Journal of Control and Decision

ISSN: 2330-7706 (Print) 2330-7714 (Online) Journal homepage: http://www.tandfonline.com/loi/tjcd20

An efficient approach for solving mixed-integer programming problems under the monotonic condition

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

Journal of Control and Decision, 2016
Vol. 3, No. 1, 44–67, http://dx.doi.org/10.1080/23307706.2015.1129916

Taylor & Francis
Taylor & Francis Group

An efficient approach for solving mixed-integer programming problems under the monotonic condition

Mikhail A. Bragin[a*], Peter B. Luh[a], Joseph H. Yan[b] and Gary A. Stern[b]

[a]*Department of Electrical and Computer Engineering, University of Connecticut, Storrs, CT, USA;* [b]*Southern California Edison, Rosemead, CA, USA*

(*Received 22 September 2015; accepted 7 December 2015*)

Many important integer and mixed-integer programming problems are difficult to solve. A representative example is unit commitment with combined cycle units and transmission capacity constraints. Complicated transitions within combined cycle units are difficult to follow, and system-wide coupling transmission capacity constraints are difficult to handle. Another example is the quadratic assignment problem. The presence of cross-products in the objective function leads to nonlinearity. In this study, building upon the novel integration of surrogate Lagrangian relaxation and branch-and-cut, such problems will be solved by relaxing selected coupling constraints. Monotonicity of the relaxed problem will be assumed and exploited and nonlinear terms will be dynamically linearised. The linearity of the resulting problem will be exploited using branch-and-cut. To achieve fast convergence, guidelines for selecting stepsizing parameters will be developed. The method opens up directions for solving nonlinear mixed-integer problems, and numerical results indicate that the new method is efficient.

Keywords: integer monotonic programming; mixed-integer monotonic programming; branch-and-cut; surrogate Lagrangian relaxation

1. Introduction

Many large systems are created by interconnecting smaller subsystems with system-wide coupling constraints, and problems involving such systems are formulated as mixed-integer programming (MIP)[1] problems. Many such problems are modelled using monotonic[2] objective functions and linear constraints. A representative example of a MIP problem with a linear objective function is the unit commitment problem (Guan, Luh, Yan, & Amalfi, 1992; Guan, Luh, Yan, & Rogan, 1994) with combined cycle units[3] (Alemany, Moitre, Pinto, & Magnago, 2013; Anders, 2005) subject to linear system-wide coupling system demand and transmission constraints. Lagrangian relaxation (Fisher, 1973, 1981; Geoffrion, 1974; Guan et al., 1992, 1994) was historically used to solve this problem by relaxing system-wide demand coupling constraints and decomposing the relaxed problem into subproblems. However, subgradient methods (Ermoliev, 1966; Goffin & Kiwiel, 1999; Polyak, 1967; Shor, 1968, 1976) traditionally used to coordinate subproblem solutions require solving all subproblems thereby leading to zigzagging of multipliers and slow convergence. The recent trend to solve such problems is by exploiting linearity using branch-and-cut (Balas, Ceria, & Cornuéjols,

*Corresponding author. Email: mab08017@engr.uconn.edu

1996; Hoffman & Padberg, 1993; Padberg & Rinaldi, 1991), which has been successful for solving many problems. However, the method handles all constraints 'globally', and 'local' subsystem characteristics of the problem are not exploited thereby leading to difficulties when solving the unit commitment problem with combined cycle units because complicated state transitions[4] within a combined cycle unit are handled 'globally' and affect the solution process of the entire problem thereby leading to slow convergence (Bragin, Luh, Yan, & Stern, 2014, 2015b). A representative example of a MIP problem with a nonlinear monotonic objective function is the quadratic assignment problem subject to linear system-wide assignment constraints (Burkard & Offermann, 1977; Dickey & Hopkins, 1972; Elshafei, 1977; Geoffrion & Graves, 1976; Koopmans & Beckmann, 1957; Krarup & Pruzan, 1978).[5,6] The presence of cross-products of binary decision variables in the objective function makes the problem nonlinear and nonseparable. Standard methods to solve the problem are the taboo search (Taillard, 1991) and the genetic algorithm (Tate & Smith, 1995), and one possible way to solve the problem is through linearisations (Xia & Yuan, 2006). Such linearisations are frequently accompanied by the introduction of new constraints and decision variables. While standard branch-and-cut suffers from slow convergence as discussed before, some of the difficulties will be efficiently overcome as will be explained in the following paragraph.

In this study, to solve nonlinear MIP problems with monotonic objective functions and linear constraints, the new method will be developed. To provide the foundation for the new method, the synergistic integration of surrogate Lagrangian relaxation and branch-and-cut (Bragin, Luh, Yan, & Stern, 2015c) will be briefly reviewed in Section 2. Surrogate Lagrangian relaxation (Bragin, Luh, Yan, Yu, & Stern, 2015a) reduces computational requirements subject only to the simple 'surrogate optimality condition', which can be easily satisfied by solving only one or few subproblems to update multipliers thereby alleviating zigzagging of multipliers and convergence has been constructively proved without requiring the optimal dual value. Each subproblem can be efficiently solved using branch-and-cut without affecting the solution process of the entire problem. Moreover, the computational complexity can be further reduced by exploiting the fact that multipliers only affect subproblem objective functions without affecting subproblem constraints, and therefore without affecting subproblem 'convex hulls'. Since subproblems are much smaller in size as compared to the original problem, subproblem convex hulls are much easier to obtain. Once obtained, such invariant convex hulls can be reused in subsequent iterations and solving subproblems becomes very easy. If convex hulls cannot be obtained, reusing cuts generated by branch-and-cut in previous iterations can also reduce the computational effort in subsequent iterations.

To efficiently solve nonlinear MIP problems with monotonic objective functions and linear constraints, the new method will be developed in Section 3. The new method resolves the nonlinearity difficulty by first relaxing system-wide coupling constraints and then by exploiting monotonicity of resulting subproblems through the dynamic linearisation. Since the objective function of the relaxed problem consists of the monotonic objective function of the original problem and the part that is associated with relaxed system-wide constraints, monotonicity of the relaxed problem objective function can be ensured by selectively relaxing system-wide constraints. After such selective relaxation, objective functions of subproblems are monotonic and this monotonicity is exploited to prove that by optimising dynamically linearised subproblems, the 'surrogate optimality condition' is satisfied thereby leading to convergence. Since resulting subproblems are linear and much smaller in size and complexity as compared to the original problem, they can be efficiently solved using branch-and-cut. Moreover, since multipliers and dynamic linearisation affect subproblem objective functions only without affecting subproblems constraints and without affecting subproblem convex hulls when all system-wide coupling constraints are relaxed, the invariability of subproblem convex hulls can be exploited to improve computational efficiency as explained in the previous paragraph. While subproblems convex hulls may no longer be invariant when system-wide coupling constraints are relaxed selectively, conceptually, cuts generated by branch-and-cut based on subproblem constraints only can still be reused to reduce the computational effort in subsequent iterations. Lastly, while convergence does not require the knowledge of the optimal dual value, convergence may require a large number of iterations, especially when solving nonlinear problems because successive linearisations may involve many iterations. To improve convergence, stepsize-updating parameters are adaptively adjusted and stepsizes are re-initialised. The selective relaxation of constraints and the novel stepsizing guidelines, developed in Section 3, can also be used to efficiently solve linear MIP problems.

In Section 4, by considering a small nonlinear example, it is demonstrated that novel stepsize-updating is more effective. By considering the unit commitment problem with combined cycle units and transmission capacity constraints, it is demonstrated that the new method can efficiently solve linear problems without full decomposition. Lastly, by considering quadratic assignment problems, it is demonstrated that the new method is efficient and scalable.

2. Synergistic integration of surrogate Lagrangian relaxation and branch-and-cut for solving mixed-integer linear programming problems

2.1. Mixed-integer linear programming

Consider the following mixed-integer linear programming problem (Padberg, 2005):

$$\min_{x,y}\{cx + dy\}, x \in \mathbb{Z}^n, y \in \mathbb{R}^p, x \geq 0, y \geq 0, \tag{1}$$

$$\text{s.t.} \quad Ax + Ey \leq b, \tag{2}$$

where c, d and b are $1 \times n$, $1 \times p$ and $m \times 1$ vectors, $\mathbb{Z}$ and $\mathbb{R}$ are sets of integers and real numbers, and A and E are $m \times n$ and $m \times p$ matrices, respectively.

2.2. Surrogate Lagrangian relaxation (Bragin et al., 2015a)

In the method, after relaxing constraints (2) by introducing Lagrange multipliers $\lambda^T = (\lambda_1, \ldots, \lambda_m) \in \mathbb{R}^m$, the *Lagrangian function* is formed:

$$L(\lambda, x, y) = cx + dy + \lambda^T(Ax + Ey - b). \tag{3}$$

The relaxed problem, resulting from minimising the Lagrangian function (3), needs to be fully optimised

$$\min_{x,y} L(\lambda, x, y). \tag{4}$$

To reduce computational requirements without fully optimising the relaxed problem (4) while ensuring that 'surrogate' multiplier-updating directions form acute angles with directions towards λ^*, the method requires the satisfaction of the only simple 'surrogate optimality condition' (Zhao, Luh, & Wang, 1999):

$$\tilde{L}(\lambda^k, x^k, y^k) < \tilde{L}(\lambda^k, x^{k-1}, y^{k-1}). \tag{5}$$

Here, $\widetilde{L}$ is a *surrogate dual value*, which is a Lagrangian function (3) evaluated at a solution (x^k, y^k):

$$\tilde{L}(\lambda, x^k, y^k) = cx^k + dy^k + \lambda^T\left(Ax^k + Ey^k - b\right). \tag{6}$$

Within the method, Lagrange multipliers are updated as:

$$\lambda^{k+1} = \left[\lambda^k + c^k\left(Ax^k + Ey^k - b\right)\right]^+, \quad k = 0, 1, \ldots, \tag{7}$$

where $[]^+$ is the projection onto the positive orthant that guarantees dual feasibility, and c^k are positive scalar stepsizes. To ensure convergence to λ^*, stepsizes are updated as:

$$c^k = \alpha_k \frac{c^{k-1}\left\|\tilde{g}\left(x^{k-1}, y^{k-1}\right)\right\|}{\left\|\tilde{g}(x^k, y^k)\right\|}, \ 0 < \alpha_k < 1, \ k = 1, 2, \ldots, \tag{8}$$

where $\tilde{g}\left(x^k, y^k\right) = Ax^k + Ey^k - b$ are surrogate subgradient directions. The surrogate subgradient norm-squared can be represented as:

$$\left\|\tilde{g}\left(x^k, y^k\right)\right\|^2 = \left\|\max\left(0, \tilde{g}\left(x^k, y^k\right)\right)\right\|^2. \tag{9}$$

Since surrogate Lagrangian relaxation does not require fully minimising the relaxed problem (4), surrogate subgradient directions do not change drastically, and they are generally smoother as compared to subgradient directions, thereby alleviating zigzagging and reducing the number of iterations required for convergence.

