UNIVERSITY OF CAMBRIDGE
Faculty of Economics

Cambridge Working Papers in Economics

Cambridge Working Papers in Economics: 1870

INCREASE-DECREASE GAME UNDER IMPERFECT COMPETITION IN TWO-STAGE ZONAL POWER MARKETS PART II: SOLUTION ALGORITHM

M. Sarfati M. R. Hesamzadeh P. Holmberg

28 November 2018

In part I of this paper, we proposed a Mixed-Integer Linear Program (MILP) to analyse imperfect competition of oligopoly producers in two-stage zonal power markets. In part II of this paper, we propose a solution algorithm which decomposes the proposed MILP model into several subproblems and solve them in parallel and iteratively. Our solution algorithm reduces the solution time of the MILP model and it allows us to analyze largescale examples. To tackle the multiple Subgame Perfect Nash Equilibria (SPNE) situation, we propose a SPNE-band approach. The SPNE band is split into several subintervals and the proposed solution algorithm finds a representative SPNE in each subinterval. Each subinterval is independent from each other, so this structure enables us to use parallel computing. We also design a pre-feasibility test to identify the subintervals without SPNE. Our proposed solution algorithm and our SPNE-band approach are demonstrated on the 6-node and the modified IEEE 30-node example systems. The computational tractability of our solution algorithm is illustrated for the IEEE 118-node and 300-node systems.

Increase-Decrease Game under Imperfect Competition in Two-stage Zonal Power Markets - Part II: Solution Algorithm

EPRG Working Paper 1838

Cambridge Working Paper in Economics 1870

M. Sarfati, M.R. Hesamzadeh, P. Holmberg

Abstract In part I of this paper, we proposed a Mixed-Integer Linear Program (MILP) to analyze imperfect competition of oligopoly producers in two-stage zonal power markets. In part II of this paper, we propose a solution algorithm which decomposes the proposed MILP model into several subproblems and solve them in parallel and iteratively. Our solution algorithm reduces the solution time of the MILP model and it allows us to analyze largescale examples. To tackle the multiple Subgame Perfect Nash Equilibria (SPNE) situation, we propose a SPNE-band approach. The SPNE band is split into several subintervals and the proposed solution algorithm finds a representative SPNE in each subinterval. Each subinterval is independent from each other, so this structure enables us to use parallel computing. We also design a pre-feasibility test to identify the subintervals without SPNE. Our proposed solution algorithm and our SPNE-band approach are demonstrated on the 6-node and the modified IEEE 30-node example systems. The computational tractability of our solution algorithm is illustrated for the IEEE 118-node and 300-node systems.

Keywords Modified Benders decomposition, Multiple Subgame Perfect Nash equilibria, Parallel computing, Wholesale electricity market, Zonal pricing

JEL Classification C61, C63, C72, D43, L13, L94

Contact par.holmberg@ifn.se
Publication November 2018
Financial Support

Increase-Decrease Game under Imperfect Competition in Two-stage Zonal Power Markets – Part II: Solution Algorithm☆
November 27, 2018

M. Sarfati[a,b], M. R. Hesamzadeh[a], P. Holmberg[b,c,d]

[a] *Electricity Market Research Group (EMReG), KTH Royal Institute of Technology, Sweden*
[b] *Research Institute of Industrial Economics (IFN), Sweden*
[c] *Energy Policy Research Group (EPRG), University of Cambridge, UK*
[d] *Program on Energy and Sustainable Development (PESD), Stanford University, CA, USA*

Abstract

In part I of this paper, we proposed a Mixed-Integer Linear Program (MILP) to analyze imperfect competition of oligopoly producers in two-stage zonal power markets. In part II of this paper, we propose a solution algorithm which decomposes the proposed MILP model into several subproblems and solve them in parallel and iteratively. Our solution algorithm reduces the solution time of the MILP model and it allows us to analyze large-scale examples. To tackle the multiple Subgame Perfect Nash Equilibria (SPNE) situation, we propose a SPNE-band approach. The SPNE band is split into several subintervals and the proposed solution algorithm finds a representative SPNE in each subinterval. Each subinterval is independent from each other, so this structure enables us to use parallel computing. We also design a pre-feasibility test to identify the subintervals without SPNE. Our proposed solution algorithm and our SPNE-band approach are demonstrated on the 6-node and the modified IEEE 30-node example systems. The computational tractability of our solution algorithm is illustrated for the IEEE 118-node and 300-node systems.

