While incremental subgradient methods [8] reduce computational requirements by optimizing one subproblem at a time and converge without requiring the optimal

[1] While Augmented Lagrangian relaxation has been a powerful method and can alleviate zigzagging, thereby reducing computational requirements, it is generally not used to optimize dual functions. Furthermore, the extra quadratic term makes the problem non-separable. Although methods were developed to overcome the resulting non-separability issue, they were not very effective.

dual value, these methods require separability of the problem and cannot be used to solve non-separable problems, or when subproblems are NP-hard.

The surrogate subgradient method, developed within the Lagrangian relaxation framework, is a variation of the subgradient method that seeks to reduce computational requirements and to obtain surrogate subgradient directions that form acute angles with directions toward to the optimal multipliers [15–20]. Without requiring the relaxed problem to be fully optimized, surrogate subgradient directions do not change drastically, thereby alleviating the zigzagging of multipliers and reducing the number of iterations required for convergence. The major difficulty of the method is its convergence, since the convergence proof and the practical implementation require the knowledge of the optimal dual value, which is unavailable in practice. In the method, since the relaxed problem is not fully optimized, surrogate dual values are no longer on but are above the dual surface. As a result, surrogate subgradient directions may not form acute angles with directions toward the optimal multipliers, and divergence may occur. In addition, the lower bound property of surrogate dual values may be lost. While such difficulties can sometimes be overcome by occasionally obtaining subgradients during the convergence process, the computational effort can still be prohibitive when the relaxed problem is non-separable or NP-hard.

In this paper, surrogate Lagrangian relaxation with novel conditions on stepsizes is developed, and convergence of the method is proved without requiring the optimal dual value and without fully optimizing the relaxed problem in Sect. 2. The idea is to select stepsizes in a way that distances between Lagrange multipliers at consecutive iterations decrease, and as a result, multipliers converge to a unique limit. At the same time, stepsizes are kept sufficiently large so that the algorithm does not terminate prematurely. At convergence, a surrogate dual value provides a lower bound to the primal cost. Moreover, a particular stepsizing formula that satisfies the set of conditions has been obtained. Convergence of the interleaved method [14], in which one subproblem is solved at a time to update multipliers, has also been proved. Under additional assumptions used in subgradient methods [8], the convergence rate of the new method is linear.

Section 3 presents testing results for a small nonlinear integer optimization problem, large generalized assignment problems, and quadratic assignment problems. For the small problem, the new method is compared with the subgradient method to demonstrate that the zigzagging is alleviated, and calculations of surrogate subgradient directions require significantly lower computational effort. The new method is then compared with the methods available for non-smooth optimization such as the simple subgradient method, the subgradient-level method and the incremental subgradient method when solving generalized assignment problems with separable dual problems, and quadratic assignment problems with non-separable dual problems.

2 Convergence of the Surrogate Lagrangian Relaxation Method

This section is on the development of the novel surrogate Lagrangian relaxation method and its convergence proof without requiring the optimal dual value. In Sect. 2.1, a generic mixed-integer problem formulation and the Lagrangian relaxation framework are presented. To maximize non-smooth dual functions, subgradient directions and stepsizes requiring the optimal dual value are frequently used [22]. To find multiplier-updating directions that form acute angles with directions toward the optimal multipliers without fully minimizing the relaxed problem, the Lagrangian relaxation, and surrogate subgradient method [15] is presented next. In Sect. 2.2, the surrogate Lagrangian relaxation method is developed, and convergence of the method is proved without requiring the optimal dual value. Convergence rate of the method is discussed in Sect. 2.3. Section 2.4 discusses practical implementation aspects of the algorithm such as a constructive stepsize-setting procedure.

2.1 Mixed-Integer Programming and Lagrangian Relaxation

Consider a mixed-integer problem formulation:

$$\min_{x} \; f(x), \; \text{subject to}\; g(x) \le 0, x \in X, \tag{1}$$

where $x = (y, z)$, $y \in \mathbb{R}^{N_r}$, $z \in \mathbb{Z}^{N_z}$, and $X \subseteq \mathbb{R}^{N_r} \times \mathbb{Z}^{N_z}$, with $\mathbb{R}$ denoting the set of real numbers, $\mathbb{Z}$ denoting the set of integers, $f : X \to \mathbb{R}$ and $g : X \to \mathbb{R}^m$ are continuous and differentiable with respect to y. In addition, $g(x)$ satisfy the following assumptions:

Assumption 2.1 There exists a scalar M such that

$$\|g(x)\| < M < \infty, \forall x \in X. \tag{2}$$

Assumption 2.2 Regularity Condition Gradient vectors of active inequality constraints with respect to y are linearly independent at a constrained local minimum $x^* = (y^*, z^*)$ of $f(x)$. □

The regularity Assumption 2.2 is needed only in the continuous subspace $\mathbb{R}^{N_r}$ to rule out possible irregularities, such as linear dependence of gradients of active constraints. In the discrete subspace $\mathbb{Z}^{N_z}$, regularity conditions are not needed [23].

When solving discrete optimization problems, the Lagrangian relaxation method has been used [15,22] and shown to be especially powerful for solving separable programming problems. In the method, the constraints of (1) are relaxed, and the Lagrangian function is formed by introducing a vector of Lagrange multipliers $\lambda^T = (\lambda_1, \ldots, \lambda_m) \in \mathbb{R}^m$:

$$L(\lambda, x) := f(x) + \lambda^T g(x). \tag{3}$$

The dual function, resulting from the minimization of the Lagrangian function (3), becomes

$$q(\lambda) := \min_{x \in X} L(\lambda, x), \tag{4}$$

and the dual problem is to maximize the concave non-smooth dual function [22]:

$$\max_{\lambda} q(\lambda)\,,\, s.t.\, \lambda \in \mathbb{R}^m, \lambda \geq 0. \tag{5}$$

When the original problem is integer or mixed-integer linear, the dual function is polyhedral concave and non-smooth [15]. In the subgradient method, to maximize the dual function, multipliers are updated according to:

$$\lambda^{k+1} = [\lambda^k + c^k g(x^k)]^+,\ \ k = 0,\ 1,\ \ldots, \tag{6}$$

where $[]^+$ denotes projection onto the positive orthant, $g(x^k)$ is the subgradient of the dual function $q(\lambda)$ at λ^k, and c^k is a positive scalar stepsize. If equality constraints $h(x) = 0$ are present in the formulation, multipliers are updated according to (6) without projecting onto the positive orthant.

Since $q(\lambda)$ is convex, dual values are not greater than the optimal dual value $q^* := q(\lambda^*)$

$$q(\lambda^k) \leq q^*. \tag{7}$$

Moreover, by the definition of subgradients, the following relationship holds:

$$q^* - q(\lambda^k) \leq (\lambda^* - \lambda^k)^T g(x^k). \tag{8}$$

Both sides of the inequality (8) are non-negative owing to the inequality (7). Therefore, subgradient directions from acute angles with directions toward λ^*, and distances between the current multipliers and the optimum λ^* can be decreased under the following condition on stepsizes [22]:

$$0 < c^k < \frac{2(q^* - q(\lambda^k))}{\left\| g(x^k) \right\|^2},\ \ k = 0,\ 1,\ \ldots. \tag{9}$$

While q^* is unknown in practice, significant research has been done to guarantee convergence to λ^* by adaptively estimating q^*. For example, in the subgradient-level method [7], estimates of q^* can be adaptively adjusted based on such criteria as a sufficient ascent of a dual value or significant oscillation of the multipliers.

While subgradient directions are traditionally used to update multipliers, such directions may almost be perpendicular to directions toward λ^*, thereby leading to slow convergence. Moreover, while optimization of the relaxed problem (4) is generally simpler than the optimization of the original problem (1), it can still be difficult when the relaxed problem is non-separable and NP-hard. This usually leads to difficulties of fully optimizing the relaxed problem (4) and computing corresponding subgradient directions of the dual function (4). Therefore, it is desirable to obtain multiplier-updating directions that form acute angles with directions toward λ^* in a computationally efficient manner and to show that multipliers are moving closer to λ^*.

To reduce computational requirements by not requiring the relaxed problem to be fully optimized, the Lagrangian relaxation and surrogate subgradient method has been developed in [15] for separable integer programming problems under the assumption that the constraint functions are $g(x) = Ax - b$. For our problem (1) under consideration, for any feasible solution $x^k \in X$ of the relaxed problem, the surrogate dual is defined following [15] as:

$$\tilde{L}(\lambda^k, x^k) := f(x^k) + (\lambda^k)^T \tilde{g}(x^k), \tag{10}$$

where

$$\tilde{g}(x^k) := g(x^k) \tag{11}$$

is the surrogate subgradient direction.

Since the relaxed problem is not fully optimized, surrogate dual values are generally above the dual surface and can be larger than q^*, thereby causing the violation of (7). As a result, surrogate subgradient directions may not form acute angles with directions toward λ^*, and divergence may occur.

To guarantee that surrogate subgradient directions form acute angles with directions toward λ^*, the relaxed problem has to be sufficiently optimized, such that surrogate dual values (10) satisfy the following *surrogate optimality condition*:

$$\tilde{L}(\lambda^k, x^k) < \tilde{L}(\lambda^k, x^{k-1}), \tag{12}$$

and stepsizes have to be sufficiently small

$$0 < c^k < \frac{q^* - \tilde{L}(\lambda^k, x^k)}{\left\| \tilde{g}(x^k) \right\|^2}, \; k = 0, 1, \ldots. \tag{13}$$

Under the assumption that constraints are $g(x) = Ax - b$, it has been proved in [15,17] that multipliers move closer to λ^*at every iteration when updated recursively:

$$\hat{\lambda}^{k+1} = \lambda^k + c^k \tilde{g}(x^k), \; k = 0, 1, \ldots \tag{14}$$

$$\lambda^{k+1} = [\hat{\lambda}^{k+1}]^+, \; k = 0, 1, \ldots \tag{15}$$

where x^k satisfy (12), and c^k satisfy (13).