It is possible that during the iterative process of the new method, surrogate subgradient norms become zero. While zero-subgradients imply that the optimal solution is found, zero-surrogate subgradients only imply that a feasible solution is found, and generally, the algorithm needs to proceed. However, the stepsizing formula (8) involves the division by zero. To resolve this issue, a small value is added to the surrogate subgradient in the denominator, and the stepsizing formula is modified as:

$$c^k = \alpha_k \frac{c^{k-1}\left\|\tilde{g}\left(x^{k-1}, y^{k-1}\right)\right\|}{\left\|\tilde{g}(x^k, y^k)\right\| + \varepsilon}, \ 0 < \alpha_k < 1. \tag{10}$$

Stepsizing formula (8) has been developed by Bragin et al., 2015a, and convergence was proved. One possible way to select stepsize-updating parameters α_k that guarantee convergence is

$$\alpha_k = 1 - \frac{1}{Mk^p}, \ p = 1 - \frac{1}{k^r}, \ M \geq 1, \ 0 < r < 1, \ k = 2, 3, \ldots \tag{11}$$

In the formula (11), parameters M and r control how fast α_k approach 1 thereby affecting how fast stepsizes approach zero, and ultimately controlling how fast multipliers converge to λ^*. On the one hand, when M and r are large, α_k approaches 1 fast and stepsizes approach zero slowly. As a result, stepsizes stay large thereby leading to oscillation of multipliers in the neighbourhood of the optimum. On the other hand, when M and r are small, stepsizes approach zero fast and become small early in the iterative process, thereby also requiring many iterations to reach λ^*.

Another important parameter within the method is the initial stepsize, which can be initialised according to Bragin et al., 2015a as:

$$c^0 = \frac{\hat{q} - q(\lambda^0)}{\|g(x^0, y^0)\|^2}, \tag{12}$$

where $\hat{q}$ is an estimate of the optimal dual value, $q(\lambda^0)$ is the dual value obtained by fully optimising the relaxed problem (4) and $g(x^0, y^0)$ is the corresponding subgradient direction. The difficulty of initialising stepsizes using (12) is that the estimate of the optimal dual value may be too large or too small thereby leading to slow convergence.

2.3. Synergistic integration of surrogate Lagrangian relaxation and branch-and-cut (Bragin et al., 2015c)

Frequently, mixed-integer linear programming problems model large systems that are created by interconnecting smaller subsystems using system-wide coupling constraints. Recently, to solve such problems by exploiting this particular structure, surrogate Lagrangian relaxation has been synergistically integrated with branch-and-cut.

Assumption 2.1. Particular Problem Structure. Subsystems are modelled using discrete decision variables $x_i \in \mathbb{Z}^{n_i}$ and continuous decision variables $y_i \in \mathbb{R}^{p_i}$ and are subject to 'local' subsystem constraints:

$$A_i x_i + E_i y_i \leq b_i, \;\; i = 1, \ldots, I. \tag{13}$$

Subsystems are coupled across the entire system through the use of system-wide coupling constraints:

$$A^0 x + E^0 y \leq b^0, \tag{14}$$

where b^0 is an $m_0 \times 1$ vector, A^0 and E^0 are $m_0 \times n$ and $m_0 \times p$ matrices and $x = (x_1, \ldots, x_I)$, $y = (y_1, \ldots, y_I)$. Constraints (14) can be written as m_0 constraints that couple subproblem decision variables x_i and y_i in the following way:

$$\sum_{i=1}^{I} \left(A^0_{j,i} x_i + E^0_{j,i} y_i \right) \leq b^0_j, \; j \in \{1, \ldots, m_0\}. \tag{15}$$

Constraints (13)–(15) are essentially constraints (2) with

$$A = \begin{pmatrix} . & A^0 & . \\ A_1 & . & 0 \\ . & . & . \\ 0 & . & A_I \end{pmatrix}, E = \begin{pmatrix} . & E^0 & . \\ E_1 & . & 0 \\ . & . & . \\ 0 & . & E_I \end{pmatrix}, b = \begin{pmatrix} b^0 \\ b_1 \\ . \\ b_I \end{pmatrix},$$
$$A^0 = \begin{pmatrix} A^0_{1,1} & . & A^0_{1,I} \\ . & . & . \\ A^0_{m_0,1} & . & A^0_{m_0,I} \end{pmatrix}, E^0 = \begin{pmatrix} E^0_{1,1} & . & E^0_{1,I} \\ . & . & . \\ E^0_{m_0,1} & . & E^0_{m_0,I} \end{pmatrix}, b^0 = \begin{pmatrix} b^0_1 \\ . \\ b^0_{m_0} \end{pmatrix}. \tag{16}$$

Here, A_i and E_i are $m_i \times n_i$ and $m_i \times p_i$ matrices, $A^0_{j,i}$ and $E^0_{j,i}$ are $1 \times n_i$ and $1 \times p_i$ vectors, $b^0 = \left(b^0_1, \ldots, b^0_{m_0} \right)$ is an $m_0 \times 1$ vector, and b_i are $m_i \times 1$ column vectors such that $n_1 + \cdots + n_I = n$, $p_1 + \cdots + p_I = p$ and $m_0 + \cdots + m_I = m$. □

Under Assumption 2.1, the class of optimisation problems (1)–(2) can be represented as smaller subsystems subject to subsystem constraints (13) and such subsystems are coupled by system-wide coupling constrains (15). After relaxing system-wide coupling constraints (15), the relaxed problem can be decomposed into I subproblems, and each subproblem i (= 1, … ,I) can be written as:

$$\min_{x_i,y_i}\{c_ix_i + d_iy_i\} + \sum_j \lambda_j(A^0_{j,i}x_i + E^0_{j,i}y_i - b^0_j), \text{ s.t. } (15), x_i \in \mathbb{Z}^{n_i}, y_i \in \mathbb{R}^{p_i}, x_i \geq 0, y_i \geq 0. \tag{17}$$

Each subproblem (17) is much smaller in size and complexity as compared to the original problem (1)–(2), and each subproblem can be efficiently solved by branch-and-cut without affecting the solution process of the entire problem. Moreover, the computational efficiency can be further improved by exploiting the fact that multipliers affect subproblem objective functions without affecting subproblem constraints and without affecting subproblem 'convex hulls'. Since subproblems are much smaller in size as compared to the original problem, subproblem convex hulls are much easier to obtain. Once obtained, such invariant convex hulls can be reused in subsequent iterations and solving subproblems becomes very easy by an appropriate LP solver. To obtain feasible solutions to the original problem (1)–(2), subproblem solutions are adjusted to satisfy violated constraints. The method has been used to solve the unit commitment problem with combined cycle units without transmission constraints and the generalised assignment problem, and great results have been obtained (Bragin et al., 2014, 2015b, 2015c).

3. An efficient approach for solving MIP problems under the monotonic condition

In subsection 3.1, under the assumption of monotonicity of the objective function and linearity of constraints, building upon the integration of surrogate Lagrangian relaxation and branch-and-cut presented in Section 2, the novel methodology will be developed to solve nonlinear MIP problems without fully exploiting separability and through the use of dynamic linearisation in subsection 3.1. In subsection 3.2, guidelines for updating stepsizes will be developed to achieve fast convergence.

3.1. An efficient approach for solving nonlinear MIP problems under the monotonic condition

To solve nonlinear MIP problems under the monotonicity assumption of the objective function and linearity of constraints, building upon the integration of surrogate Lagrangian relaxation and branch-and-cut, the new method will be developed based on the exploitation of problem structure after selective relaxation of system-wide constraints, and monotonicity of resulting subproblems through a dynamic linearisation while efficiently coordinating subproblem solutions and guaranteeing convergence. The monotonicity of the relaxed problem objective function can be ensured by selectively relaxing system-wide constraints. After such selective relaxation, objective functions of subproblems are monotonic and this monotonicity is exploited to prove that by optimising dynamically linearised subproblems, the 'surrogate optimality condition' is satisfied thereby leading to convergence.

Consider the following MIP problem with a nonlinear objective function:

$$\min_{x,y} f(x,y), \ \text{s.t.}(13),(15), \ x \in \mathbb{Z}^n, \ y \in \mathbb{R}^p, \ x \geq 0, y \geq 0. \tag{18}$$

In order to exploit the particular problem structure mentioned in Assumption 2.1, the objective function of (18) is assumed to be separable.