Keywords: Modified Benders decomposition, Multiple Subgame Perfect Nash equilibria, Parallel computing, Wholesale electricity market, Zonal pricing
JEL Classification: C61, C63, C72, D43, L13, L94

1. Introduction

This paper is a continuation of [1]. In [1], we formulated a Two-stage Stochastic Equilibrium Problem with Equilibrium Constraints (TS-EPEC) to model producers' bidding game in two-stage zonal power markets. In the literature, two approaches are commonly used to solve TS-EPECs. In the first approach, each producer's profit maximization problem is formulated as a Mathematical Problem with Equilibrium Constraints (MPEC) and then it is reformulated as a Nonlinear Program (NLP). The Subgame Perfect Nash Equilibrium (SPNE) of the game is found by solving the Karush-Kuhn-Tucker (KKT) conditions of

Email addresses: `sarfati@kth.se` (M. Sarfati), `mrhesamzadeh@ee.kth.se` (M. R. Hesamzadeh), `par.holmberg@ifn.se` (P. Holmberg)

all producers' NLPs simultaneously [2, 3]. The second approach uses the diagonalization method [4] which solves each producer's MPEC model sequentially until the SPNE is found [5, 6]. The mathematical models in both approaches are nonconvex and therefore there is no guarantee to find the global solution. In the first part of this paper [1], we use a discrete approximation that allows us to reformulate the TS-EPEC as a MILP model. Our MILP model in [1] can be solved using the commercial MILP solvers. However, solving large-scale MILP models is not an easy task (it is NP-hard). The commercial solvers such as CPLEX and XPRESS use the branch-and-cut algorithm to solve MILP models. The branch-and-cut algorithm relaxes the integrality constraints on the integer variables and formulates a relaxed Linear Program (LP). It partitions the feasibility region into smaller subsets. The relaxed LP is solved in each subset iteratively until the global solution is reached. This solution algorithm often becomes intractable for MILP models with many binary variables. An alternative way to solve big MILP models, which we use, is to decompose it into several smaller optimization problems and then solve smaller ones iteratively and hopefully in parallel. Benders decomposition [7] is used in the literature to improve the tractability of MILP models [8, 9]. Benders algorithm decomposes the MILP model into a master problem and subproblem. Then it solves the master problem and the subproblem iteratively. In this manner, a series of smaller problems are solved instead of a single large one.

Another issue is that, game-theoretic models of power markets may have multiple equilibria. Several researchers propose different methods to tackle multiple Nash equilibria in one-stage power markets [10–19]. A common approach is to find all Nash equilibria as in [10–12]. To do this, after finding a new Nash equilibrium, a constraint is added to the model to exclude the already found Nash equilibrium from the feasibility set. For large-scale examples, finding all Nash equilibria may not be an easy task. Authors in [13–17] analyze the Nash equilibria with the highest dispatch cost and the one with the lowest dispatch cost. A Nash-equilibria band which consists of all Nash equilibria is constructed and used for merger analysis in [18]. Authors in [19] apply the same Nash-equilibria-band approach to a ramp-game between producers.

The contributions of the current paper are as follows: (a) we propose a solution algorithm which decomposes the proposed MILP model in part I of this paper [1] into several subproblems and solves them iteratively. Our proposed solution algorithm finds the optimal solution in shorter time for the 6-node and the modified IEEE 30-node systems as compared to the Benders decomposition embedded in CPLEX. Moreover it finds the optimal solution in larger systems where the Benders decomposition embedded in CPLEX fails to find any solution. (b) In the proposed MILP model in part I of this paper [1], each producer chooses their price bids from a discrete price grid. The price grid might be rather coarse. In this paper, we propose an iterative grid-refinement technique which allows us to improve the accuracy of the price grid and the computed equilibria. (c) We also contribute to the existing literature by tackling multiple SPNE in large-scale examples. SPNE are computed by means of the SPNE-band concept. The band is created by placing a lower bound where the dispatch cost is lower than the one in the best SPNE (with the lowest dispatch cost) and an upper bound where the dispatch cost is higher than the one in the worst SPNE (with the highest dispatch cost). The SPNE band is divided into several subintervals and a representative SPNE is found in each subinterval. This method enables us to parallelize our proposed solution algorithm over subintervals and solve for a SPNE in each subinterval concurrently. (d) We design a pre-feasibility test to identify the subintervals which has no SPNE, so that we can avoid searching for a SPNE in those subintervals. The proposed solution algorithm and the proposed SPNE-band approach are demonstrated on the 6-node and the modified IEEE 30-node systems. The IEEE 118-node and 300-node systems are implemented to show the computational efficiency of our proposed solution algorithm.

The rest of this paper is organized as follows. In addition to the symbols in the first part of this paper [1], the additional symbols used in the proposed solution algorithm are presented in Section 2. Sections 3 and 4 explain our proposed solution algorithm and the SPNE-band approach, respectively. The application of the proposed solution algorithm and the SPNE-band approach are demonstrated on the 6-node, modified IEEE 30-node, 118-node and 300-node systems in section 5. Section 6 concludes the paper.