In addition, it has been shown that the lower-bound property of a surrogate dual function is preserved

$$\tilde{L}(\lambda^k, x^k) < q^*, \; k = 0, 1, \ldots. \tag{16}$$

While convergence was proved [15,17] when the constraints are $g(x) = Ax - b$, the proof in [15,17] does not use or require linearity of $g(x)$ to establish convergence. Therefore, for general constraints $g(x)$ considered in our paper under the regularity condition of Assumption 2.2, multipliers converge to λ^*, and the lower bound property

(16) is preserved if multipliers are updated according to (14)–(15), and stepsizes satisfy (13).

When the original problem (1) is separable, that is, the objective function $f(x)$ and constraints $g(x)$ are of an additive form, the relaxed problem (4) can be separated into N_s individual subproblems. Within the surrogate subgradient framework, it is sufficient to optimize the relaxed problem (4) with respect to several subproblems ($< N_s$), subject to the surrogate optimality condition (12), to obtain surrogate subgradient directions. The accompanying computational effort is approximately $1/N_s$ per subproblem of the effort required to obtain subgradient directions.

When the original problem (1) is non-separable or difficult to decompose into individual subproblems, the relaxed problem (4), subject to the surrogate optimality condition (12), can also be optimized with a sizable efficiency gain as compared to the subgradient method to obtain surrogate subgradient directions by optimizing the relaxed problem with respect to selected decision variables, while keeping other decision variables fixed.

The major difficulty of the surrogate subgradient method is its convergence, since the upper bound on stepsizes (13) cannot be specified due to the unavailability of q^*. In practical implementations, estimates of q^* may violate (13), thereby leading to divergence.

2.2 The Surrogate Lagrangian Relaxation Method

In this section, the main theoretical contribution of this paper, a new method is developed, and convergence to λ^* is proved without requiring q^*. In the method, the surrogate optimality condition (12) and the multiplier-updating formulas (14) and (15) will be used. To prove and guarantee convergence without requiring q^*, instead of the stepsizing formula (13), a new formula to set stepsizes will be developed, and convergence of multipliers (14)–(15) to λ^* will be proved. In addition, it will be proved that an interleaved method [14] with the new stepsizing formula also converges to λ^*.

The main idea is to obtain stepsizes such that distances between multipliers[2] at consecutive iterations decrease, i.e.,

$$\left\| \hat{\lambda}^{k+1} - \lambda^k \right\| = \alpha_k \left\| \hat{\lambda}^k - \lambda^{k-1} \right\|, \ 0 < \alpha_k < 1, \ k = 1, 2, \ldots. \tag{17}$$

The stepsizing formula satisfying (17) can be derived by using (14). Indeed, (14) and (17) imply

$$\left\| c^k \tilde{g}(x^k) \right\| = \alpha_k \left\| c^{k-1} \tilde{g}(x^{k-1}) \right\|, \ 0 < \alpha_k < 1, \ k = 1, 2, \ \ldots. \tag{18}$$

[2] Strictly speaking, when dealing with inequality constraints $g(x) \leq 0$, distances between multipliers and projections of multiples from the previous iteration are considered rather than distances between multipliers.

In the new method, stepsizes c^k satisfying (18) can always be uniquely obtained, unless norms of surrogate subgradients are zero.[3] Therefore, norms of surrogate subgradients are subject to a strict positivity requirement:

$$\left\| \tilde{g}(x^k) \right\| > 0. \tag{19}$$

Since c^k and c^{k-1} are positive scalars,[4] and norms of surrogate subgradients are strictly positive, (18) implies

$$c^k = \alpha_k \frac{c^{k-1} \left\| \tilde{g}(x^{k-1}) \right\|}{\left\| \tilde{g}(x^k) \right\|}, \ \ 0 < \alpha_k < 1, \ \ k = 1, 2, \ldots. \tag{20}$$

The combined multiplier-updating formula (14)–(15) and (17) can be viewed as a mapping from $\hat{\lambda}^k (\in R^m)$ to $\hat{\lambda}^{k+1} (\in R^m)$. Since the distances between multipliers at consecutive iterations always strictly decrease per (17), multipliers converge to a limit, and stepsizes approach zero. When $\{\alpha_k\}$ are too small, however, stepsizes can approach 0 too fast, and the algorithm may terminate prematurely. To avoid that, stepsizes (20) should be kept sufficiently large, and this can be achieved by keeping α_k sufficiently close to 1 as proved in the following theorem, the main result of this paper.

Theorem 2.1 *Suppose that multiplier-updating directions satisfy the conditions (12) and (19), and constraints of (1) satisfy the regularity condition of Assumption 2.2. If α_k satisfies the following conditions:*

$$\prod_{i=1}^{k} \alpha_i \to 0, \tag{21a}$$

and

$$\lim_{k \to \infty} \frac{1 - \alpha_k}{c^k} = 0, \tag{21b}$$

then the mapping (14)-(15), with c^k satisfying (20) has a unique fixed point λ^*. □

This theorem is proved in three stages. In Stage 1, convergence to a unique fixed point (not necessarily λ^*) is proved under the condition (21a). In Stage 2, convergence to λ^* is proved by temporarily using q^* to establish a lower bound on stepsizes. In Stage 3, the proof is completed with an additional asymptotical condition (21b) without requiring q^*.

---

[3] In the subgradient method, zero-subgradient implies that the optimum is obtained, and the algorithm terminates with the optimal primal solution. In the surrogate subgradient method, zero-surrogate subgradient implies that only a feasible solution is obtained, and the algorithm must proceed.

[4] Initial stepsize c^0 is initialized to be a positive scalar, therefore, stepsizes c^k, $k = 1, 2, \ldots$ satisfying (18) are positive.

Proposition 2.2 *With the stepsizing formula (20), the Lagrange multipliers (14)-(15) converge to a unique fixed point* $\bar{\lambda} \equiv \lim_{k\to\infty} \lambda^k$ *(not necessarily to* λ^**), provided (21a) and the norm positivity requirement (19) hold.* □

Proof From (20) it follows that

$$c^k = \prod_{i=1}^{k} \alpha_i \frac{c^0 \left\| \tilde{g}(x^0) \right\|}{\left\| \tilde{g}(x^k) \right\|}, \quad k = 1,\ 2,\ \ldots. \tag{22}$$

Then by using (14) and (22), we get

$$\left\| \hat{\lambda}^{k+1} - \lambda^k \right\| = c^0 \left\| \tilde{g}(x^0) \right\| \prod_{i=1}^{k} \alpha_i. \tag{23}$$

Since projections are non-expansive, (23) can be written as an inequality

$$\left\| \lambda^{k+1} - \lambda^k \right\| \le c^0 \left\| \tilde{g}(x^0) \right\| \prod_{i=1}^{k} \alpha_i. \tag{24}$$

Since (21a) holds, $\left\| \lambda^{k+1} - \lambda^k \right\|$ approach zero.

To prove that multipliers converge to a unique fixed point, it will be proved that surrogate dual values approach dual values as $\left\| \lambda^{k+1} - \lambda^k \right\|$ become small for a sufficiently large iteration $k = L$. After that, the proof uses an argument similar to convergence results of the subgradient method with a diminishing stepsize rule as in [1,2,8,9,22].

To prove that surrogate dual values approach dual values, consider an arbitrary and a fixed value of multipliers λ at an arbitrary iteration M. A series of surrogate optimizations for the fixed value of λ, subject to the surrogate optimality condition (12), consecutively finds solutions $x^{M+1}, x^{M+2}, \ldots$ that satisfy

$$q(\lambda) < \ldots < \tilde{L}(\lambda, x^{M+2}) < \tilde{L}(\lambda, x^{M+1}) < \tilde{L}(\lambda, x^{M}), \tag{25}$$

until a dual value $q(\lambda)$ is reached. Given the discrete nature of the original problem (1), only a finite number of iterations in (25) is required to reach $q(\lambda)$, and $\tilde{L}(\lambda, x^{M+k_0}) = q(\lambda)$ for a positive number k_0. For example, when a problem has N_s subproblems, and one subproblem is optimized at a time, then $q(\lambda)$ is obtained within at most $k_0 = N_s$ iterations.

Following the same logic, when $\left\| \lambda^{k+1} - \lambda^k \right\|$ are sufficiently small, surrogate subgradient directions approach subgradient directions. Indeed, since $\left\| \lambda^{k+1} - \lambda^k \right\|$ in (24) converge to zero, there exists an iteration L and a positive finite number l such that the distance between λ^L and λ^{L+l} is sufficiently small such that values $q(\lambda^L)$ and $q(\lambda^{L+l})$ belong to the same facet of the dual function $q(\lambda)$. As in (25), starting from an iteration L, a surrogate dual value $\tilde{L}(\lambda^L, x^L)$ converges to a dual value $q(\lambda^{L+l})$ within a finite number of iterations l. Therefore, starting from iteration $L+l$, surrogate subgradient directions become subgradient directions. In the subgradient method with stepsizes approaching zero, multipliers converge to a fixed point: $\bar{\lambda} = \lim_{k\to\infty} \lambda^k$. □

The condition (21a) alone is not sufficient to guarantee convergence to λ^*, since stepsizes may approach 0 fast, thereby leading to a premature algorithm termination. To avoid that, stepsizes will be kept sufficient large by temporarily introducing q^*.