Assumption 3.1. Separability. Assume that the function $f(x,y)$ can be represented as:

$$f(x,y) = \sum_{i=1}^{I} f_i(x_i, y_i). \tag{19}$$

Additionally, assume that each function $f_i(x_i,y_i)$ is monotonic and the monotonicity is defined as follows:

Definition 3.2. Monotonicity. The function $f_i(x_i,y_i)$ is monotonically increasing in one of its components $x_{i,j}$ if and only if

$$f_i(x_{i,1}, x_{i,2}, \ldots, x_{i,j-1}, a, x_{i,j+1}, \ldots, x_{i,n_i}, y_i) \le f_i(x_{i,1}, x_{i,2}, \ldots, x_{i,j-1}, a, x_{i,j+1}, \ldots, x_{i,n_i}, y_i), \quad \text{for } a \le b, \tag{20a}$$

where $y_i = (y_{i,1}, \ldots, y_{i,p_i})$. Likewise, function $f_i(x_i,y_i)$ is monotonically decreasing in one of its components $x_{i,j}$ if and only if

$$f_i(x_{i,1}, x_{i,2}, \ldots, x_{i,j-1}, a, x_{i,j+1}, \ldots, x_{i,n_i}, y_i) \ge f_i(x_{i,1}, x_{i,2}, \ldots, x_{i,j-1}, a, x_{i,j+1}, \ldots, x_{i,n_i}, y_i), \quad \text{for } a \le b. \tag{20b}$$

Function $f_i(x_i,y_i)$ is monotonically increasing (decreasing) if it is increasing (decreasing) in all of its variables. Monotonicity of the function $f_i(x_i,y_i)$ in $y_{i,j}$ can be defined in a similar way. □

Generally, the class of problems (18) does not belong to the class of biconvex problems (Gorski, Pfeuffer, & Klamroth, 2007; Li, Wen, Zheng, & Zhao, 2015) since objective function in (18) it is not defined over a convex set because of the presence of integer variables x. Also, the class of problems (18) is broader than the class of bilinear problems (Al-Khayyal, 1990; Li, Wen, & Zhang, 2015) because function $f(x,y)$ can generally be any monotonic function.

To solve the problem (18), system-wide coupling constraints (15) are first relaxed, and under Assumption 3.1, the relaxed problem becomes:

$$\min_{x,y}\left\{\sum_{i=1}^{I} f_i(x_i,y_i) + \sum_j \lambda_j \left(\sum_{i=1}^{I}\left(A_{j,i}^0 x_i + E_{j,i}^0 y_i\right) - b_j^0\right)\right\}, \ \text{s.t.}(13), x_i \in \mathbb{Z}^{n_i}, y_i \in \mathbb{R}^{p_i}, \lambda_j \in \mathbb{R}, x_i \ge 0, y_i \ge 0, \lambda_j \ge 0. \tag{21}$$

The relaxed problem is then decomposed into I subproblems, and each subproblem i (= 1, … , I) is

$$\min_{x_i,y_i}\left\{f_i(x_i,y_i) + \sum_j \lambda_j (A_{j,i}^0 x_i + E_{j,i}^0 y_i)\right\}, \text{s.t.}(13), x_i \in \mathbb{Z}^{n_i}, y_i \in \mathbb{R}^{p_i}, \lambda_j \in \mathbb{R}, x_i \ge 0, y_i \ge 0, \lambda_j \ge 0. \tag{22}$$

Multipliers are updated to coordinate subproblem solutions after solving only one or few such subproblems. However, there are difficulties that accompany this approach. First, functions $f_i(x_i,y_i)$ are nonlinear, and as a result, subproblems (22) are nonlinear and they cannot be solved by branch-and-cut. Second, while functions $f_i(x_i,y_i)$ are monotonic by assumption, the subproblem objective function in (22) is the sum of $f_i(x_i, y_i)$ and $\sum \lambda_j(A^0_{j,i}x_i + E^0_{j,i}y_i)$, and the subproblem objective function may not be monotonic. Third, if the number of system-wide coupling constraints in (15) is large, a large number of multipliers may be required in (21) thereby leading to slow convergence. Relaxation of many constraints may also result in a loose lower bound provided by the dual value, and this bound may not be tight enough to provide a good measure of solution quality. Additionally, searching for feasible solutions may be problematic.

3.1.1. *Linearisation procedure*

The first nonlinearity difficulty will be resolved by dynamically linearising $f_i(x_i,y_i)$ in two steps as follows. In the first step, nonlinear terms that depend on several variables will be linearised by selecting one variable as a decision variable and by fixing the remaining variables at the previously obtained solution (x_i^{k-1}, y_i^{k-1}). The resulting terms become functions of a single variable. In the second step, nonlinear terms of $f_i(x_i, y_i)$ that depend on a single variable will then be linearised using linear terms of Taylor series around the previously obtained solution (x_i^{k-1}, y_i^{k-1}). Decision variables are positive, and the linear part of Taylor series is an increasing function whenever the original function is increasing. The resulting linear function $f_i(x_i,y_i)$ is guaranteed to be monotonically increasing if the original function $f_i(x_i,y_i)$ is monotonically increasing. A similar argument holds for decreasing functions.

Since $f_i(x_i,y_i)$ is a function of several variables, additional iterations are required to perform the first step of linearisation with respect to all the remaining variables. To speed up the process, a function is constructed by taking the average value over all possible linearised functions. For example, consider a nonlinear function $f(x_1,x_2,x_3) = x_1x_2x_3$, x_1, x_2, $x_3 \in \{0,1\}$. The linearised function then becomes:

$$\bar{f}(x_1,x_2,x_3) = \frac{x_1^{k-1}x_2^{k-1}x_3 + x_1^{k-1}x_2x_3^{k-1} + x_1x_2^{k-1}x_3^{k-1}}{3}. \tag{23}$$

Since all system-wide coupling constraints are relaxed in (21) and the linearisation described above only affects subproblem objective functions in (22) without affecting subproblem constraints and without affecting subproblem convex hulls, the invariability of subproblem convex hulls can also be exploited as explained in Section 2.

3.1.2. *Selective relaxation of system-wide constraints*

To resolve the second difficulty associated with the possible loss of monotonicity of the subproblem objective function in (22), the following Proposition 3.3 provides a criterion to select system-wide constraints in (15) to be relaxed.[7]

Proposition 3.3. Monotonicity of subproblem objective functions. Suppose $f_i(x_i,y_i)$ is an increasing function and $\Omega \in \{1, \ldots, m_0\}$ is a subset of constraint indices j such that terms $A^0_{j,i}x_i + E^0_{j,i}y_i$ in (22) are increasing functions. Then, after the relaxation of constraints from (15) such that $j \in \Omega$, the subproblem objective function in (22) is monotonically increasing. □

Proof: Since $f_i(x_i,y_i)$ is monotonically increasing, $A^0_{j,i}x_i + E^0_{j,i}y_i$ are increasing for $j \in \Omega$, and multipliers are non-negative, and the objective function (22) is monotonically increasing. □

In a similar fashion, it can be proved that when the function $f_i(x_i,y_i)$ is monotonically decreasing, the subproblem objective function in (22) will be monotonically decreasing after relaxing constraints (15) such that $A^0_{j,i}x_i + E^0_{j,i}y_i$ are decreasing.

Consider a linearised subproblem i (= 1, … , I) that is created after relaxing system-wide constraints with indices $j \in \Omega$ and after linearising the function $f_i(x_i,y_i)$ according to the Linearisation Procedure[8]:

$$\min_{x_i,y_i}\left\{\bar{f}_i(x_i,y_i) + \sum_{j\in\Omega}\lambda^k_j(A^0_{j,i}x_i + E^0_{j,i}y_i)\right\},\ x_i \in \mathbb{Z}^{n_i},\ y_i \in \mathbb{R}^{p_i},\ \lambda^k_j \in \mathbb{R},\ x_i \geq 0,\ y_i \geq 0,\ \lambda_j \geq 0, \tag{24}$$

$$\text{s.t. } A_i x_i + E_i y_i \leq b_i, \tag{25a}$$

$$\sum_{l=1}^{I}\left(A^0_{j,l}x_l + E^0_{j,l}y_l\right) \leq b^0_j,\ j \notin \Omega. \tag{25b}$$

This linearised subproblem i is also minimised subject to the following linear version of the surrogate optimality condition (5) for subproblems:

$$\bar{f}_i(x^k_i,y^k_i) + \sum_{j\in\Omega}\lambda^k_j(A^0_{j,i}x^k_i + E^0_{j,i}y^k_i) < \bar{f}_i(x^{k-1}_i,y^{k-1}_i) + \sum_{j\in\Omega}\lambda^k_j(A^0_{j,i}x^{k-1}_i + E^0_{j,i}y^{k-1}_i). \tag{26}$$

The convex hull corresponding to the linearised subproblem (24)–(25) obtained by selective relaxation of system-wide constraints may no longer be invariant because system-wide coupling constraints (25b) depend on decision variable values other than x_i and y_i, and such decision variable values are changing throughout the iterative process. Still, since linearised subproblem constraints (25a) do not change throughout the iterative process, cuts generated by branch-and-cut based on constraints (25a) only can be reused to reduce the computational effort in subsequent iterations and solving subproblems with cuts retained from previous iterations will be easier than starting from ground zero. However, since branch-and-cut generates cuts based on all constraints (25a) and (25b), the reusing cuts may be difficult in practical implementations.

The monotonicity of subproblem objective functions will now be exploited to establish that under condition (26), the surrogate optimality condition for the relaxed problem (21) is satisfied in Proposition 3.4 and that the overall method converges in Theorem 3.5 below.