2. Nomenclature

In addition to the symbols presented in part I of this paper, the following symbols are introduced.

Indices

q	iteration index

Parameters

$\tau^{(q)}$	Tolerance of price grid in iteration q,
$\bar{\tau}$	Predefined tolerance of price grid,
ϵ	Tolerance for SPNE-band approach,
ε	Tolerance for modified Benders decomposition,
$\overline{c}_u$,($\underline{c}_u$)	upper (lower) limit of day-ahead price-bid of producer u
$\overline{c}_u^{up}$,($\underline{c}_u^{up}$)	upper (lower) limit of up-regulation price-bid of producer u
$\overline{c}_u^{dn}$,($\underline{c}_u^{dn}$)	upper (lower) limit of down-regulation price-bid of producer u

3. Proposed solution algorithm

The SPNE in two-stage zonal power markets is formulated as a stochastic MILP model in part I of this paper. It is set out in (1).

$$\underset{\Phi}{\text{Minimize}} \quad \sum_u \zeta_u \tag{1a}$$

Subject to:

$$\sum_u (g_u + g_{u,s}^{up} - g_{u,s}^{dn}) = \sum_n (v_{n,s} + D_n - \Delta W_{n,s}) : (\delta_s^{(1)}),\ \forall s \tag{1b}$$

$$F_k - \sum_n H_{k,n} (\sum_{n:u} (g_u + g_{u,s}^{up} - g_{u,s}^{dn}) - v_{n,s} - D_n + \Delta W_{n,s}) \geq 0 : (\delta_{k,s}^{(2)}),\ \forall k, s \tag{1c}$$

$$0 \leq g_{u,s}^{up} \leq G_u - g_u : (\delta_{u,s}^{(3)}, \delta_{u,s}^{(4)}),\ \forall u, s \tag{1d}$$

$$0 \leq g_{u,s}^{dn} \leq g_u : (\delta_{u,s}^{(5)}, \delta_{u,s}^{(6)}),\ \forall u, s \tag{1e}$$

$$0 \leq v_{n,s} \leq \bar{W}_n + \Delta W_{n,s} : (\delta_{n,s}^{(7)}, \delta_{n,s}^{(8)}),\ \forall n, s \tag{1f}$$

$$-\sigma_s \hat{c}_u^{up} + \alpha_s - \sum_{n:u} \sum_k H_{k,n} \mu_{k,s} + \kappa_{u,s} - \beta_{u,s} = 0 : (\delta_{u,s}^{(9)}),\ \forall u, s \tag{1g}$$

$$\sigma_s \hat{c}_u^{dn} - \alpha_s + \sum_{n:u} \sum_k H_{k,n} \mu_{k,s} + \psi_{u,s} - \varphi_{u,s} = 0 : (\delta_{u,s}^{(10)}),\ \forall u, s \tag{1h}$$

$$\alpha_s - \sum_k H_{k,n} \mu_{k,s} - \theta_{n,s} + \chi_{n,s} = 0 : (\delta_{n,s}^{(11)}),\ \forall n, s \tag{1i}$$

$$\rho'_{z,s} = (\alpha_s - \sum_k H_{k,n} \mu_{k,s})/\sigma_s : (\delta_{n,s}^{(19)}),\ \forall n \in z,\ \forall s \tag{1j}$$

$$(\overline{R}_{z,s} \beta_{u,s} (G_u - g_u) + \underline{R}_{z,s} \varphi_{u,s} g_u)/\sigma_s + C_u^{dn} g_{u,s}^{dn} - C_u^{up} g_{u,s}^{up} + \hat{c}_u^{up} g_{u,s}^{up} - \hat{c}_u^{dn} g_{u,s}^{dn} \geq 0 : (\delta_{u,s}^{(20)}),\ \forall u, s \tag{1k}$$

$$-\sigma_s \hat{c}_u^{up} + \lambda_s^A - \sum_{n:u} \sum_k H_{k,n} \lambda_{k,s}^B + \lambda_{u,s}^C - \lambda_{u,s}^D + \lambda_{u,s}^M \hat{c}_u^{up} - \lambda_{u,s}^M C_u^{up} = 0,\ \forall u, s \tag{1l}$$

$$\sigma_s \hat{c}_u^{dn} - \lambda_s^A + \sum_{n:u} \sum_k H_{k,n} \lambda_{k,s}^B + \lambda_{u,s}^E - \lambda_{u,s}^F + \lambda_{u,s}^M C_u^{dn} - \lambda_{u,s}^M \hat{c}_u^{dn} = 0,\ \forall u, s \tag{1m}$$

$$\lambda_s^A - \sum_k H_{k,n} \lambda_{k,s}^B - \lambda_{n,s}^G + \lambda_{n,s}^H = 0,\ \forall n, s \tag{1n}$$