Proposition 2.3 Sufficient Condition for Convergence to λ^* *With the stepsizing formula (20), condition (21a), and the norm positivity requirement (19), the Lagrange multipliers (14)-(15) converge to λ^* if there exist $\kappa > k$ for all k and stepsizes satisfy the following lower-bound condition:*

$$\frac{q^* - \tilde{L}(\lambda^\kappa, x^\kappa)}{\|\tilde{g}(x^\kappa)\|^2} \leq c^\kappa. \tag{26}$$

Proof To prove that the multipliers $\bar{\lambda}$ are optimal when stepsizes c^k satisfy conditions (20) and (26), and stepsize-setting parameters α^k satisfy (21a), the following equality is to be established

$$q(\bar{\lambda}) = q^*. \tag{27}$$

The lower-bound condition on stepsizes (26) leads to

$$q^* - \tilde{L}(\lambda^\kappa, x^\kappa) \leq c^\kappa \left\| \tilde{g}(x^\kappa) \right\|^2. \tag{28}$$

Conditions (21a) and (22) imply $c^k \to 0$. Since $\kappa > k$, then $c^\kappa \to 0$ as $k \to \infty$, and inequality (28) yields

$$q^* - \tilde{L}(\lambda^\kappa, x^\kappa) \leq 0. \tag{29}$$

According to Proposition 2.2, $\bar{\lambda} = \lim\limits_{k\to\infty} \lambda^k$. Since $\kappa > k$, then $\bar{\lambda} = \lim\limits_{\kappa\to\infty} \lambda^\kappa$ implies

$$q^* - \tilde{L}(\bar{\lambda}, x^\kappa) \leq 0. \tag{30}$$

From the inequality (25) it follows that $\tilde{L}(\bar{\lambda}, x^\kappa) \to q(\bar{\lambda})$. Therefore, by using (25) and (30) we get

$$q^* - q(\bar{\lambda}) \leq 0. \tag{31}$$

Therefore, $\bar{\lambda}$ maximizes the dual function, and $\bar{\lambda}$ is an optimum. □

Theorem 2.1 will now be proved by contradiction by using condition (21b) without requiring q^*. It will be shown that a condition contrary to (26) does not hold under condition (21b), thereby proving that multipliers converge to λ^*.

Proof of Theorem 2.1.

Proof The formal proof follows by a contradiction.

Step 1: Assuming that a condition contrary to Condition (26) holds, there exists κ such that for all $k \geq \kappa$,

$$c^k < \frac{q^* - \tilde{L}(\lambda^k, x^k)}{\left\| \tilde{g}(x^k) \right\|^2}. \tag{32}$$

Under the surrogate optimality condition (12) and the condition (32), surrogate subgradient directions form acute angles with directions toward λ^*, multipliers move closer to λ^*, the lower bound property (16) of the surrogate dual is preserved, and the following inequality holds [15,17]:

$$0 < q^* - \tilde{L}(\lambda^k, x^k) \leq (\lambda^* - \lambda^k)^T \tilde{g}(x^k) \leq \left\| \lambda^* - \lambda^k \right\| \left\| \tilde{g}(x^k) \right\|. \tag{33}$$

From (32) and (33) it follows that

$$c^k < \frac{\left\| \lambda^* - \lambda^k \right\| \left\| \tilde{g}(x^k) \right\|}{\left\| \tilde{g}(x^k) \right\|^2} = \frac{\left\| \lambda^* - \lambda^k \right\|}{\left\| \tilde{g}(x^k) \right\|}. \tag{34}$$

Therefore, for all $k \geq \kappa$,

$$c^k \left\| \tilde{g}(x^k) \right\| < \left\| \lambda^* - \lambda^k \right\|. \tag{35}$$

In general, stepsizes satisfying (20) and (35) may not lead to convergence, since stepsizes c^k may decrease faster than distances between λ^* and λ^k, and multipliers may not reach λ^*.

Step 2: It will be proved that the condition (21b) ensures that stepsizes c^k decrease slower than distances between λ^* and λ^k, and that the inequality (35) is violated as a result.

Consider the inequality (35) at an iteration $\kappa + m$ $(m > 0)$

$$c^{\kappa+m} \left\| \tilde{g}(x^{\kappa+m}) \right\| < \left\| \lambda^* - \lambda^{\kappa+m} \right\|. \tag{36}$$

Since the inequality (32) holds by assumption, multipliers move closer to λ^*, and there exists $0 < \beta^{\kappa+m-1} < 1$ such that

$$\left\| \lambda^* - \lambda^{\kappa+m} \right\| = \beta^{\kappa+m-1} \left\| \lambda^* - \lambda^{\kappa+m-1} \right\|. \tag{37}$$

The value of $\beta^k (k \geq \kappa)$ is the rate with which multipliers approach λ^*. When $\beta^k \sim 1$,[5] the contradiction will be established by showing that the left-hand side of (35) decreases slower than the right-hand side for sufficiently large values of $\alpha_k (< 1)$ as k

[5] When $\beta^k << 1$, the right-hand side of (35) decreases faster than the left-hand side as k increases. This leads to the contradiction, and the theorem is proved.

increases. With the stepsizing formula (20) and with the equality (37), the inequality (36) becomes

$$\alpha_{\kappa+m-1} c^{\kappa+m-1} \left\| \tilde{g}(x^{\kappa+m-1}) \right\| < \beta^{\kappa+m-1} \left\| \lambda^* - \lambda^{\kappa+m-1} \right\|. \tag{38}$$

Following the same logic, the inequality (38) can be inductively represented in the following way:

$$c^{\kappa} \left\| \tilde{g}(x^{\kappa}) \right\| < \frac{\prod_{i=\kappa}^{\kappa+m-1} \beta_i}{\prod_{i=\kappa}^{\kappa+m-1} \alpha_i} \left\| \lambda^* - \lambda^{\kappa} \right\|, \ m > 0. \tag{39}$$

To arrive at the contradiction, given that the left-hand of (39) is positive, the right-hand side of (39) will be proved to be arbitrarily small under (21b) as m increases.

From (14)–(15), (37), and the non-expansive property of projections, it follows that

$$\beta^k := \frac{\left\| \lambda^* - \lambda^{k+1} \right\|}{\left\| \lambda^* - \lambda^k \right\|} \leq \frac{\left\| \lambda^* - \lambda^k - c^k \tilde{g}(x^k) \right\|}{\left\| \lambda^* - \lambda^k \right\|}. \tag{40}$$

The right-hand side of (40) can be expanded in Taylor series around $c^k \to 0$, while keeping first two terms of the expansion by using the following relation:

$$\frac{\partial}{\partial c} \|h(c)\|_2 = \frac{h(c)^T \frac{\partial}{\partial c} h(c)}{\|h(c)\|_2}, \tag{41}$$

where h is a vector-valued function of c. Therefore,

$$\beta^k \leq 1 - \frac{(\lambda^* - \lambda^k)^T \ \tilde{g}(x^k) \ c^k}{\left\| \lambda^* - \lambda^k \right\|^2} + O((c^k)^2). \tag{42}$$

Consider the following ratio:

$$\frac{1-\alpha_k}{1-\beta^k} \leq \frac{1-\alpha_k}{c^k} \left(\frac{(\lambda^* - \lambda^k)^T \ \tilde{g}(x^k)}{\left\| \lambda^* - \lambda^k \right\|^2} - O(c^k) \right)^{-1}. \tag{43}$$

The second term of the product in the right-hand side of (43) is bounded. Indeed, from the relation (33) it follows that

$$\varepsilon < \frac{(\lambda^* - \lambda^k)^T \ \tilde{g}(x^k)}{\left\| \lambda^* - \lambda^k \right\|^2} \leq \frac{\left\| \tilde{g}(x^k) \right\|}{\left\| \lambda^* - \lambda^k \right\|}, \tag{44}$$

for any small $\varepsilon > 0$.

For sufficiently small c^k, we can assume that $-\varepsilon/2 < O(c^k) < \varepsilon/2$, therefore,

$$\frac{\varepsilon}{2} < \frac{(\lambda^* - \lambda^k)^T \, \tilde{g}(x^k)}{\left\| \lambda^* - \lambda^k \right\|^2} - O(c^k). \tag{45}$$

Assuming $\left\| \lambda^k - \lambda^* \right\| > \varepsilon > 0$, and $\left\| g(x) \right\| < M < \infty$, from (44) and (45) it follows that

$$0 < \frac{\varepsilon}{2} < \frac{(\lambda^* - \lambda^k)^T \, \tilde{g}(x^k)}{\left\| \lambda^* - \lambda^k \right\|^2} - O(c^k) < \frac{\left\| \tilde{g}(x^k) \right\|}{\left\| \lambda^* - \lambda^k \right\|} + \frac{\varepsilon}{2} < \frac{M}{\varepsilon} + \frac{\varepsilon}{2} < \infty. \tag{46}$$

Therefore, the reciprocal value in (43) is also bounded. On the other hand, if $\left\| \lambda^k - \lambda^* \right\| < \varepsilon$ for any small $\varepsilon > 0$, then $\lambda^k \to \lambda^*$, and the convergence is proved.