Proposition 3.4. Satisfaction of the Surrogate Optimality Condition. Solutions (x^k_i, y^k_i) to the linearised subproblem (24)–(25) that satisfy (26), also satisfy the surrogate optimality condition for the relaxed problem:

$$\begin{aligned} f(x^k,y^k) + \sum_{j\in\Omega}\lambda^k_j\left(\sum_{i=1}^{I}\left(A^0_{j,i}x^k_i + E^0_{j,i}y^k_i\right) - b^0_j\right) &< f(x^{k-1},y^{k-1}) \\ &+ \sum_{j\in\Omega}\lambda^k_j\left(\sum_{i=1}^{I}\left(A^0_{j,i}x^{k-1}_i + E^0_{j,i}y^{k-1}_i\right) - b^0_j\right). \end{aligned} \tag{27}$$

Proof: Suppose that one subproblem i is solved at a time and condition (26) is satisfied as a strict inequality.[9] Since subproblems other than subproblem i are not solved, their solutions (x^k_s, y^k_s) satisfy

$$\bar{f}_s(x_s^k, y_s^k) + \sum_{j\in\Omega} \lambda_j^k (A_{j,s}^0 x_s^k + E_{j,s}^0 y_s^k) = \bar{f}_s(x_s^{k-1}, y_s^{k-1}) + \sum_{j\in\Omega} \lambda_j^k (A_{j,s}^0 x_s^{k-1} + E_{j,s}^0 y_s^{k-1}). \quad (28)$$

Adding (26) and (28) for all s leads to the linearised surrogate optimality condition for the relaxed problem:

$$\bar{f}(x^k, y^k) + \sum_{j\in\Omega} \lambda_j^k \left(\sum_{i=1}^{I} \left(A_{j,i}^0 x_i^k + E_{j,i}^0 y_i^k \right) - b_j^0 \right) < \bar{f}(x^{k-1}, y^{k-1}) + \sum_{j\in\Omega} \lambda_j^k \left(\sum_{i=1}^{I} \left(A_{j,i}^0 x_i^{k-1} + E_{j,i}^0 y_i^{k-1} \right) - b_j^0 \right). \quad (29)$$

From the Linearisation Procedure and Proposition 3.3, it follows that the objective function of the relaxed problem (21) with $j \in \Omega$ and the linearised objective function $\bar{f}(x,y) + \sum_{j\in\Omega} \lambda_j^k \left(\sum_{i=1}^{I} \left(A_{j,i}^0 x_i + E_{j,i}^0 y_i \right) - b_j^0 \right)$ are increasing functions. Therefore, solutions (x^k, y^k) and (x^{k-1}, y^{k-1}) that satisfy (29) will also satisfy (27). Lastly, since $\bar{f}(x,y)$ is constructed by fixing variables at (x^{k-1}, y^{k-1}) and using the Taylor series expansion, the following equality holds: $\bar{f}\left(x^{k-1}, y^{k-1}\right) = f\left(x^{k-1}, y^{k-1}\right)$. Therefore, right-hand sides of (29) and (27) are equal, and the proposition is proved. □

In a similar fashion, it can be proved that the surrogate optimality condition holds when the function $f_i(x_i, y_i)$ is monotonically decreasing.

Theorem 3.5. Convergence of the new method. Suppose that each function $\bar{f}_i(x_i, y_i)$, $i = 1, \ldots, I$ is constructed from a monotonic function $f_i(x_i, y_i)$ using the Linearisation Procedure, and the linear version of the surrogate optimality condition (26) for subproblems is satisfied. If stepsizes c^k satisfy (8) and stepsize-updating parameters α_k satisfy (9), then multipliers (7) converge to λ^*. □

Proof: Per Proposition 3.4, the surrogate optimality condition is satisfied. As reviewed in Section 2, since surrogate multiplier-updating directions satisfy the surrogate optimality condition, stepsizes satisfy (8), and stepsize-updating parameters α_k satisfy (9), multipliers (7) converge to λ^*. □

In the presence of equality constraints, multipliers corresponding to equality constraints are not restricted to be non-negative and the monotonicity of subproblem objective function may not be preserved. To ensure convergence of the method in the presence of equality constraints, equality constraints that correspond to negative multipliers are not relaxed. The idea can be operationalised fairly easily by first relaxing constraints as discussed in Proposition 3.3, while projecting all the multipliers onto the positive orthant as in (7). If during the iterative process some of the multipliers corresponding to equality constraints become zero, the respective constraints are put back into the formulation.

3.2. *Adaptive adjustment and re-initialisation of stepsizes*

As discussed before, the choice of stepsize-updating parameters M and r may lead to the large number of iterations and slow convergence. In this subsection, the number of iterations required for convergence will be reduced by adaptively adjusting stepsize-updating parameters M and r and re-initialising stepsizes thereby making sure that stepsizes are large enough to reach the optimum quickly and by alleviating oscillations of multipliers near the optimum.

3.2.1. Adaptive adjustment of stepsize-updating parameters

To improve convergence of the method developed in subsection 3.1, novel guidelines for setting α_k will now be developed. By considering non-increasing series $\{M_k\}$ and $\{r_k\}$ with large M_0 and r_0, stepsizes are large in the beginning of the iterative process thereby allowing multipliers to reach the neighbourhood of the optimum relatively fast. But, as parameters $\{M_k\}$ and $\{r_k\}$ decrease, stepsize-updating parameters α_k approach 1 slower. As a result, stepsizes approach zero faster, thereby alleviating oscillations near the optimum as proved in the following Proposition 3.6.

Proposition 3.6: If stepsize-updating parameters α_k are updated as follows:

$$\alpha_k = 1 - \frac{1}{M_k k^p}, \; p = 1 - \frac{1}{k^{r_k}}, \; M_k \geq 1, \; 0 \leq r_k \leq 1, \; k = 2, 3, \ldots, \tag{30}$$

where M_k and r_k are monotonically non-increasing such that:

$$M_k \to M \geq 1, \; r_k \to r \geq \delta > 0 \tag{31}$$

then the multipliers converge to λ^*.

Proof: Since parameters M_k and r_k in (31) satisfy conditions in (11) for all possible values k, stepsizes with parameters α_k defined in (30) ensure convergence to λ^*. □

3.2.2. Stepsize re-initialisation

In practical implementations, a feasible cost[10] can be used as an estimate of the optimal dual value to initialise stepsizes in (12). However, when an initial estimate is too large, stepsizes may stay large thereby leading to oscillations of multipliers near the optimum. To alleviate this issue, stepsizes are to be re-initialised during the iterative process. For example, if at iteration k the following condition holds

$$c^k > \frac{\hat{q}^k - q(\lambda^k)}{\|g(x^k, y^k)\|^2}, \tag{32}$$

then stepsizes need to be reset as:

$$c^k = \frac{\hat{q}^k - q(\lambda^k)}{\|g(x^k, y^k)\|^2}. \tag{33}$$

This re-initialisation will decrease stepsizes thereby alleviating the oscillations of multipliers near the optimum. Stepsizes may also become too small during the process such that is the following condition holds

$$c^k << \frac{\hat{q}^k - q(\lambda^k)}{\|g(x^k, y^k)\|^2}. \tag{34}$$

In this case, stepsizes are also re-initialised per (33).

Proposition 3.7: The method developed in subsection 3.1 will converge if stepsizes are re-initialised per (33) finite number of times.

Proof: Since the number of re-initialisations is finite, the analysis of convergence after the last re-initialisation follows the same logic as in Proposition 3.6.

Since the method presented in Section 2 uses the same stepsizing formula as the method developed in subsection 3.1, the results obtained in this subsection 3.2 also hold for the method presented in Section 2.

4. Numerical section

The new method is implemented using CPLEX 12.6 on a laptop Intel® Core™ i7 CPU Q720 @ 1.60 GHz and 4.00 GB of RAM. To illustrate convergence under Propositions 3.6–3.7, a small nonlinear integer example is considered in Example 1. To illustrate the efficiency of the method for solving mixed-integer linear programming problems without fully exploiting separability, the unit commitment problem with combined cycle units and transmission capacity constraints is considered in Example 2. To illustrate the efficiency and the scalability of the method for solving integer nonlinear programming problems under the monotonic condition, quadratic assignment problem is considered in Example 3.

Example 1. A Nonlinear integer problem. The purpose of this example is to demonstrate convergence with stepsize-setting parameters that satisfy Proposition 3.6. To achieve this goal, the following small and relatively simple integer nonlinear programming example subject to linear constraints with known optimal value is considered:

$$\min_{\{x_1,x_2,x_3,x_4,x_5,x_6\}\in Z_+\cup\{0\}} \{x_1^2+x_2^2+x_3^2+x_4^2+x_5^2+x_6^2\} \tag{35}$$

$$\begin{aligned}&\text{s.t.}\, 48-x_1+0.2x_2-x_3+0.2x_4-x_5+0.2x_6\leq 0,\\ &250-5x_1+x_2-5x_3+x_4-5x_5+x_6\leq 0.\end{aligned} \tag{36}$$

After relaxing constraints (36), the relaxed problem can be decomposed into six subproblems:

$$\begin{aligned}&\min_{\{x_i\}\in Z_+\cup\{0\},\ i=1,\ldots,6} L(x_1,x_2,x_3,x_4,x_5,x_6,\lambda_1,\lambda_2)\\ &=\min_{\{x_i\}\in Z_+\cup\{0\},\ i=1,\ldots,6}\{(x_1^2-\lambda_1x_1-5\lambda_2x_1)+(x_2^2+0.2\lambda_1x_2+\lambda_2x_2)+\\ &(x_3^2-\lambda_1x_3-5\lambda_2x_3)+(x_4^2+0.2\lambda_1x_4+\lambda_2x_4)+(x_5^2-\lambda_1x_5-5\lambda_2x_5)\\ &+(x_6^2+0.2\lambda_1x_6+\lambda_2x_6)+48\lambda_1+250\lambda_2\}.\end{aligned} \tag{37}$$

Since the problem is small, optimal multipliers $(\lambda_1, \lambda_2)^* = (9, 4.8)$[11] can be easily computed, and efficiency will be assessed by calculating distances from multipliers to $(\lambda_1, \lambda_2)^*$ at every iteration.