$$\sum_n (\lambda_{n,s}^L/\sigma_s - \Delta W_{n,s} - \lambda_{n,s}^K) + \sum_u (\lambda_{u,s}^I - \lambda_{u,s}^J) = 0,\ \forall s \tag{1o}$$

$$-F_k + \sum_n H_{k,n} (\Delta W_{n,s} - D_n + \lambda_{n,s}^K + \lambda_{n,s}^L/\sigma_s + \sum_{u:n} (g_u + \lambda_{u,s}^J - \lambda_{u,s}^I)) + \lambda_{k,s}^N = 0,\ \forall k, s \tag{1p}$$

$$\lambda_{u,s}^I + \lambda_{u,s}^O = 0,\ \forall u, s \tag{1q}$$

$$g_u - G_u - \lambda_{u,s}^I + \sum_{z:u} \overline{R}_{z,s} (G_u - g_u) \lambda_{u,s}^M/\sigma_s + \lambda_{u,s}^P = 0,\ \forall u, s \tag{1r}$$

$$\lambda_{u,s}^J + \lambda_{u,s}^Q = 0,\ \forall u, s \tag{1s}$$

$$g_u + \lambda_{u,s}^J - \sum_{z:u} \underline{R}_{z,s} g_u \lambda_{u,s}^M/\sigma_s - \lambda_{u,s}^R = 0,\ \forall u, s \tag{1t}$$

$$\lambda_{n,s}^K + \lambda_{n,s}^S = 0,\ \forall n, s \tag{1u}$$

$$\lambda_{n,s}^T - \lambda_{n,s}^K - (W_n + \Delta W_{n,s}) = 0,\ \forall n, s \tag{1v}$$

$$\sum_{n \in z} \lambda_{n,s}^L = 0 : (\delta_{z,s}^{(32)}),\ \forall z, s \tag{1w}$$

$$\begin{aligned}
&\sigma_s \sum_u (-\hat{c}_u^{up} g_{u,s}^{up} + \hat{c}_u^{dn} g_{u,s}^{dn}) - (\alpha_s \sum_n \Delta W_{n,s} + \sum_k \mu_{k,s} (F_k - \sum_n H_{k,n} (\sum_{n:u} g_u - D_n + \\
&\quad \Delta W_{n,s})) + \sum_u (\beta_{u,s} (G_u - g_u) + \varphi_{u,s} g_u) + \sum_n \chi_{n,s} (W_n + \Delta W_{n,s})) - (\lambda_s^A \sum_n \Delta W_{n,s} + \\
&\quad \sum_k \lambda_{k,s}^B (F - \sum_n (H_{k,n} (\sum_{n:u} g_u + \Delta W_{n,s} - D_n))) + \sum_u (\lambda_{u,s}^D (G_u - g_u) + \lambda_{u,s}^F g_u - \\
&\quad \lambda_{u,s}^I \hat{c}_u^{up} \sigma_s + \lambda_{u,s}^J \hat{c}_u^{dn} \sigma_s) + \sum_n \lambda_{n,s}^H (W_n + \Delta W_{n,s})) = 0 : (\delta_s^{(48)}),\ \forall s
\end{aligned} \tag{1x}$$

$$\sum_u g_u = \sum_n D_n : (\delta^{(49)}) \tag{1y}$$

$$\bar{F}_l - \sum_z H'_{l,z} (\sum_{u:z} g_u - \sum_{n:z} D_n) \geq 0 : (\delta_l^{(50)}),\ \forall l \tag{1z}$$

$$0 \leq g_u \leq G_u : (\delta_u^{(51)}, \delta_u^{(52)}) \ \forall u \tag{1aa}$$

$$\hat{c}_u - \xi + \sum_{z:u} \sum_l H'_{l,z} \gamma_l - \eta_u + \nu_u = 0 : (\delta_u^{(53)}), \ \forall u \tag{1ab}$$

$$-\sum_u \hat{c}_u g_u - (\xi \sum_n -D_n + \sum_l \gamma_l (\bar{F}_l + \sum_z H'_{l,z} (\sum_{n:z} D_n)) + \sum_u \nu_u G_u) = 0 : (\delta^{(57)}) \tag{1ac}$$