When the asymptotical condition (21b) holds, the right-hand side of (43) converges to zero as $k \to \infty$. Therefore, the left-hand side of (43) converges to zero

$$\frac{1 - \alpha_k}{1 - \beta^k} \to 0. \tag{47}$$

To arrive at the contradiction, we need to show that, while the left-hand side of (39) is constant for a given κ, the right-hand side can be made arbitrarily small as $m \to \infty$, provided (47) holds. That is, it remains to be proved that, for any predetermined and arbitrarily small value $\varepsilon > 0$, there exists an iteration m_ε satisfying

$$\frac{\prod_{i=\kappa}^{\kappa+m_\varepsilon-1} \beta^i}{\prod_{i=\kappa}^{\kappa+m_\varepsilon-1} \alpha_i} < \varepsilon. \tag{48}$$

Based on (47), α_k approaches 1 faster than the entire expression in (47) approaches zero. Therefore, there exists an iteration N such that for any $n > N$ there exist a positive constant $\delta_n > 0$, and for an arbitrarily small positive $\varepsilon_1 > 0$ the following conditions hold:

$$\frac{1 - \alpha_n}{1 - \beta^n} < \varepsilon_1 \tag{49}$$

and

$$1 - \alpha_n = \varepsilon_1^{1+\delta_n}. \tag{50}$$

From the inequality (49), it follows that

$$\frac{\beta^n}{\alpha_n} < \frac{1}{\alpha_n} \left(1 + \frac{\alpha_n}{\varepsilon_1} - \frac{1}{\varepsilon_1} \right) = \frac{1}{\varepsilon_1} + \frac{1}{\alpha_n} \left(1 - \frac{1}{\varepsilon_1} \right). \tag{51}$$

Based on (50), the inequality (51) becomes

$$\frac{\beta^n}{\alpha_n} < \frac{1}{\varepsilon_1} + \frac{1}{1-\varepsilon_1^{1+\delta_n}}\left(\frac{\varepsilon_1 - 1}{\varepsilon_1}\right) = \frac{1}{\varepsilon_1}\left(1 + \frac{\varepsilon_1 - 1}{1-\varepsilon_1^{1+\delta_n}}\right)$$
$$= \frac{1}{\varepsilon_1}\left(\frac{-\varepsilon_1^{1+\delta_n} + \varepsilon_1}{1-\varepsilon_1^{1+\delta_n}}\right) = \left(\frac{1-\varepsilon_1^{\delta_n}}{1-\varepsilon_1^{1+\delta_n}}\right). \tag{52}$$

Given that $\varepsilon_1 > 0$ is arbitrarily small, (52) becomes

$$\frac{\beta^n}{\alpha_n} < \left(\frac{1-\varepsilon_1^{\delta_n}}{1-\varepsilon_1^{1+\delta_n}}\right) \sim 1-\varepsilon_1^{\delta_n} < 1-\varepsilon_1^{1+\delta_n} = \alpha_n. \tag{53}$$

Therefore, given (21a),

$$\prod_{i=n}^{k}\frac{\beta^i}{\alpha_i} < \prod_{i=n}^{k}\alpha_i \to 0. \tag{54}$$

Thus, the inequality (48) is established for an arbitrary small value $\varepsilon > 0$ and iteration m_ε. Therefore, (39) becomes

$$c^\kappa \left\|\tilde{g}(x^\kappa)\right\| < \varepsilon \left\|\lambda^* - \lambda^\kappa\right\|. \tag{55}$$

Since $\varepsilon > 0$ is arbitrarily small, the inequality (55) does not hold for a fixed iteration κ. Therefore, the inequality (34) does not hold for $k = \kappa + m_\varepsilon$. This contradicts the assumption, and convergence to λ^* is proved. □

Based on the convergence results proved in Theorem 2.1, the following Corollary discusses the convergence of the interleaved method [14] developed for separable problems. In the method, Lagrange multipliers are updated after each subproblem is solved.

Corollary 2.4 *The interleaved method converges with the novel stepsizing formula (20) provided the conditions (21a) and (21b) hold.*

Proof The interleaved method is defined for separable problems. For such problems, after constraints are relaxed, the Lagrangian function can be represented in an additive form $L = L_1 + \cdots + L_{N_s}$, and the relaxed problem can be separated into N_s subproblems. To prove this Corollary, it is sufficient to show that the surrogate optimality condition (12) holds after one subproblem is solved. Indeed, after a subproblem i is solved to optimality, and x_i^k is obtained, then by definition of an optimum

$$L_i(\lambda^k, x_i^k) \le L_i(\lambda^k, \xi), \ \ \forall \xi. \tag{56}$$

Since the inequality (56) holds for all feasible ξ, it also holds for x^{k-1}

$$L_i(\lambda^k, x_i^k) \le L_i(\lambda^k, x_i^{k-1}), \tag{57}$$

Since subproblems, other than i, are not optimized, the following equality holds

$$L_{-i}(\lambda^k, x_{-i}^k) = L_{-i}(\lambda^k, x_{-i}^{k-1}), \tag{58}$$

where $x_{-i}^{k-1} = x_j^{k-1}$, $j = 1, \ldots, N_s$, $j \neq i$. Given that the problem is separable, and the Lagrangian is additive,

$$L(\lambda^k, x^k) \leq L(\lambda^k, x^{k-1}), \tag{59}$$

where $x^k = (x_1^{k-1}, \ldots, x_i^k, \ldots x_{N_s}^{k-1})$.

If the inequality (59) is strict, then the Corollary is proved. If the inequality (59) holds as an equality, then the method proceeds by optimizing the next subproblem until the inequality (59) holds as a strict inequality. If, nevertheless, after solving all the subproblems, the surrogate optimality condition is satisfied as an equality, this means that a surrogate dual value equals to a dual value, and a surrogate subgradient direction equals to a subgradient direction. This can happen if λ^{k-1} and λ^k belong to the same facet of the dual function, and subgradient directions at iterations k and k-1 are equal. At this point, the rest of the proof is identical to the results proved in Theorem 2.1 and Proposition 2.2. □

2.3 Convergence Rate of the Surrogate Lagrangian Relaxation Method

Following the general framework of standard subgradient methods [8], it is proved in Proposition 2.5 that when λ^k are not too close to λ^*, convergence rate is linear assuming that stepsizes c^k are sufficiently small.

Proposition 2.5 *Under the Assumption 2.1 [8], the new method converges with a linear rate for sufficiently small stepsizes c^k, assuming there exists a scalar $\mu > 0$ that satisfies*

$$q^* - \tilde{L}(\lambda^k, x^k) \geq \mu \left\| \lambda^* - \lambda^k \right\|^2, \quad k = 0, 1, \ldots, \tag{60}$$

and stepsizes c^k satisfy

$$0 < c^k < \frac{1}{2\mu}, \quad k = 0, 1, \ldots, \tag{61}$$

and

$$\frac{c^k \left\| \tilde{g}(x^k) \right\|^2}{\mu} < \left\| \lambda^* - \lambda^k \right\|^2, \quad k = 0, 1, \ldots. \tag{62}$$

Proof From the inequalities (33) and (60), it follows that

$$\left\| \lambda^* - \lambda^{k+1} \right\|^2 \leq \left\| \lambda^* - \lambda^k \right\|^2 (1 - 2c^k \mu) + (c^k)^2 \left\| \tilde{g}(x^k) \right\|^2. \tag{63}$$

Dividing both sides of (63) by $\left\|\lambda^* - \lambda^k\right\|^2$ yields

$$(\beta^k)^2 \equiv \frac{\left\|\lambda^* - \lambda^{k+1}\right\|^2}{\left\|\lambda^* - \lambda^k\right\|^2} \le (1 - 2c^k\mu) + \frac{(c^k)^2\left\|\tilde{g}(x^k)\right\|^2}{\left\|\lambda^* - \lambda^k\right\|^2}. \tag{64}$$

Intuitively, given that stepsizes are sufficiently small, and multipliers are sufficiently far from λ^*, the last term is negligibly small, and the convergence rate is linear with $\beta^k \approx \sqrt{(1 - 2c^k\mu)} < 1$. To determine the neighborhood of λ^*, to which the linear convergence can be guaranteed, the right-hand side of (64) should be less than 1, that is

$$(1 - 2c^k\mu) + \frac{(c^k)^2\left\|\tilde{g}(x^k)\right\|^2}{\left\|\lambda^* - \lambda^k\right\|^2} < 1. \tag{65}$$

Under the condition (62), the inequality (65) holds. □

Remark 2.6 Assumptions (60) and (62) imply the following assumption on stepsizes

$$c^k < \frac{q^* - \tilde{L}(\lambda^k, x^k)}{\left\|\tilde{g}(x^k)\right\|^2}. \tag{66}$$

The assumption (66) is the condition on the stepsizes (13) used in the convergence proof of the surrogate subgradient method [15]. As stated earlier, under the condition (66), the lower bound of the surrogate dual is preserved per (16), thereby implying that the left-hand side of (60) is positive, and $\mu > 0$ that satisfies (60) exists. In other words, under (62), assumptions (60) and (66) are equivalent. □

2.4 Practical Implementation of the Surrogate Lagrangian Relaxation Method

This subsection discusses practical implementation aspects of our method. A constructive rule for setting parameters α_k is developed in Proposition 2.7 and proved to satisfy conditions (21a) and (21b) required for convergence to λ^* without requiring q^*. Lastly, an algorithm of the method is presented.

Proposition 2.7 *The stepsize-setting parameters α_k can be updated as follows to ensure that the multipliers converge to λ^*:*

$$\alpha_k = 1 - \frac{1}{Mk^p},\ \ p = 1 - \frac{1}{k^r},\ \ M \ge 1,\ \ 0 < r < 1,\ \ k = 2, 3, \ldots. \tag{67}$$

Proof Step 1: To show that stepsizes (22) converge to zero, it is sufficient to show that the following product converges to zero

$$\prod_{i=1}^{k} \alpha_i = \prod_{i=1}^{k}\left(1 - \frac{1}{Mi^p}\right). \tag{68}$$

For the ease of the proof, convergence of (68) to zero will be established by proving that the natural logarithm of (68) converges to $-\infty$. After taking the logarithm of the product (68), it becomes the sum of the logarithms

$$\log\left(\prod_{i=1}^{k}\alpha_i\right)=\sum_{i=1}^{k}\log(\alpha_i)=\sum_{i=1}^{k}\log\left(1-\frac{1}{Mi^p}\right). \tag{69}$$

Indeed, as $i\to\infty$, the first term of the Taylor series expansion of $\log(1-\frac{1}{Mi^p})$ is $-\frac{1}{Mi}$. Therefore, the sum (69) converges to $-\infty$, and stepsizes (22) converge to zero.