Following (12), stepsizes are initialised as:

$$c^0=\frac{1}{n}\frac{\hat{q}-q(\lambda^0)}{\|g(x^0)\|}, \tag{38}$$

where n is the number of subproblems ($n = 6$). An estimate of the optimal dual value $\hat{q}$ is chosen to be a feasible cost 867 of (35)–(36) corresponding to a feasible solution

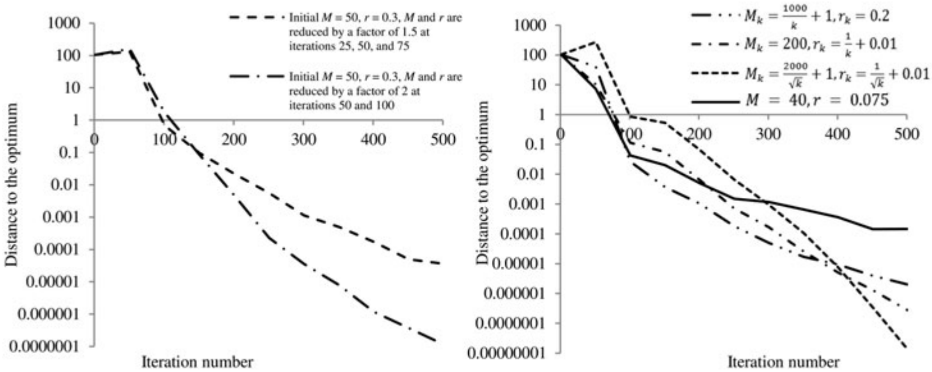

Figure 1. Results for $\{M_k\}$ and $\{r_k\}$ satisfying Proposition 3.6.

(17, 0, 17, 0, 17, 0). Multipliers will be updated using the formula (7), stepsizes will be updated using the formula (8), and stepsize-setting parameters will be updated following Proposition 3.6.

Results for $\{M_k\}$ and $\{r_k\}$ satisfying Proposition 3.6. To illustrate convergence under Proposition 3.6, few particular sequences $\{M_k\}$ and $\{r_k\}$ will be considered. For example, consider sequences with initial $M_0 = 50$, $r_0 = 0.3$. In the first sequence, M_k and r_k are reduced by a factor of 2 at iterations 50 and 100, and in the second sequence, M_k and r_k are reduced by a factor of 1.5 at iterations 25, 50 and 75. The results are shown in Figure 1.

In the sequences of $\{M_k\}$ and $\{r_k\}$ mentioned above, the initial parameters $M_0 = 50$ and $r_0 = 0.3$ are chosen to be large thereby allowing stepsizes to stay large in the beginning of the iterative process and making significant progress towards the optimum. As M_k and r_k decrease, stepsizes approach zero faster thereby leading to faster convergence. Other possible sequences $\{M_k\}$ and $\{r_k\}$ that satisfy Proposition 3.6 are (a) $M_k = \frac{1000}{k} + 1, r_k = 0.2$; (b) $M_k = 200, r_k = \frac{1}{k} + 0.01$; (c) $M_k = \frac{2000}{\sqrt{k}} + 1, r_k = \frac{1}{\sqrt{k}} + 0.01$

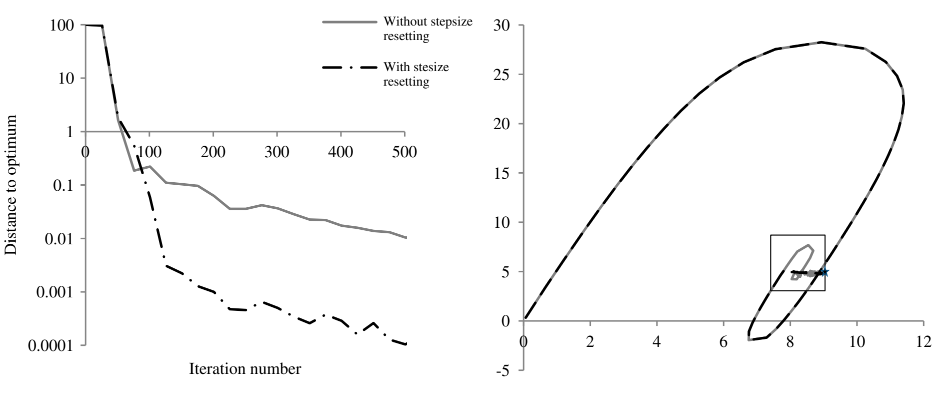

Figure 2. Convergence with and without stepsize re-initialisation.

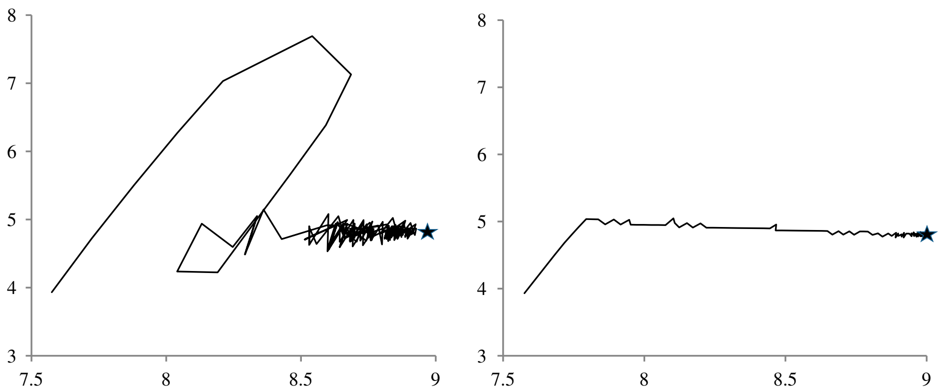

Figure 3. Trajectories of multipliers without re-initialisation (left) and trajectories of multipliers with re-initialisation (right). The optimum is shown by a star.

The results will be compared with the results obtained using constant parameters M and r that satisfy (11). For comparison, several constant parameters M and r were chosen for simulations, but for the sake of brevity, results obtained using only best combination of parameters M and r ($M = 40$, $r = 0.075$) will be compared with the results obtained using sequences of $\{M_k\}$ and $\{r_k\}$ shown before. The comparison results are demonstrated in Figure 1 (right). Convergence can also be sped up by re-initialising stepsizes during convergence as will be demonstrated ahead.

Results with Stepsize Re-initialisation. The stepsize is initialised according to (12) with $\hat{q} = 867$ as explained before. At iteration 30, a new feasible cost 846 is obtained and is used to re-initialise stepsizes according to (33). The results shown in Figure 2 (left) indicate that multipliers approach the optimum faster when re-initialised during the iterative process. To illustrate why convergence is improved when stepsizes are re-initialised (reduced) during the iterative process, Figure 2 (right) also shows trajectories of multipliers with and without of re-initialisations of stepsizes.

To demonstrate behaviour of multipliers near the optimum, the boxed-in portion of the Figure 2 (right) is zoomed in and is shown in Figure 3. As demonstrated in Figure 3 (left), without the re-initialisation, stepsizes remain large in the neighbourhood of the optimum (9, 4.8), which results in oscillations. In contrast, when stepsizes are re-initialised, multipliers approach optimum in a much smoother fashion as shown in Figure 3 (right).

Example 2. Unit Commitment and Economic Dispatch with Combined Cycle Units and Transmission Capacity Constraints. In this example, to demonstrate efficiency of the new method, the Unit Commitment and Economic Dispatch (UCED) problem with combined cycle units (Alemany et al., 2013; Anders, 2005) and transmission capacity constraints will be considered. The UCED problem seeks to minimise the total cost consisting of the total generation and the total start-up costs by determining which generators to commit and deciding their generation levels that satisfy generator capacity, ramp-rate and minimum up- and down-time constraints (Guan et al., 1992, 1994) and following transitions among states of combined cycle units while meeting the demand P_i^D at each node i and satisfying transmission capacity $f_{l,\max}$ in each transmission line l (Bragin et al., 2014, 2015b). The constraints are formulated as follows:

Generation Capacity Constraints: Status of each bid[12] m_i (= 1, … , M_i) at node i (= 1, … , I) indexed by (i,m_i) is modelled by binary decision variables $x_{(i,m_i)}(t)$: $x_{(i,m_i)}(t)$ = 1 indicates that the bid was selected, and $x_{(i,m_i)}(t)$ = 0 otherwise. If the bid is selected, energy $p_{(i,m_i)}(t)$ output should satisfy minimum/maximum generation levels:

$$x_{(i,m_i)}(t)p_{(i,m_i)\,\mathrm{min}} \le p_{(i,m_i)}(t) \le x_{(i,m_i)}(t)p_{(i,m_i)\,\mathrm{max}}. \tag{39}$$

The start-up cost $S_{(i,m_i)}(t)$ is incurred if and only if the unit i has been turned an 'on' from an 'off' state at hour t

$$S_{(i,m_i)}(t) \ge S_{(i,m_i)}\left(x_{(i,m_i)}(t) - x_{(i,m_i)}(t-1)\right). \tag{40}$$

Ramp-rate constraints ensure that the increase/decrease in the output of a unit does not exceed a pre-specified ramp-rate within one hour.

Minimum up- and down-time constraints ensure that a unit must be kept online/offline for a pre-specified number of hours. Formulation of ramp-rate and minimum up- and down-time constraints can be found in Guan et al., 1992 and Rajan & Takriti, 2005.

Transitions within Combined Cycle Units: Combined cycle units can operate at multiple configurations of combustion turbines (CTs) and steam turbines (STs). However, transitions among configurations may be constrained. For example, steam turbines cannot be turned on if there is not enough heat from combustion turbines. Transition rules (Alemany et al., 2013; Anders, 2005) for a configuration with two combustion turbines and one steam turbine (2CT + 1ST) and their linear formulation can be found in (Bragin et al., 2014, 2015b).