$$\begin{aligned}&\mu_{k,s}, \kappa_{u,s}, \beta_{u,s}, \psi_{u,s}, \varphi_{u,s}, \theta_{n,s}, \chi_{n,s}, \lambda^B_{k,s}, \lambda^C_{u,s}, \lambda^D_{u,s}, \lambda^E_{u,s}, \lambda^F_{u,s}, \lambda^G_{n,s}, \lambda^H_{n,s}, \lambda^M_{u,s}, \lambda^N_{k,s}, \lambda^O_{u,s}, \lambda^P_{u,s},\\ &\quad \lambda^Q_{u,s}, \lambda^R_{u,s}, \lambda^S_{n,s}, \lambda^T_{n,s}, \gamma_l, \eta_u, \nu_u \geq 0 : (\delta^{(12)}_{k,s}, \delta^{(13)}_{u,s}, \delta^{(14)}_{u,s}, \delta^{(15)}_{u,s}, \delta^{(16)}_{u,s}, \delta^{(17)}_{n,s}, \delta^{(18)}_{n,s}, \delta^{(33)}_{k,s}, \delta^{(34)}_{u,s},\\ &\quad \delta^{(35)}_{u,s}, \delta^{(36)}_{u,s}, \delta^{(37)}_{u,s}, \delta^{(38)}_{n,s}, \delta^{(39)}_{n,s}, \delta^{(40)}_{u,s}, \delta^{(41)}_{k,s}, \delta^{(42)}_{u,s}, \delta^{(43)}_{u,s}, \delta^{(44)}_{u,s}, \delta^{(45)}_{u,s}, \delta^{(46)}_{n,s}, \delta^{(47)}_{n,s}, \delta^{(54)}_{l}, \delta^{(55)}_{u}, \delta^{(56)}_{u})\end{aligned} \tag{1ad}$$

$$\hat{c}^{up}_u = \sum_a \hat{B}_{u,a} x^{up}_{u,a} C^{up}_u, \qquad \hat{c}^{dn}_u = \sum_a \tilde{B}_{u,a} x^{dn}_{u,a} C^{dn}_u \tag{1ae}$$

$$x^{up}_{u,a}, x^{dn}_{u,a} \in \{0,1\} \tag{1af}$$

$$\pi_u = \nu_u G_u + g_u(\hat{c}_u - C_u), \ \forall u \tag{1ag}$$

$$\hat{c}_u = \sum_a B_{u,a} x_{u,a} C_u, \ \forall u, \quad x_{u,a} \in \{0,1\}, \tag{1ah}$$

$$g_u = \sum_r E_{u,r} y_{u,r}, \quad \sum_r y_{u,r} \leq 1, \ \forall u, \quad y_{u,r} \in \{0,1\} \tag{1ai}$$

$$\begin{aligned}&\text{Linerization of bilinear terms } g_u x_{u,a},\ x^{up}_{u,a} g^{up}_{u,s},\ x^{dn}_{u,a} g^{dn}_{u,s},\ \lambda^I_{u,s} x^{up}_{u,a},\ \lambda^J_{u,s} x^{dn}_{u,a},\ \lambda^M_{u,s} x^{up}_{u,a},\\ &\quad \lambda^M_{u,s} x^{dn}_{u,a},\ \beta_{u,s} g_u,\ \varphi_{u,s} g_u,\ \mu_{k,s} \sum_{n,n:u} H_{k,n} g_u,\ \lambda^B_{k,s} \sum_{n,n:u} H_{k,n} g_u,\ \lambda^D_{u,s} g_u,\ \lambda^F_{u,s} g_u,\ \lambda^M_{u,s} g_u\end{aligned} \tag{1aj}$$

$$\text{Constraints } (1y) - (1ac), (1ag) - (1ai), \ \forall j \tag{1ak}$$

$$\text{Constraints } (1b) - (1x), (1ad) - (1af), (1aj), \ \forall i, j \tag{1al}$$

$$\mathbb{E}_s[\phi_{u,s}] \geq \mathbb{E}_s[\phi^{(i),(j)}_{u,s}], \quad \forall u, i, j \tag{1am}$$

$$\pi_u + \mathbb{E}_s[\phi_{u,s}] - \pi^{(j)}_u - \mathbb{E}_s[\phi^{(i),(j)}_{u,s}] + \zeta_u \geq 0, \ \forall u, i, j \tag{1an}$$

The set of decision variables in (1) is Φ and the elements of Φ is defined in part I of this paper [1]. Our proposed solution algorithm has the following three modules.

3.1. Module 1: Grid refinement technique

The proposed stochastic MILP (1) computes SPNE for a discrete approximation of prices. One potential problem with this approximation technique is that the price grid might be rather coarse and it may influence the SPNE outcomes. To tackle this problem we propose an iterative grid-refinement technique as illustrated in Fig. 1. In the first iteration (price grid is shown in black color) we find the SPNE which is shown as black dot in the grid. In the next iteration we build a new grid (shown in blue color) around the SPNE found in the first iteration. The new SPNE is found analogously to the first SPNE, but for an updated grid. This grid-refinement process continues until the predefined tolerance is reached. This technique iteratively improves the accuracy of the discrete approximation technique used when deriving stochastic MILP model (1).