Step 2: To show that condition (21b) of Theorem 2.1 holds, given (67), condition (21b) can be rewritten as

$$\frac{1-\alpha_k}{c^k}\sim\frac{\frac{1}{Mk^p}}{\prod\limits_{i=1}^{k}\left(1-\frac{1}{Mi^p}\right)}. \tag{70}$$

As before, it will be shown that that the logarithm of (70) approaches $-\infty$. Consider the logarithm of the right-hand side of (70)

$$\log\left(\frac{1}{Mk^p}\right)-\sum_{i=1}^{k}\log\left(1-\frac{1}{Mi^p}\right). \tag{71}$$

To prove the asymptotical condition (21b), it is sufficient to demonstrate that

$$\log\left(1-\frac{1}{Mk^p}\right)=o\left(\frac{1}{k}\log\left(\frac{1}{Mk^p}\right)\right). \tag{72}$$

Given that $p\to 1$ as $k\to\infty$, the following relation holds:

$$\lim_{k\to\infty}\frac{\log(1-\frac{1}{Mk^p})}{\frac{1}{k}\log(\frac{1}{Mk^p})}=\lim_{k\to\infty}\frac{\log(1-\frac{1}{Mk})}{\frac{1}{k}\log(\frac{1}{Mk})}. \tag{73}$$

Using the L'Hopital's rule leads to

$$\lim_{k\to\infty}\frac{\log\left(1-\frac{1}{Mk}\right)}{\frac{1}{k}\log\left(\frac{1}{Mk}\right)}=\lim_{k\to\infty}\frac{\frac{\mathrm{d}}{\mathrm{d}k}\log\left(1-\frac{1}{Mk}\right)}{\frac{\mathrm{d}}{\mathrm{d}k}\left(\frac{1}{k}\log\left(\frac{1}{Mk}\right)\right)}=\lim_{k\to\infty}\frac{k}{(1-Mk)(1-\log(Mk))}. \tag{74}$$

Applying L'Hopital's rule one more time yields

$$\begin{aligned}\lim_{k\to\infty}\frac{k}{(1-Mk)(1-\log(Mk))}&=\lim_{k\to\infty}\frac{1}{k^{-1}-M\log(Mk)}\\&=-\lim_{k\to\infty}\frac{1}{M\log(Mk)}=0.\end{aligned} \tag{75}$$

As proved in the steps above, $c^k \to 0$, and the condition (21b) holds. Therefore, convergence of multipliers to λ^* is proved. □

The entire algorithm can be summarized in the following steps:

Step 0: Initialize multipliers λ^0, obtain x^0 by optimizing the relaxed problem, estimate $\hat{q}$ of q^* by using best heuristics available for a particular problem, initialize c^0 according to

$$c^0 = \frac{\hat{q} - \tilde{L}(\lambda^0, x^0)}{\left\| \tilde{g}(x^0) \right\|^2}. \tag{76}$$

Step 1: Update α_k, for example, by using (67). For given values (α_k, x^k), update stepsizes c^k according to (20). For given values (x^k, λ^k, c^k), update multipliers according to (14)-(15) to obtain λ^{k+1}.

Step 2: For the given λ^{k+1}, minimize the Lagrangian function until the surrogate optimality condition (12) is satisfied. As a special case, for separable problems, it is sufficient to optimize just one subproblem (Corollary 2.4).

Step 3: Check stopping criteria: CPU time, number of iterations, surrogate subgradient norm, distance between multipliers, etc. If stopping criteria are satisfied, then go to Step 4. Otherwise, go to Step 1.

Step 4: Obtain feasible solutions. Problem-specific heuristics may be used to obtain feasible costs while a dual value provides a lower bound on the optimal cost. A duality gap can then be calculated by using the best available feasible cost and the largest available dual value.

As proved before, at convergence of multipliers, a surrogate dual value converges to a dual value. If the algorithm is terminated before convergence, a dual value can be obtained by fully optimizing the relaxed problem. In Sect. 3, it will be demonstrated that owing to reduced computational requirements, the new method can obtain a better dual value, a better feasible cost, and a lower duality gap as compared to other methods.

3 Numerical Testing

The purpose of this section is to compare the surrogate Lagrangian relaxation method with other methods that are used for optimizing non-smooth dual function such as the subgradient-level method and the incremental subgradient method. In Example 1, a small nonlinear (quadratic) integer problem is considered to demonstrate that, surrogate subgradient directions frequently form small acute angles with directions toward the optimal multipliers, thereby alleviating the zigzagging issues that often accompany the subgradient method. In Example 2, linear integer generalized assignment problems are considered to demonstrate that the new method is capable of handling large separable optimization problems. It is then demonstrated that when simple heuristics are used to adjust relaxed problem solutions to obtain feasible costs, the method is capable of reducing the duality gap as compared to other methods such as the incremental subgradient method. In Example 3, nonlinear integer quadratic assignment problem is considered to demonstrate the quality of the method for optimizing non-separable non-smooth dual problems, and the method is compared with the subgradient-level method. The new method is implemented using CPLEX 12.2 on Intel® Xeon® CPU E5620 (12M Cache, 5.86 GT/s Intel® QPI) @ 2.40 GHz (2 processors) and 36.00 GB of RAM.

Example 3.1 **A Nonlinear Integer Problem** Consider the following nonlinear integer optimization problem

$$\min_{\{x_1,x_2,x_3,x_4,x_5,x_6\}\in\mathbb{Z}_+\cup\{0\}} \left\{0.5x_1^2+0.1x_2^2+0.5x_3^2+0.1x_4^2+0.5x_5^2+0.1x_6^2\right\} \quad (77)$$

$$\begin{aligned} s.t.\quad & 48-x_1+0.2x_2-x_3+0.2x_4-x_5+0.2x_6\le 0,\\ & 250-5x_1+x_2-5x_3+x_4-5x_5+x_6\le 0. \end{aligned} \quad (78)$$

After constraints (78) are relaxed by the multipliers λ_1 and λ_2, respectively, the Lagrangian function becomes

$$\begin{aligned} L(x_1,x_2,x_3,x_4,x_5,x_6,\lambda_1,\lambda_2) = {} & 0.5x_1^2+0.1x_2^2+0.5x_3^2\\ & +0.1x_4^2+0.5x_5^2+0.1x_6^2+\\ & \lambda_1(48-x_1+0.2x_2-x_3+0.2x_4-x_5+0.2x_6)\\ & +\lambda_2(250-5x_1+x_2-5x_3+x_4-5x_5+x_6). \end{aligned} \quad (79)$$

Given that the objective function and coupling constraints in (77)–(79) are of an additive form, the relaxed problem can be separated into six individual subproblems:

$$\begin{aligned} & \min_{\{x_i\}\in\mathbb{Z}_+\cup\{0\},\ i=1,\ldots,6} L(x_1,x_2,x_3,x_4,x_5,x_6,\lambda_1,\lambda_2)\\ & = \min_{\{x_i\}\in\mathbb{Z}_+\cup\{0\},\ i=1,\ldots,6} \{(0.5x_1^2-\lambda_1x_1-5\lambda_2x_1)+(0.1x_2^2+0.2\lambda_1x_2+\lambda_2x_2)\\ & +(0.5x_3^2-\lambda_1x_3-5\lambda_2x_3)+(0.1x_4^2+0.2\lambda_1x_4+\lambda_2x_4)\\ & +(0.5x_5^2-\lambda_1x_5-5\lambda_2x_5)+(0.1x_6^2+0.2\lambda_1x_6+\lambda_2x_6)+48\lambda_1+250\lambda_2\}. \end{aligned} \quad (80)$$

To compare subgradient and surrogate Lagrangian relaxation methods, the stepsizing formula (20) is used to update the multipliers within both frameworks. The stepsize is initialized according to (76) by using an optimal value of the LP relaxation of (77)–(78), as an estimate of q^*. In the subgradient method, the relaxed problem (80) is optimized with respect to all $\{x_i\}$, $i = 1, \ldots, 6$. Since the relaxed problem (80) is separable, individual subproblems can be solved individually. In this example, three out of six subproblems are solved per iteration to obtain surrogate subgradient directions. The multipliers λ_1 and λ_2 are updated 18 iterations within the subgradient framework and 36 iterations within the surrogate Lagrangian relaxation framework. In both frameworks, each subproblem is solved 18 times. The trajectories of the multipliers are shown in Figs. 1 and 2.