Demand Constraints: Committed generators need to satisfy energy nodal load levels $P_i^D(t)$ either locally or by transmitting power through transmission lines. The total power generated should be equal to the system demand:

$$\sum_{i=1}^{I}\sum_{m=1}^{M_i} p_{(i,m)}(t) = \sum_{i=1}^{I} P_i^D(t). \tag{41}$$

Power Flow Constraints: The power flow $f_{(b_1,b_2)}(t)$ in a line that connects nodes b_1 and b_2 can be expressed as a linear combination of net nodal injections of energy (Bragin et al., 2014):

$$f_{(b_1,b_2)}(t) = \sum_{i=1}^{I} a^i_{(b_1,b_2)} \cdot \left(\sum_{m=1}^{M_i} p_{(i,m)}(t) - P_i^D(t)\right). \tag{42}$$

Power flows in a line are essentially a linear combination of nodal injections with weights being a_l^i, referred to as 'shift factors'.

Transmission Capacity Constraints: Power flows in any line cannot exceed the transmission capacity $f_{l\max}$ which for simplicity is set to be the same for each direction

$$-f_{(b_1,b_2)\,\mathrm{max}} \le f_{(b_1,b_2)}(t) \le f_{(b_1,b_2)\,\mathrm{max}}. \tag{43}$$

Objective Function. The objective of the UCED problem with conventional and combined cycle unit is to minimise the cost consisting on the total bid and start-up costs:

$$\sum_{i=1}^{I}\sum_{m=1}^{M_i}\left(\sum_{t=1}^{T}c_{(i,m)}\left(p_{(i,m)}(t),t\right)+\sum_{t=1}^{T}S_{(i,m)}\right) \tag{44}$$

while satisfying all constraints mentioned before.

Testing IEEE 30-bus system (Zhang Song Hu & Yao, 2011). To test the new method, consider the IEEE 30-bus system that consists of 30 buses ($I = 30$) and 41 transmission lines ($L = 41$). The original data are modified so that each bus numbered 1 through 10 has exactly one combined cycle unit ($M_i = 1$), and each of the buses 11 and 12 has exactly one conventional generator.

To solve the problem, only nodal demand constraints (41) are relaxed and the relaxed problem becomes:

$$\begin{aligned}&\sum_{i=1}^{I}\sum_{m=1}^{M_i}\left(\sum_{t=1}^{T}c_{(i,m)}\left(p_{(i,m)}(t),t\right)+\sum_{t=1}^{T}S_{(i,m)}u_{(i,m)}(t)\right)\\&\quad+\sum_{t=1}^{T}\lambda(t)\left(\sum_{i=1}^{I}\sum_{m=1}^{M_i}p_{(i,m)}(t)-\sum_{i=1}^{I}P_i^D(t)\right),\end{aligned} \tag{45}$$

subject to all constraints mentioned before with the exception of nodal demand constraints (41).

A subproblem at iteration k can be written as:

$$\sum_{t=1}^{T}c_{(i,m_i)}\left(p_{(i,m_i)}(t),t\right)+\sum_{t=1}^{T}S_{(i,m_i)}u_{(i,m_i)}(t)+\sum_{t=1}^{T}\lambda^k(t)\left(p_{(i,m_i)}(t)\right), \tag{46}$$

subject to

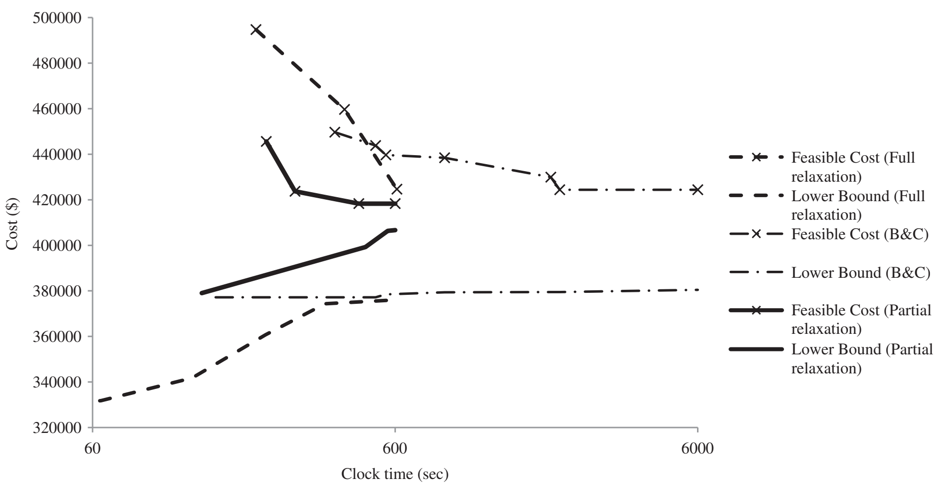

Figure 4. Comparison of the new method and branch-and-cut for the unit commitment problem.

$$-f_{(b_1,b_2)\max} \leq \sum_{\substack{j=1\\ j\neq i}}^{I} a^j_{(b_1,b_2)} \cdot \left(\sum_{m=1}^{M_j} p^{k-1}_{(j,m)}(t) - P^D_j(t)\right) + a^i_{(b_1,b_2)} \cdot \left(\sum_{m=1}^{M_i} p_{(i,m)}(t) - P^D_i(t)\right) \leq f_{(b_1,b_2)\max}, \tag{47}$$

and subject to generation capacity constraints, ramp-rate constraints, minimum up- and down-time constraints, and transitions among combined cycle states. Performance of the new method is compared to that of branch-and-cut, and the results are demonstrated in Figure 4.

Figure 4 shows that without full decomposition, the new method obtains a good feasible solution within 10 min of clock time. Upon comparison with the integration of surrogate Lagrangian relaxation and branch-and-cut with full relaxation, the new method converges faster judging by the quality of the lower bound, and the method obtains better feasible solutions. Performance of the method is also much better as compared to that of standard branch-and-cut.

As mentioned in subsection 3.1, when all system-wide coupling constraints are relaxed, subproblem convex hull invariance can be exploited. Without relaxing all system-wide constraints, subproblem convex hulls may no longer be invariant. Still, cuts generated by branch-and-cut that are based on subproblem constraints (with the exception of (47)) can be saved and reused in subsequent iterations to further reduce the computational effort. Without relaxing all system-wide constraints, however, reusing cuts may be difficult in practical implementations as discussed in subsection 3.1.

Example 3. Quadratic Assignment Problems. The problem was first formulated in 1957 by Koopmans and Beckmann (1957), and since then, the problem has been applied to many fields (Burkard & Offermann, 1977; Dickey & Hopkins, 1972; Elshafei, 1977; Geoffrion & Graves, 1976; Krarup & Pruzan, 1978; Miranda et al., 2005). The problem can be formulated as an integer nonlinear programming problem:

$$\min_{x_{i,j},x_{h,l}} \sum_{i,j=1}^{n} \sum_{h,l=1}^{n} d_{i,h} f_{j,l} x_{i,j} x_{h,l}, \; x_{i,j} \in \{0, 1\}, \; d_{i,h} \geq 0, \; f_{j,l} \geq 0, \tag{48}$$

$$\text{s.t.} \sum_{i=1}^{n} x_{i,j} = 1, j = 1, \ldots, n, \tag{49}$$

$$\sum_{j=1}^{n} x_{i,j} = 1, \; i = 1, \ldots, n. \tag{50}$$

In terms of the electronic board design problem (Miranda et al., 2005), the Quadratic Assignment problem can be explained as follows. Given n electronic components and locations, $d_{i,h}$ is the distance between location i and location h, $f_{j,l}$ are levels of interactivity and energy/data flow between component j and component l. Binary decision variables $x_{i,j}$ corresponds to component i being placed in location j iff $x_{i,j} = 1$. Assignment constraints (49) and (50) ensure that only one component can be assigned to a specific location. The problem falls in the category of integer monotonic programming problems since the objective function is monotonically increasing and constraints are linear.

To demonstrate the new method, consider a quadratic assignment problem instance with $n = 26$ (Burkard & Offermann, 1977). To solve the problem, the dynamic linearisation along the lines developed in subsection 3.1 will be used. Consider the following linearisation of the objective function:

$$\bar{f}\left(x, x^{k-1}\right) = \sum_{i,j=1}^{n} \sum_{h,l=1}^{n} d_{i,h} f_{j,l} \left(x_{i,j} x_{h,l}^{k-1} + x_{i,j}^{k-1} x_{h,l} - x_{i,j}^{k-1} x_{h,l}^{k-1} \right). \tag{51}$$

Constraints (49) can be viewed as system-wide coupling constraints, and constraints (50) can be viewed as subproblem constraints. The relaxed problem can be constructed by relaxing two constraints from (49), and the linearised relaxed problem becomes:

$$\min_{x_{i,j}, x_{h,l}} \left\{ \sum_{i,j=1}^{n} \sum_{h,l=1}^{n} d_{i,h} f_{j,l} \left(x_{i,j} x_{h,l}^{k-1} + x_{i,j}^{k-1} x_{h,l} - x_{i,j}^{k-1} x_{h,l}^{k-1} \right) + \sum_{j=1}^{2} \lambda_j \left(\sum_{i=1}^{n} x_{i,j} - 1 \right) \right\},$$
$$x_{i,j} \in \{0, 1\} \tag{52}$$

$$\text{s.t.} \quad \sum_{i=3}^{n} x_{i,j} = 1, j = 1, \ldots, n, \tag{53}$$

$$\sum_{j=1}^{n} x_{i,j} = 1, \; i = 1, \ldots, n, \tag{54}$$

and the linearised surrogate optimality condition is

$$\sum_{i,j=1}^{n} \sum_{h,l=1}^{n} d_{i,h} f_{j,l} \left(x_{i,j}^{k} x_{h,l}^{k-1} + x_{i,j}^{k-1} x_{h,l}^{k} - x_{i,j}^{k-1} x_{h,l}^{k-1} \right)$$
$$+ \sum_{j=1}^{2} \lambda_j \left(\sum_{i=1}^{n} x_{i,j}^{k} - 1 \right) < \sum_{i,j=1}^{n} \sum_{h,l=1}^{n} d_{i,h} f_{j,l} x_{i,j}^{k-1} x_{h,l}^{k-1} + \sum_{j=1}^{2} \lambda_j \left(\sum_{i=1}^{n} x_{i,j}^{k-1} - 1 \right). \tag{55}$$