The tolerance in iteration q is calculated by $\tau^{(q)} = \tau^{(q-1)}/(A-1)$ where A is the number of bidding actions. Based on $\tau^{(q)}$, the bid grid ($B_{u,a}$,$\hat{B}_{u,a}$,$\tilde{B}_{u,a}$) is determined. The algorithm stops when $\tau^{(q)}$ is less than or equal to the predefined tolerance, $\bar{\tau}$.

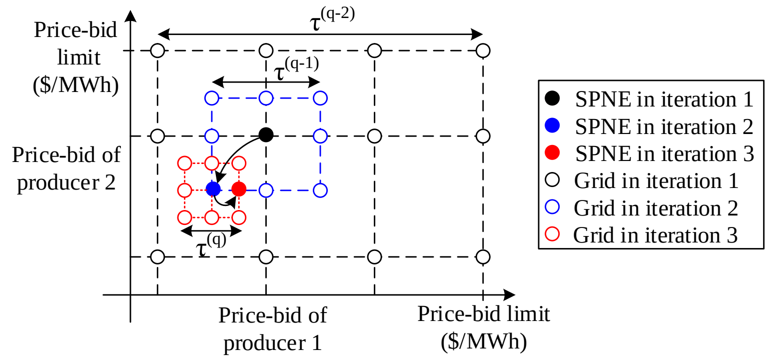

Figure 1: Iterative grid-refinement technique

3.2. Module 2: Constraint-reduction technique

The proposed stochastic MILP in (1) has constraints for the SPNE strategy and for alternative strategies for all producer. The connections between these constraints are by the profits in day-ahead market and real-time market in (1am) and (1an). Fig. 2 shows that for a given SPNE strategy (shown in blue shaded cell), we can form a set which consists all alternative strategies of each producer while holding its competitors' strategies fixed (shown in black shaded cells). Using this set, each producer's profits in day-ahead and real-time markets can be calculated for all alternative strategies separately. This allows us to remove the constraints related to the alternative strategies (constraints (1ak) and (1al)) from stochastic MILP model (1). For calculating each producer's profits for all of its alternative bidding strategies (while holding its competitors' strategies fixed), we formulate the feasibility model in (2). The bidding decisions ($\hat{c}_u$, $\hat{c}_u^{up}$ and $\hat{c}_u^{dn}$) are fixed in their values in alternative strategy (i,j). Model (2) has only $y_{u,r}$ as binary variable. Accordingly, model (2) for each alternative strategy is independent from each other so they can be run in parallel.

$$\text{Find } \hat{\Phi} = \Pi \cup \{\lambda^A_s, \lambda^B_{k,s}, \lambda^C_{u,s}, \lambda^D_{u,s}, \lambda^E_{u,s}, \lambda^F_{u,s}, \lambda^G_{n,s}, \lambda^H_{n,s}, \lambda^I_{u,s}, \lambda^J_{u,s}, \lambda^K_{u,s}, \lambda^L_{n,s}, \lambda^M_{u,s}, \lambda^N_{k,s}, \lambda^O_{u,s},$$
$$\lambda^P_{u,s}, \lambda^Q_{u,s}, \lambda^R_{u,s}, \lambda^S_{n,s}, \lambda^T_{n,s}, g_u, \xi, \gamma_l, \eta_u, \nu_u, \pi_u, y_{u,r}\} \tag{2a}$$

Such that

$$\text{Constraints} \quad (1b)-(1ad),(1ag),(1ai),(1aj) \tag{2b}$$
$$\hat{c}_u = \hat{c}_u^{(j)}, \qquad \hat{c}_u^{up} = \hat{c}_u^{up,(i),(j)}, \qquad \hat{c}_u^{dn} = \hat{c}_u^{dn,(i),(j)}, \quad \forall u \tag{2c}$$

The elements of set Π is defined in part I of this paper [1].

Table 1 shows that our constraint-reduction technique reduces the size of our SPNE model and helps us to achieve the optimal solution in shorter time. To be able to apply the constraint-reduction technique, we use the decomposition technique explained in Module 3.