Figure 1 demonstrates that the subgradient directions $g(x^k)$ are frequently almost perpendicular to the directions $\lambda^* - \lambda^k$ toward λ^* (respective directions are shown in

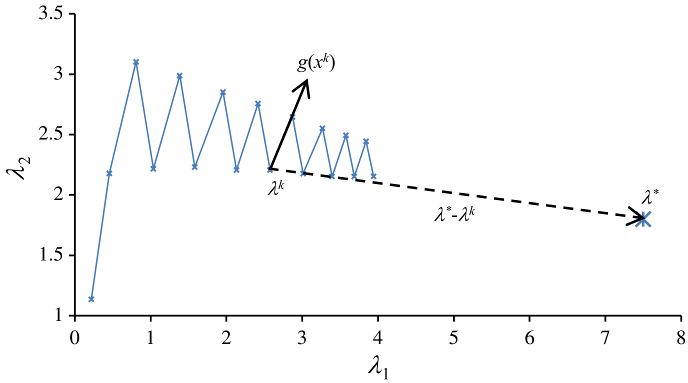


Fig. 1 Trajectories of the multipliers using the subgradient method

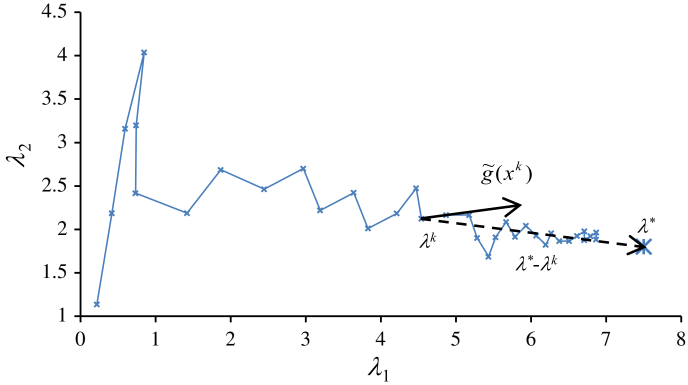


Fig. 2 Trajectories of the multipliers using the surrogate subgradient method

Table 1 Comparison of the subgradient method and the surrogate Lagrangian relaxation method

Method	Number of iterations	CPU time (s)	Distance to the optimum
Subgradient method	18	2.75	3.574777
Surrogate Lagrangian relaxation method	36	1.62	0.798711

Fig. 1 by solid and dashed arrows), and the multipliers zigzag causing slow convergence.

In contrast, the surrogate directions $\tilde{g}(x^k)$ (shown by a solid arrow in Fig. 2) are smoother and frequently form smaller angles with the directions $\lambda^* - \lambda^k$ toward λ^* (shown by a dashed arrow in Fig. 2), thereby alleviating zigzagging and leading to faster convergence.

Table 1 demonstrates that within the surrogate Lagrangian relaxation framework, the multipliers move closer to λ^* as compared to the multipliers updated by using subgradient directions, thereby reducing the number of iterations required for convergence. In addition, since the relaxed problem is not fully optimized in the new method, the surrogate subgradient directions are easier to obtain. This also leads to faster convergence in terms of the computation time.

Example 3.2 **Generalized Assignment Problems** In generalized assignment problems, the total cost for assigning a given set of jobs to available machines is minimized. Each job is assigned to one machine, and the total processing time for all jobs assigned to a machine should not exceed the machine's time available. Mathematically, the generalized assignment problem is formulated in the following way:

$$\min_{x_{i,j}} \sum_{i=1}^{I} \sum_{j=1}^{J} g_{i,j} x_{i,j},\, x_{i,j} \in \{0,\ 1\},\ \ g_{i,j} \geq 0,\ \ a_{i,j} \geq 0,\ \ b_j \geq 0, \tag{81}$$

$$s.t. \quad \sum_{i=1}^{I} a_{i,j} x_{i,j} \leq b_j,\, j = 1,\ \ldots,\ J, \tag{82}$$

$$\sum_{j=1}^{J} x_{i,j} = 1,\ i = 1,\ \ldots,\ I, \tag{83}$$

where I is the number of jobs, and J is the number of machines, $a_{i,j}$ is time required by machine j to perform job i, and $g_{i,j}$ is cost for assigning job i to machine j. Capacity constraints (82) ensure that the total amount of time, required by the jobs to be performed on a given machine, does not exceed the machine j's time available b_j. Constraints (83) ensure that each job is to be performed on one and one machine only. For more details, refer to [24–31].

Since the objective function of (81) and constraints (82)–(83) are of an additive form, after relaxing constraints (83) by introducing the Lagrange multipliers, the problem is formulated in a separable form

$$\begin{aligned} q(\lambda) &= \min_{x_{i,j}} \sum_{j=1}^{J} \sum_{i=1}^{I} \left(g_{i,j} + \lambda_i\right) x_{i,j} - \sum_{i=1}^{J} \lambda_i, \\ s.t. \quad & \sum_{i=1}^{I} a_{i,j} x_{i,j} \leq b_j,\, j = 1,\ \ \ldots,\ \ J,\, x_{i,j} \in \{0,\ 1\},\ \ g_{i,j} \geq 0,\ \ a_{i,j} \geq 0, \\ & b_j \geq 0. \end{aligned} \tag{84}$$

As proved in Corollary 2.4, optimization with respect to only one subproblem is sufficient to satisfy the surrogate optimality condition

$$\begin{aligned} &\sum_{i=1}^{I} \left(\sum_{j=1}^{J} g_{i,j} x_{i,j}^{k+1} + \lambda_i^{k+1} \left(\sum_{j=1}^{J} x_{i,j}^{k+1} - 1 \right) \right) \\ &< \sum_{i=1}^{I} \left(\sum_{j=1}^{J} g_{i,j} x_{i,j}^{k} + \lambda_i^{k+1} \left(\sum_{j=1}^{J} x_{i,j}^{k} - 1 \right) \right). \end{aligned} \tag{85}$$

As discussed earlier, the accompanying computational effort is approximately $1/J$ per subproblem compared to the effort required to fully optimize the relaxed problem and obtain subgradient directions.

3.1 Comparison to Standard Methods for Non-Smooth Optimization

To demonstrate the quality of the surrogate Lagrangian relaxation method, it is compared to existing methods available for optimizing non-smooth dual functions, such as the simple subgradient method, the simple subgradient-level method, and the incremental subgradient method. The comparison to the last two methods is especially important, since they do not require q^* for convergence to λ^*.

3.1.1 The Simple Subgradient Method

In the method [22], the relaxed problem (84) is fully optimized, and stepsizes are updated according to the following relation

$$0 < c^k < \alpha \frac{UB - q(\lambda^k)}{\left\| g(x^k) \right\|^2}, \ 0 < \alpha < 2, \tag{86}$$

where UB is the best feasible cost available at iteration k.

3.1.2 The Simple Subgradient-Level Method

In the method [7], the relaxed problem (84) is fully optimized, and stepsizes are updated according to the following relation

$$0 < c^k < \alpha \frac{q^{lev} + \delta_k - q(\lambda^k)}{\left\| g(x^k) \right\|^2}, \ 0 < \alpha < 2. \tag{87}$$

3.1.3 The Incremental Subgradient Method

In the method [8], each subproblem is solved to optimality. After each problem is optimized, multipliers are updated, and stepsizes are updated, similarly to the subgradient-level method, according to

$$0 < c^k < \alpha \frac{q^{lev} + \delta_k - q(\lambda^k)}{n \left\| g(x^k) \right\|^2}, \ 0 < \alpha < 2, \tag{88}$$

where n is the number of subproblems, q^{lev} is the best dual value obtained up until iteration k, and δ_k is a parameter that decreases by a factor of 2 every time a significant oscillation of multipliers is detected, that is when multipliers "travel" a distance

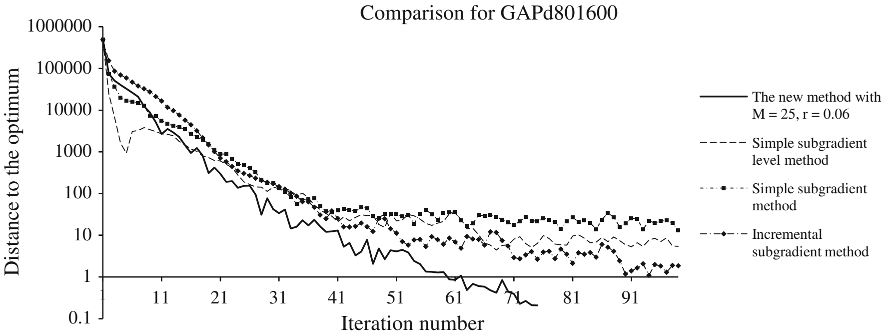


Fig. 3 Comparison of the surrogate Lagrangian relaxation method with parameters $M = 25$ and $r = 0.06$ against: *1* the simple subgradient method with $\alpha = 1$; *2* the subgradient-level method with parameters $\delta_0 = 100000$ and $B = 1000$; *3* the incremental subgradient method with parameters $\delta_0 = 100000$ and $B = 1000$

exceeding a predetermined value B

$$\sigma_k > B, \tag{89}$$

where

$$\sigma_k = \sigma_{k-1} + \left\| \lambda^k - \lambda^{k-1} \right\|. \tag{90}$$

Once significant oscillations are detected, and condition (89) is satisfied, σ_k is reset to 0. For more information, refer to [7,8].

For a fair comparison of the methods, each subproblem is solved exactly once per iteration. For example, within the subgradient method, minimization of the relaxed problem counts as one iteration. In the incremental subgradient method, one iteration is complete once each subproblem is solved exactly once. In the new method, 10, 2 and 1 subproblems were chosen to be solved for instances GAPd801600, GAPd201600, and GAPd15900,[6] respectively. Therefore, for these instances the number of sub-iterations is 8, 10, and 15, respectively.

Figures 3, 4, 5 demonstrate performance of the surrogate Lagrangian relaxation method, as compared to subgradient methods.[7] Numerical results indicate that within the incremental subgradient framework, multipliers approach λ^* slowly, since stepsizes decrease to zero slowly. This happens because as stepsizes decrease, it takes more iterations for multipliers to "travel" distance B. This leads to slow convergence when multipliers move closer to λ^*. For a similar reason, convergence of the subgradient-level method can be slower as compared to the surrogate Lagrangian relaxation method.

6 For the GAP15900 instance, the implementation of the new method may resemble that of the interleaved method [14] since only one subproblem is optimized at a time. The important difference between the new method and the interleaved method is the stepsizing formula.

7 Performance of all methods in Figs. 3, 4, 5 is tested by comparing distances to multipliers obtained by a subgradient method with non-summable stepsizes [22] after sufficiently many iterations (>20000).