A 'subproblem' m can be written as

$$\begin{gathered} \min_{x_{i,j}, x_{h,l}} \left\{ \sum_{i=m}^{m} \sum_{j=1}^{n} \sum_{h,l=1}^{n} d_{i,h} f_{j,l} x_{i,j} x_{h,l}^{k-1} + \sum_{i,j=1}^{n} \sum_{h=m}^{m} \sum_{l=1}^{n} d_{i,h} f_{j,l} x_{i,j}^{k-1} x_{h,l} \right. \\ \left. - \sum_{i,h=m}^{m} \sum_{j=1}^{n} \sum_{l=1}^{n} d_{i,h} f_{j,l} x_{i,j}^{k-1} x_{h,l}^{k-1} + \sum_{j=1}^{2} \lambda_j \sum_{m=1}^{m} x_{i,j} \right\} x_{i,j} \in \{0, 1\}, \end{gathered} \tag{56}$$

$$\text{s.t.} \sum_{i=3}^{n} x_{i,j} = 1, j = 1, \ldots, n, \tag{57}$$

$$\sum_{j=1}^{n} x_{i,j} = 1, \; i = m. \tag{58}$$

In this example, 10 linearised 'subproblems' are grouped together and the resulting objective function is

$$\min_{x_{i,j},x_{h,l}} \left\{ \sum_{i=m}^{m+9} \sum_{j=1}^{n} \sum_{h,l=1}^{n} d_{i,h} f_{j,l} x_{i,j} x_{h,l}^{k-1} + \sum_{i,j=1}^{n} \sum_{h=m}^{m+9} \sum_{l=1}^{n} d_{i,h} f_{j,l} x_{i,j}^{k-1} x_{h,l} \right. \\ \left. - \sum_{i,h=m}^{m+9} \sum_{j=1}^{n} \sum_{l=1}^{n} d_{i,h} f_{j,l} x_{i,j}^{k-1} x_{h,l}^{k-1} + \sum_{j=1}^{2} \lambda_j \sum_{m=1}^{m+9} x_{i,j} \right\} x_{i,j} \in \{0, 1\}. \tag{59}$$

Even for $n = 26$, the number of terms in the objective function (59) is fairly large, and this objective function may not be handled efficiently in practical implementations. To deal with this issue, the number of terms is reduced by removing terms that involve decision variable values that are zero. Moreover, regrouping terms in a way that the number of outer summations is small will also reduce the computational effort. Lastly, the third term in (59) does not contain decision variables, and it can be removed as it will not affect the solution process. The resulting linearised 'subproblems' can be written as:

$$\min_{x_{i,j}} \left\{ \sum_{i=m}^{m+9} \sum_{\substack{j,h,l=1 \\ x_{h,l}^{k-1} \neq 0}}^{n} \left(d_{i,h} f_{j,l} + d_{h,i} f_{l,j}\right) x_{i,j} x_{h,l}^{k-1} + \sum_{j=1}^{2} \lambda_j \sum_{m=1}^{m+9} x_{i,j} \right\}, x_{i,j} \in \{0, 1\}, \tag{60}$$

$$\text{s.t.} \quad \sum_{i=3}^{n} x_{i,j} = 1, j = 1, \ldots, n, \tag{61}$$

$$\sum_{j=1}^{n} x_{i,j} = 1, \ i = m, \ldots, m+9. \tag{62}$$

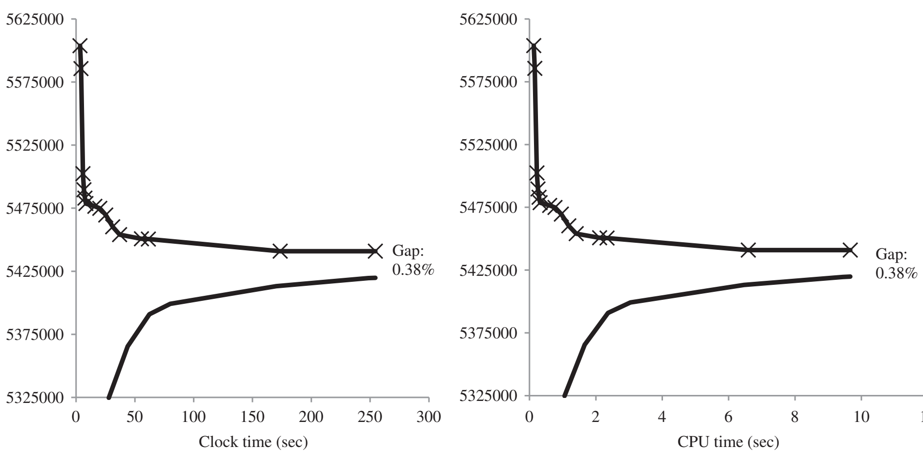

Figure 5. Solution profile for the Quadratic Assignment Problem instance Bur26a (Burkard & Offermann, 1977). Feasible costs are marked by the cross ×.

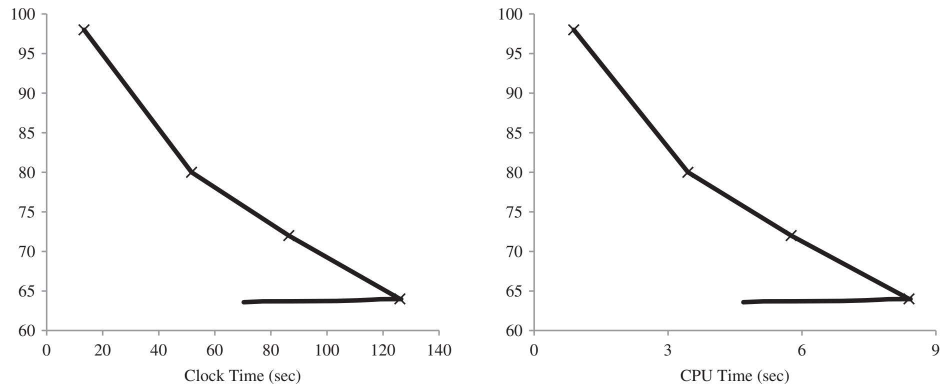

Figure 6. Solution profile for the Quadratic Assignment Problem instance Esc128 (Eschermann & Wunderlich, 1990). Feasible costs are marked by the cross ×.

The parameter m is chosen sequentially starting from $m = 1$ and increasing in increments of 2 until the value $m = 17$ is reached, after which the count starts from $m = 1$ again.

Each subproblem (60)–(62) is now easy to solve. Moreover, according to the numerical experience, multipliers that correspond to the relaxed constraints were positive throughout the entire iterative process and the monotonicity of subproblem objective functions was preserved.

Figure 5 shows that the new method obtains a good feasible solution within a few minutes. Moreover, if only CPU time spent solving subproblems is counted, the new method obtains a feasible solution with 0.38% gap within under 10 seconds thereby indicating that the dynamic linearisation is efficient. To test the scalability of the new method, consider a problem instance with $n = 128$. Figure 6 demonstrates that the new method is scalable and capable of efficiently solving Quadratic Assignment problems of large sizes ($n = 128$).

5. Conclusion

In this study, building upon the recently developed integration of surrogate Lagrangian relaxation and branch-and-cut, a new method is developed to solve difficult and nonlinear MIP problems under the condition of monotonicity of the objective function and linearity of constraints. The method exploits problem structure after selective relaxation of system-wide constraints, monotonicity of resulting subproblems through a dynamic linearisation while efficiently coordinating subproblem solutions and guaranteeing convergence. When all system-wide constraints are relaxed, subproblem convex hull invariance of linearised subproblems can be exploited to improve efficiency of the method. Through novel guidelines for selecting stepsize-updating parameters, fast convergence is achieved. The idea of the new method is generic and can be applied for solving integer and MIP problems under an assumption of monotonicity of objective functions and linearity of constraints. The method opens up new directions for solving other difficult nonlinear integer and MIP problems.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This work was supported by the United States National Science Foundation [grant numbers ECCS-1028870 and ECCS-1509666] and Southern California Edison.

Notes

1. The analysis of integer programming problems is similar to the analysis of mixed-integer programming problems.
2. Monotonic objective functions can be increasing/decreasing linear functions or increasing/decreasing nonlinear functions.
3. As compared to conventional generators, combined cycle units are more efficient because heat from combustion turbines is not wasted into the atmosphere but is used to generate steam in steam turbines to generate more electricity.
4. For example, steam turbines cannot be turned on if there is not enough heat from combustion turbines.
5. One of the recent applications of the problem is the electronic board design problem (Miranda, Luna, Mateus, & Ferreira, 2005). In a circuit board, a number of electronic components need to be placed to a number of locations. To avoid signal delays, the distance among components with greater levels of interactivity and energy/data flow is minimised.
6. Quadratic assignment problems are considered to be one of the most difficult problems, and instances with the size larger than 30 cannot be solved 'in reasonable CPU time' (Hahn, Zhu, Guignard, Hightower, & Saltzman, 2012).
7. The selective relaxation of system-wide coupling constraints also alleviates the third difficulty associated with relaxation of many system-wide coupling constraints (15). Typically, the choice of constraints to be relaxed depends on the nature of the problem.
8. When solving subproblem i, all other decision variables $x_l, l \neq i$ in constraints (25b) are fixed at x_l^{k-1}.
9. Same argument holds if several linearised subproblems (24)–(25) are solved at a time.
10. Feasible cost can be obtained by adjusting subproblem solutions to satisfy violated constraints as mentioned in Section 2.
11. These values are obtained using surrogate Lagrangian relaxation and running the algorithm for sufficiently many iterations.
12. Each bid corresponds to either a conventional unit, or to a combustion/steam turbine generator that comprises a combined cycle unit.