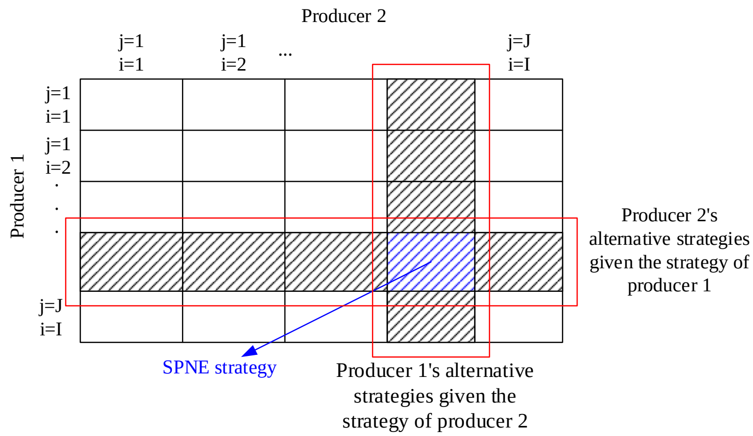

Figure 2: The concept of our constraint-reduction technique

Table 1: The size of the MILP model (1) before and after applying constraint-reduction technique, CR: Constraint reduction, Q_1=$KS(5+8A^U)+US(26 + 24A+20A^U)+13NS+3S$, Q_2=$2L+8U+2$, Q_3=$US(5A^U+6A+18)+KS(2A^U+3)+9NS+ZS+2U+2S$, Q_4=$1+L+UA+5U$, see also nomenclature in part I [1]

	MILP model (1)	
	Before CR	After CR
# of constraint	$Q_1(IJ+1)+Q_2(J+1)+2$	Q_1+Q_2+2
# of binary variables	$U(3\text{A}+A^U+AJ+AIJ+JA^U)$	$U(3A+A^U)$
# of continuous variables	$Q_3(1+JI)+Q_4(1+J)+U$	Q_3+Q_4+U

3.3. Module 3: Decomposition technique

We use modified and parallelized Benders decomposition to decompose the MILP model in (1) into a master problem and a subproblem. The master problem is a relaxation of the MILP model in (1) which contains only the binary variables. The subproblem is obtained by fixing the binary variables in the MILP model in (1). Their formulations are explained below.

3.3.1. Subproblem

Given the set of binary variables, the subproblem is a LP and formulated in (3).

$$\underset{\hat{\Psi}}{\text{Minimize}} \qquad \Upsilon = \sum_u \zeta_u \tag{3a}$$

Subject to:

$$\text{Constraints } (1b) - (1ae), (1ag) - (1aj) \tag{3b}$$

$$\mathbb{E}_s[\phi_{u,s}] \geq \mathbb{E}_s[\bar{\phi}_{u,s}^{(i),(j)}], \quad \forall u, i, j \tag{3c}$$

$$\pi_u + \mathbb{E}_s[\phi_{u,s}] - \bar{\pi}_u^{(j)} - \mathbb{E}_s[\bar{\phi}_{u,s}^{(i),(j)}] + \zeta_u \geq 0, \ \forall u, i, j \tag{3d}$$

$$\tilde{x}_{u,a} = \bar{x}_{u,a} : \Lambda_{u,a}, \ \forall u, a, \quad \tilde{x}_{u,a}^{up} = \bar{x}_{u,a}^{up} : \Lambda_{u,a}^{up}, \ \forall u, a \tag{3e}$$

$$\tilde{x}_{u,a}^{dn} = \bar{x}_{u,a}^{dn} : \Lambda_{u,a}^{dn}, \ \forall u, a, \quad \tilde{y}_{u,r} = \bar{y}_{u,r} : \ \hat{\Lambda}_{u,r} \ \forall u, r \tag{3f}$$

The set of decision variables in (3) is $\hat{\Psi}$=$\hat{\Phi} \setminus \{y_{u,r}\} \cup \{\hat{c}_u,\ \hat{c}_u^{up}, \hat{c}_u^{dn}, \tilde{x}_{u,a}, \tilde{x}_{u,a}^{up}, \tilde{x}_{u,a}^{dn}, \tilde{y}_{u,r}\}$. Here $\bar{x}_{u,a}$, $\bar{x}_{u,a}^{up}$, $\bar{x}_{u,a}^{dn}$ and $\bar{y}_{u,r}$ are the given binary decisions and $\tilde{x}_{u,a}$, $\tilde{x}_{u,a}^{up}$, $\tilde{x}_{u,a}^{up}$ and $\tilde{y}_{u,r}$ represent the continuous variables. The binary variables ($x_{u,a}$, $x_{u,a}^{up}$, $x_{u,a}^{dn}$, $y_{u,r}$) are replaced by continuous variables ($\tilde{x}_{u,a}$, $\tilde{x}_{u,a}^{up}$, $\tilde{x}_{u,a}^{up}$, $\tilde{y}_{u,r}$) in constraint (3b). Note that $\bar{\pi}_u^{(j)}$ and $\mathbb{E}_s[\bar{\phi}_{u,s}^{(i),(j)}]$ are parameters and calculated in Module 2.