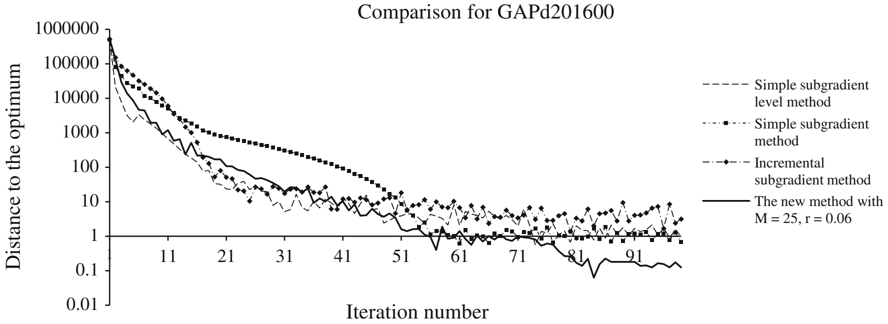


Fig. 4 Comparison of the surrogate Lagrangian relaxation method with parameters $M = 25$ and $r = 0.06$ against: *1* the simple subgradient method with $\alpha = 1$; *2* the subgradient-level method with $\delta_0 = 100000$ and $B = 500$; *3* the incremental subgradient method with parameters $\delta_0 = 100000$ and $B = 1000$

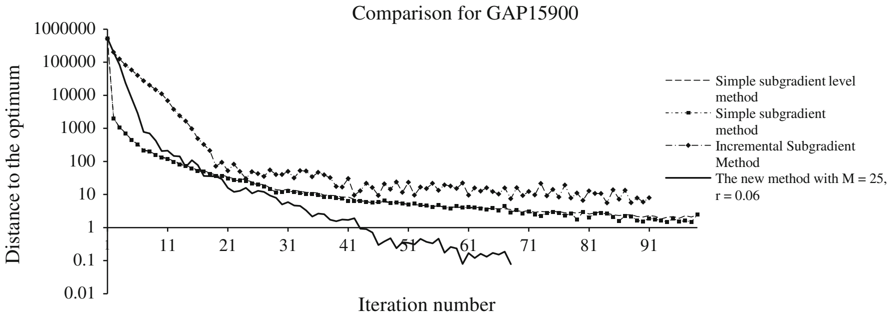


Fig. 5 Comparison of the surrogate Lagrangian relaxation method with parameters $M = 25$ and $r = 0.06$ against: *1* the simple subgradient method with $\alpha = 1$; *2* the subgradient-level method with parameters $\delta_0 = 50000$ and $B = 750$; *3* the incremental subgradient method with parameters $\delta_0 = 50000$ and $B = 750$

Since duality gaps of generalized assignment problems are typically small, a feasible cost can provide a reasonably good approximation of q^* within the simple subgradient method. However, convergence to λ^* does not occur.

The following figure demonstrates a comparison of duality gaps obtained by the new method and the incremental subgradient method for the GAP d201600 instance.

As demonstrated in Fig. 6, owing to the reduced computational effort, in the new method the dual value increases faster, and with the help of heuristics, feasible costs obtained are better, as compared to the incremental subgradient method. As a result, the duality gap obtained by using the new method is smaller than the gap obtained by using the incremental subgradient method.

Example 3.3 **Quadratic Assignment Problems** The objective of the Quadratic Assignment Problem (QAP) of order n is to find the best allocation of n facilities to n locations. Formulated in 1957 by [32], the problem has been applied to the planning of buildings in university campuses [33], arrangement of departments in hospitals [34], scheduling parallel production lines [35], and ranking of archeological data [36]. It has also been shown that QAPs can be applied to the field ergonomics to solve the typewriter keyboard design problem [37].

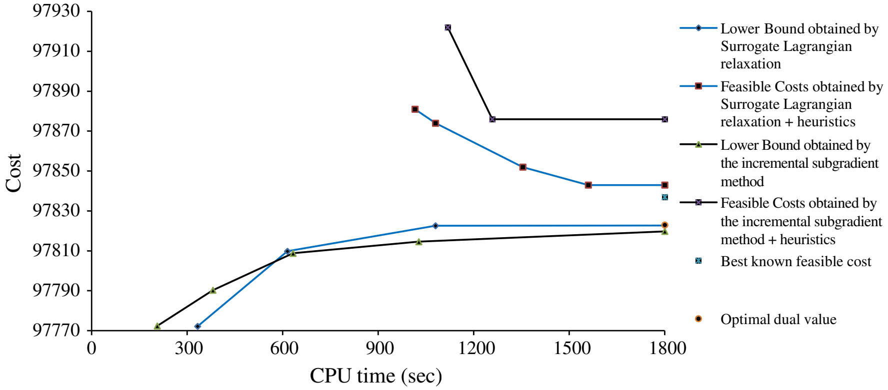


Fig. 6 Comparison of the surrogate Lagrangian relaxation method with parameters $M = 20$ and $r = 0.1$ against the incremental subgradient method with parameters $\delta_0 = 500$ and $B = 15$ for solving the GAPd201600 instance

Mathematically, the quadratic assignment problem can be formulated as an integer programming problem:

$$\min_{x_{i,j},x_{h,l}} \sum_{i,j=1}^{n} \sum_{h,l=1}^{n} d_{i,h} f_{j,l} x_{i,j} x_{h,l}, \, x_{i,j} \in \{0,\ 1\},\ d_{i,h} \geq 0,\ f_{j,l} \geq 0, \tag{91}$$

$$s.t. \quad \sum_{i=1}^{n} x_{i,j} = 1,\ j = 1,\ \ldots,\ n, \tag{92}$$

$$\sum_{j=1}^{n} x_{i,j} = 1,\ i = 1,\ \ldots,\ n, \tag{93}$$

where n is the number of facilities and locations, $d_{i,h}$ is the distance between location i and location h, $f_{j,l}$ is the weight/flow between facility j and facility l (the net transfer of goods/supplies from facility j to l). Intuitively, two facilities with high flow should be built close to each other. Binary decision variables $x_{i,j}$ correspond to facility i being placed in location j iff $x_{i,j} = 1$. Assignment constraints (92) and (93) ensure that one and one facility only can be assigned to a specific location.

The problem formulation (91)–(93) is non-separable because of the cross-product of decision variables in the objective function of (91). For a fair comparison of the methods, after the problem is linearized, branch-and-cut will be used to obtain approximate solutions of the relaxed problem for the surrogate Lagrangian relaxation method and exact solutions of the relaxed problem for the subgradient-level method.

After relaxing constraints (93) by introducing Lagrange multipliers, the relaxed problem becomes:

$$\min_{x_{i,j},x_{h,l}} \left\{ \sum_{i,j=1}^{n} \sum_{h,l=1}^{n} d_{i,h} f_{j,l} x_{i,j} x_{h,l} + \sum_{i=1}^{n} \lambda_i \left(\sum_{j=1}^{n} x_{i,j} - 1 \right) \right\},$$
$$s.t.\, x_{i,j} \in \{0,\ 1\} \text{ and } (92). \tag{94}$$

Since decision variables $x_{i,j}$ and $x_{h,l}$ are binary, feasible region for the product $x_{i,j} \cdot x_{h,l}$ consists of the four points: (0,0), (0,1), (1,0), and (1,1). Moreover, the product takes on the value of 1 if and only if both decision variables equal to 1. Based on this observation, the relaxed problem can be equivalently rewritten in a linear form as:

$$\min_{x_{i,j},x_{h,l},F_{i,j,h,l}} \left\{ \sum_{i,j=1}^{n} \sum_{h,l=1}^{n} d_{i,h} f_{j,l} F_{i,j,h,l} + \sum_{i=1}^{n} \lambda_i \left(\sum_{j=1}^{n} x_{i,j} - 1 \right) \right\},$$
$$s.t.\ x_{i,j} \in \{0,\ 1\},\ \ F_{i,j,h,l} \geq x_{i,j} + x_{h,l} - 1, \text{ and } (92). \tag{95}$$

To obtain subgradient and surrogate multiplier-updating directions, the linear problems formulation (95) is optimized by using branch-and-cut. In the subgradient method, the relaxed problem (95) is fully optimized. In the surrogate Lagrangian relaxation method, the relaxed problem (95) is optimized approximately subject to the surrogate optimality condition:

$$\sum_{i,j=1}^{n} \sum_{h,l=1}^{n} d_{i,h} f_{j,l} F_{i,j,h,l}^{k+1} + \sum_{i=1}^{n} \lambda_i \left(\sum_{j=1}^{n} x_{i,j}^{k+1} - 1 \right) < \sum_{i,j=1}^{n} \sum_{h,l=1}^{n} d_{i,h} f_{j,l} F_{i,j,h,l}^{k} + \sum_{i=1}^{n} \lambda_i \left(\sum_{j=1}^{n} x_{i,j}^{k} - 1 \right). \tag{96}$$

In practice, the inequality (96) can be operationalized within the commercial solver CPLEX. Given initial values $x_{i,j}^k$ and $F_{i,j,h,l}^k$ as a warm MIP start explained in the beginning of the section, once branch-and-cut finds one solution that is strictly better than $x_{i,j}^k$ and $F_{i,j,h,l}^k$, optimization stops, the surrogate optimality condition is satisfied by definition, and surrogate multiplier-updating directions are computed by using $x_{i,j}^{k+1}$.

Figure 7 demonstrates performance comparison of surrogate Lagrangian relaxation and the subgradient-level method. In the surrogate Lagrangian method, multipliers converge to the optimum with tolerance 0.001 within 800 iterations. In the subgradient-level method, parameters δ_k and B can be chosen to ensure fast convergence within first 200 iterations. However, as stepsizes decrease, it can take many iterations for multipliers to "travel" distance B, thereby leading to slow convergence as multipliers approach λ^*.