References

Alemany, J., Moitre, D., Pinto, H., & Magnago, F. (2013). Short-term scheduling of combined cycle units using mixed integer linear programming solution. *Energy and Power Engineering, 5*, 161–170.

Al-Khayyal, F. A. (1990). Jointly constrained bilinear programs and related problems: An overview. *Computers & Mathematics with Applications, 19*, 53–62.

Anders, G. (Principal Investigator). (2005). *Commitment techniques for combined cycle generating units* (CEATI Report No. T053700-31-3). Toronto: Kinectrics.

Balas, E., Ceria, S., & Cornuéjols, G. (1996). Mixed 0-1 programming by lift-and-project in a branch-and-cut framework. *Management Science, 42*, 1229–1246.

Bragin, M. A., Luh, P. B., Yan, J. H., & Stern, G. A. (2014). Surrogate Lagrangian relaxation and branch-and-cut for unit commitment with combined cycle units. *Proceedings of the IEEE Power and Energy Society*. General Meeting, National Harbor, MA.

Bragin, M. A., Luh, P. B., Yan, J. H., Yu, N., & Stern, G. A. (2015a). Convergence of the surrogate Lagrangian relaxation method. *Journal of Optimization Theory and Applications, 164*, 173–201.

Bragin, M. A., Luh, P. B., Yan, J. H., & Stern, G. A. (2015b). Novel exploitation of convex hull invariance for solving unit commitment by using surrogate Lagrangian relaxation and branch-and-cut. *Proceedings of the IEEE Power and Energy Society*. General Meeting, Denver, CO.

Bragin, M. A., Luh, P. B., Yan, J. H., & Stern, G. A. (2015c). Surrogate Lagrangian relaxation and branch-and-cut for mixed-integer linear programming problems. Submitted.

Burkard, R. E., & Offermann, J. (1977). Entwurf von Schreibmaschinentastaturen mittels quadratischer Zuordnungsprobleme [Design of typewriter keyboards using quadratic assignment problems]. *Zeitschrift fur Operations Research, 21*, 121–132. (in German).

Dickey, J. W., & Hopkins, J. W. (1972). Campus building arrangement using topaz. *Transportation Research, 6*, 59–68.

Elshafei, A. N. (1977). Hospital layout as a quadratic assignment problem. *Journal of the Operational Research Society, 28*, 167–179.

Ermoliev, Y. M. (1966). Methods for solving nonlinear extremal problems. *Cybernetics, 2*, 1–17.

Eschermann, B., & Wunderlich, H. J. (1990). Optimized synthesis of self-testable finite state machines. *Proceedings of the 20th International Symposium on Fault-Tolerant Computing (FFTCS 20)*. Newcastle upon Tyne

Fisher, M. L. (1973). Optimal solution of scheduling problems using Lagrange multipliers: Part I. *Operations Research, 21*, 1114–1127.

Fisher, M. L. (1981). The Lagrangian relaxation method for solving integer programming problems. *Management Science, 27*, 1–18.

Geoffrion, A. M. (1974). Lagrangian relaxation for integer programming. *Mathematical Programming Studies, 2*, 82–114.

Geoffrion, A. M., & Graves, G. W. (1976). Scheduling parallel production lines with changeover costs: Practical applications of a quadratic assignment/LP approach. *Operations Research, 24*, 595–610.

Goffin, J.-L., & Kiwiel, K. (1999). Convergence of a simple subgradient level method. *Mathematical Programming, 85*, 207–211.

Gorski, J., Pfeuffer, F., & Klamroth, K. (2007). Biconvex sets and optimization with biconvex functions: A survey and extensions. *Mathematical Methods of Operations Research, 66*, 373–407.

Guan, X., Luh, P. B., Yan, H., & Amalfi, J. A. (1992). An optimization-based method for unit commitment. *International Journal of Electrical Power and Energy Systems, 14*, 9–17.

Guan, X., Luh, P. B., Yan, H., & Rogan, P. M. (1994). Optimization-based scheduling of hydrothermal power systems with pumped-storage units. *IEEE Transactions on Power Systems, 9*, 1023–1031.

Hahn, P. M., Zhu, Y. R., Guignard, M., Hightower, W. L., & Saltzman, M. J. (2012). A level-3 reformulation-linearization technique-based bound for the quadratic assignment problem. *INFORMS Journal on Computing, 24*, 202–209.

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

Koopmans, T. C., & Beckmann, M. J. (1957). Assignment problems and the location of economic activities. *Econometrica, 25*, 53–76.

Krarup, J., & Pruzan, P. M. (1978). Computer-aided layout design. *Mathematical Programming Study, 9*, 75–94.

Li, G., Wen, C., Zheng, W. X., & Zhao, G. (2015). Iterative identification of block-oriented nonlinear systems based on biconvex optimization. *Systems and Control Letters, 79*, 68–75.

Li, G., Wen, C., & Zhang, A. (2015). Fixed point iteration in identifying bilinear models. *Systems and Control Letters, 83*, 28–37.

Miranda, G., Luna, H. P. L., Mateus, G. R., & Ferreira, R. P. M. (2005). A performance guarantee heuristic for electronic components placement problems including thermal effects. *Computers and Operations Research, 32*, 2937–2957.

Padberg, M., & Rinaldi, G. (1991). A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems. *SIAM Review, 33*, 60–100.

Padberg, M. (2005). Classical cuts for mixed-integer programming and branch-and-cut. *Annals of Operations Research, 139*, 321–352.

Polyak, B. T. (1967). A general method of solving extremum problems. *Soviet Mathematics Doklady, 8*, 593–597.

Rajan, D., & Takriti, S. (2005). *Minimum up/down polytopes of the unit commitment problem with start-up costs* (Technical Report RC23628). Yorktown Heights, NY: IBM T. J. Watson Res. Center.

Shor, N. Z. (1968). On the rate of convergence of the generalized gradient method. *Cybernetics, 4*, 79–80.

Shor, N. Z. (1976). Generalized gradient methods for non-smooth functions and their applications to mathematical programming problems. *Economic and Mathematical Methods, 12*, 337–356. (in Russian).

Taillard, E. (1991). Robust taboo search for the quadratic assignment problem. *Parallel Computing, 17*, 443–455.

Tate, D. M., & Smith, A. E. (1995). A genetic approach to the quadratic assignment problem. *Computers & Operations Research, 22*, 73–83.

Xia, Y., & Yuan, Y. X. (2006). A new linearization method for quadratic assignment problems. *Optimization Methods and Software, 21*, 805–818.

Zhang, S., Song, Y., Hu, Z., & Yao, L. (2011). Robust optimization method based on scenario analysis for unit commitment considering wind uncertainties. *Proceedings of the IEEE Power and Energy Society.* General Meeting, Detroit, MI.

Zhao, X., Luh, P. B., & Wang, J. (1999). Surrogate gradient algorithm for Lagrangian relaxation. *Journal of Optimization Theory and Applications, 100*, 699–712.

Notes on contributors

Mikhail A. Bragin received B.S. and M.S. degree in mathematics from the Voronezh State University, Russia, in 2004, the M.S. degree in physics and astronomy from the University of Nebraska-Lincoln, USA, in 2006, and the M.S. degree in Electrical and Computer Engineering from the University of Connecticut, USA, in 2014. He is currently pursuing the PhD degree in electrical and computer engineering at the University of Connecticut, Storrs. His research interests include mathematical optimisation methods, operations, and economics of electricity markets.

Peter B. Luh received his BS from National Taiwan University, MS from MIT and PhD from Harvard University. He has been with the Department of Electrical and Computer Engineering at the University of Connecticut, Storrs, CT, US, since 1980, and is the SNET Professor of Communications & Information Technologies. He is also a member of the Chair Professors Group at the Center for Intelligent and Networked Systems (CFINS) in the Department of Automation at Tsinghua University, Beijing, China. He was on sabbatical leave at the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, as a Chair Professor from July to December 2015. He is a fellow of IEEE and a member of the Periodicals Committee of IEEE Technical Activities Board. He was the founding Editor-in-Chief of the IEEE Transactions on Automation Science and Engineering, and an EiC of IEEE Transactions on Robotics and Automation. His research interests include smart, green and safe buildings; smart grid, electricity markets and effective renewable integration to the grid; and intelligent manufacturing systems. He is the recipient of the 2013 Pioneer Award of the IEEE Robotics and Automation Society for his pioneering contributions to the development of near-optimal and efficient planning, scheduling and coordination methodologies for manufacturing and power systems.

Joseph H. Yan is the Principle Manager of Portfolio Development and Valuation at Southern California Edison (SCE). For the past decades, he has led the strategy development of SCE's generation portfolios to optimise the value of these resources and reduce the cost of serving its customers. He has also actively engaged in California electricity market design representing SCE and made extraordinary contributions to the development of the ISO markets. His research interest includes operation research, optimisation, unit commitment/scheduling and transaction evaluation, and optimal simultaneous auction in deregulated ISO/RTO markets. Joseph Yan holds a PhD in Electrical and Systems Engineering of the University of Connecticut.

Gary A. Stern is the Director of Energy Policy for Southern California Edison Company (SCE). Reporting to the Vice President of Energy & Environmental Policy, he manages a division responsible for the development of policy in matters relating to the California Public Utilities Commission, and the California Energy Commission. He is responsible for strategic projects primarily relating to emerging technologies and climate change. His organisation is also responsible for case management associated with proceedings with these entities. Previously, Gary directed SCE's resource planning, market design and analysis, and strategic project groups. Gary Stern holds a PhD in Economics from the University of California at San Diego. He has an M.A. in Economics, and a B.A. in mathematics also from UC San Diego.