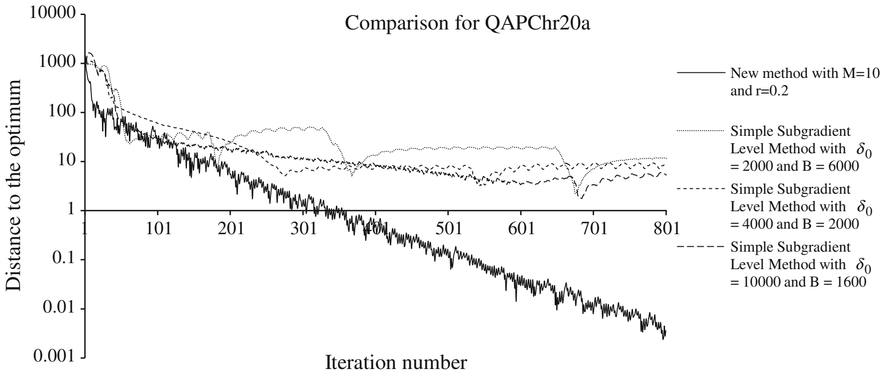


Fig. 7 Comparison of the surrogate Lagrangian relaxation method with parameters $M = 10$ and $r = 0.2$ against the subgradient-level method with parameters: *1* $\delta_0 = 2000$ and $B = 6000$; *2* $\delta_0 = 4000$ and $B = 2000$, and *3* $\delta_0 = 10000$ and $B = 1600$ for the QAPChr20a instance [38]

4 Conclusions

The major breakthrough of this paper is on the development of the novel Surrogate Lagrangian relaxation method and its convergence proof without requiring the optimal dual value and without fully optimizing the relaxed problem. Stepsizes that guarantee convergence without requiring the optimal dual value have been obtained. Under additional assumptions, convergence rate of the new method is proved to be linear. Also, at convergence of the multipliers, the new method generates a valid lower bound. Numerical results demonstrate that the method reduces computational requirements by reducing the effort required to obtain surrogate directions and by alleviating zigzagging of the multipliers. From the application point of view, an important extension of the method would be its combination with other methods in order to efficiently solve mixed-integer programming problems. In particular, the future work would be to prove that the method can be combined with branch-and-cut in order to efficiently solve mixed-integer linear programming problems by exploiting both separability and linearity, thereby resolving the difficulties that frequently accompany pure branch-and-cut.

Acknowledgments This work was supported in part by grants from Southern California Edison and by the National Science Foundation under Grant ECCS–1028870. The authors would like to acknowledge Congcong Wang and Yaowen Yu for their careful perusal of the paper, insightful comments and valuable suggestions during numerous discussions.

References

1. Ermoliev, Y.M.: Methods for solving nonlinear extremal problems. Cybernetics **2**(4), 1–17 (1966)
2. Polyak, B.T.: A general method of solving extremum problems. Sov. Math. Doklady **8**, 593–597 (1967)
3. Polyak, B.T.: Minimization of unsmooth functionals. USSR Comput. Math. Math. Phys. **9**(3), 14–29 (1969). (in Russian)
4. Shor, N.Z.: On the rate of convergence of the generalized gradient method. Cybernetics **4**(3), 79–80 (1968)

5. Shor, N.Z.: Generalized gradient methods for non-smooth functions and their applications to mathematical programming problems. Econ. Math. Methods **12**(2), 337–356 (1976). (in Russian)
6. Goffin, J.-L.: On the finite convergence of the relaxation method for solving systems of inequalities. Operations Research Center Report ORC 71–36, University of California at Berkeley, Berkeley (1971).
7. Goffin, J.-L., Kiwiel, K.: Convergence of a simple subgradient level method. Math. Program. **85**(1), 207–211 (1999)
8. Nedic, A., Bertsekas, D.P.: Convergence rate of incremental subgradient algorithms. In: Uryasev, S., Pardalos, P.M. (eds.) Stochastic Optimization: Algorithms and Applications, pp. 263–304. Kluwer Academic, New York (2000)
9. Nedic, A., Bertsekas, D.: Incremental subgradient methods for nondifferentiable optimization. SIAM J. Optim. **56**(1), 109–138 (2001)
10. Nesterov, Y.: Smooth minimization of non-smooth functions. Math. Program. **103**(1), 127–152 (2005)
11. Bertsekas, D.P.: Incremental gradient, subgradient, and proximal methods for convex optimization: a survey. LIDS Technical Report no. 2848, MIT, (2010).
12. Bertsekas, D.P.: Incremental proximal methods for large scale convex optimization. Math. Program. **129**, 163–195 (2011)
13. Lemarechal, C., Nemirovskii, A.S., Nesterov, Y.E.: New variants of bundle methods. Math. Program. **69**, 111–147 (1995)
14. Kaskavelis, C.A., Caramanis, M.C.: Efficient Lagrangian relaxation algorithms for industry size job-shop scheduling problems. IIE Trans. **30**(11), 1085–1097 (1998)
15. Zhao, X., Luh, P.B., Wang, J.: Surrogate gradient algorithm for Lagrangian relaxation. J. Optim. Theory Appl. **100**(3), 699–712 (1999)
16. Luh, P.B., Blankson, W.E., Chen, Y., Yan, J.H., Stern, G.A., Chang, S.C., Zhao, F.: Payment cost minimization auction for the deregulated electricity markets using surrogate optimization. IEEE Trans. Power Syst. **21**(2), 568–578 (2006)
17. Sun, T., Zhao, Q.C., Luh, P.B.: On the surrogate gradient algorithm for Lagrangian relaxation. J. Optim. Theory Appl. **133**(3), 413–416 (2007)
18. Chang, T.S.: Comments on "Surrogate gradient algorithm for Lagrangian relaxation". J. Optim. Theory Appl. **137**(3), 691–697 (2008)
19. Bragin, M.A., Han, X., Luh, P.B., Yan, J.H.: Payment cost minimization using Lagrangian relaxation and modified surrogate optimization approach. In: Proceedings of the IEEE Power Engineering Society, General Meeting, Detroit, Michigan (2011)
20. Bragin, M.A., Luh, P.B., Yan, J.H., Yu, N., Han, X., Stern, G.A.: An efficient surrogate subgradient method within Lagrangian relaxation for the payment cost minimization problem. In: Proceedings of the IEEE Power Engineering Society, General Meeting, San Diego (2012)
21. Allen, E., Nelgason, R., Kennongton, J., Shettym, B.: A generalization of Polyak's convergence result for subgradient optimization. Math. Program. **37**(3), 309–317 (1987)
22. Bertsekas, D.P.: Nonlinear Programming. Athena Scientific, Massachusetts (2008)
23. Wah, B.W., Chen, Y.X.: Subgoal partitioning and global search for solving temporal planning problems in mixed space. Int. J. Artif. Intell. Tools **13**(4), 767–790 (2004)
24. Chu, P.C., Beasley, J.E.: A genetic algorithm for the generalized assignment problem. Comput. Oper. Res. **24**(1), 17–23 (1997)
25. Yagiura, M., Yamaguchi, T., Ibaraki, T.: A variable depth search algorithm with branching search for the generalized assignment problem. Optim. Methods Softw. **10**, 419–441 (1998)
26. Yagiura, M., Ibaraki, T., Glover, F.: A path relinking approach with ejection chains for the generalized assignment problem. Eur. J. Oper. Res. **169**(2), 548–569 (2006)
27. Avella, P., Boccia, M., Vasilyev, I.: A computational study of exact knapsack separation for the generalized assignment problem. Comput. Optim. Appl. **45**(3), 543–555 (2010)
28. Posta, M., Ferland, J.A., Michelon, P.: An exact method with variable fixing for solving the generalized assignment problem. Comput. Optim. Appl. **52**(3), 629–644 (2012)
29. Özbakir, L., Baykasoglu, A., Tapkan, P.: Bees algorithm for generalized assignment problem. Appl. Math. Comput. **215**(11), 3782–3795 (2010)
30. Asahiro, Y., Ishibashi, M., Yamashita, M.: Independent and cooperative parallel search methods for the generalized assignment problem. Optim. Methods Softw. **18**(2), 129–141 (2003)
31. Laguna, M., Kelly, J.P., Gonzalez-Velarde, J.L., Glover, F.: Tabu search for the multilevel generalized assignment problem. Eur. J. Oper. Res. **82**, 176–189 (1995)

32. Koopmans, T.C., Beckmann, M.J.: Assignment problems and the location of economic activities. Econometrica **25**(1), 53–76 (1957)
33. Dickey, J.W., Hopkins, J.W.: Campus building arrangement using TOPAZ. Transp. Res. **6**, 59–68 (1972)
34. Elshafei, A.N.: Hospital layout as a quadratic assignment problem. Oper. Res. Q. **28**, 167–179 (1977)
35. Geoffrion, A.M., Graves, G.W.: Scheduling parallel production lines with changeover costs: practical applications of a quadratic assignment/LP approach. Oper. Res. **24**, 595–610 (1976)
36. Krarup, J., Pruzan, P.M.: Computer-aided layout design. Math. Program. Study **9**, 75–94 (1978)
37. Burkard, R.E., Offermann, J.: Entwurf von Schreibmaschinentastaturen mittels quadratischer Zuordnungsprobleme. Zeitschrift fur Oper. Res. **21**(4), 121–132 (1977). (in German)
38. Christofides, N., Benavent, E.: An exact algorithm for the quadratic assignment problem. Oper. Res. **37**(5), 760–768 (1989)