If optimization model (3) has an optimal solution, the upper bound of MILP (1) is updated by $UB = min(UB, \Upsilon)$. If optimization model (3) is infeasible, the feasibility subproblem formulated in (4) is solved.

$$\underset{\tilde{\Psi}}{\text{Minimize}} \qquad \hat{\Upsilon} = \sum_u \zeta_u + \sum_s (\hat{\Gamma}_s + \tilde{\Gamma}_s) \tag{4a}$$

Subject to:

$$\text{Constraints } (1b) - (1ae), (1ag) - (1aj), (3c) - (3f) \tag{4b}$$

$$\begin{aligned}&\sigma_s \sum_u (-\hat{c}_u^{up} g_{u,s}^{up} + \hat{c}_u^{dn} g_{u,s}^{dn}) - (\alpha_s \sum_n \Delta W_{n,s} + \sum_k \mu_{k,s}(F_k - \sum_n H_{k,n}(\sum_{n:u} g_u - D_n + \\ &\quad \Delta W_{n,s})) + \sum_u (\beta_{u,s}(G_u - g_u) + \varphi_{u,s} g_u) + \sum_n \chi_{n,s}(W_n + \Delta W_{n,s})) - (\lambda_s^A \sum_n \Delta W_{n,s} + \\ &\quad \sum_k \lambda_{k,s}^B (F - \sum_n (H_{k,n}(\sum_{n:u} g_u + \Delta W_{n,s} - D_n))) + \sum_u (\lambda_{u,s}^D (G_u - g_u) + \lambda_{u,s}^F g_u - \\ &\quad \lambda_{u,s}^I \hat{c}_u^{up} \sigma_s + \lambda_{u,s}^J \hat{c}_u^{dn} \sigma_s) + \sum_n \lambda_{n,s}^H (W_n + \Delta W_{n,s})) = \hat{\Gamma}_s - \tilde{\Gamma}_s,\ \forall s\end{aligned} \tag{4c}$$

The set of decision variables in (4) is $\tilde{\Psi}$=$\hat{\Psi} \cup \{\hat{\Gamma}_s, \tilde{\Gamma}_s\}$.

3.3.2. Master problem

Given the sensitivities in iteration q, ($\Lambda_{u,a}^{(q)}$, $\Lambda_{u,a}^{up,(q)}$, $\Lambda_{u,a}^{dn,(q)}$, $\hat{\Lambda}_{u,r}^{(q)}$), we formulate the master problem in (5).

$$\underset{\hat{\Pi}}{\text{Minimize}} \qquad \varrho \tag{5a}$$

Subject to

$$\begin{aligned}&\varrho \geq \Upsilon^{(q)} + \sum_{u,a} (\Lambda_{u,a}^{(q)} (x_{u,a} - \bar{x}_{u,a}^{(q)}) + \Lambda_{u,a}^{up,(q)} (x_{u,a}^{up} - \bar{x}_{u,a}^{up,(q)}) + \Lambda_{u,a}^{dn,(q)} (x_{u,a}^{dn} - \bar{x}_{u,a}^{dn,(q)})) + \\ &\quad \sum_{u,r} (\hat{\Lambda}_{u,r}^{(q)} (y_{u,r} - \bar{y}_{u,r}^{(q)})),\ \forall q \in R\end{aligned} \tag{5b}$$

$$\begin{aligned}&0 \geq \hat{\Upsilon}^{(q)} + \sum_{u,a} (\Lambda_{u,a}^{(q)} (x_{u,a} - \bar{x}_{u,a}^{(q)}) + \Lambda_{u,a}^{up,(q)} (x_{u,a}^{up} - \bar{x}_{u,a}^{up,(q)}) + \Lambda_{u,a}^{dn,(q)} (x_{u,a}^{dn} - \bar{x}_{u,a}^{dn,(q)})) + \\ &\quad \sum_{u,r} (\hat{\Lambda}_{u,r}^{(q)} (y_{u,r} - \bar{y}_{u,r}^{(q)})),\ \forall q \in O\end{aligned} \tag{5c}$$

$$\varrho \geq \underline{\varrho} \tag{5d}$$

The set of decision variables in (5) is $\hat{\Pi}$=$\{x_{u,a}, x_{u,a}^{up}, x_{u,a}^{dn},\ y_{u,r}, \varrho\}$. Constraints (5b) and (5c) are the optimality and feasibility cuts, respectively. The lower bound of ϱ is modeled in

(5d). The objective value of optimization model (5) gives a lower bound of MILP model (1). In each iteration, the lower bound is updated by $LB = \varrho$. Note that if master problem (5) is infeasible in any iteration, the algorithm stops. It means the game does not have any SPNE.

When UB and LB are close enough, a SPNE is found. If the tolerance of the price grid is less than or equal to the predefined tolerance, the algorithm stops. Otherwise, a new price grid is selected around the found SPNE.

The whole proposed solution algorithm is illustrated in Fig 3. In Module 2, we use the fork-join-parallelization method [20].

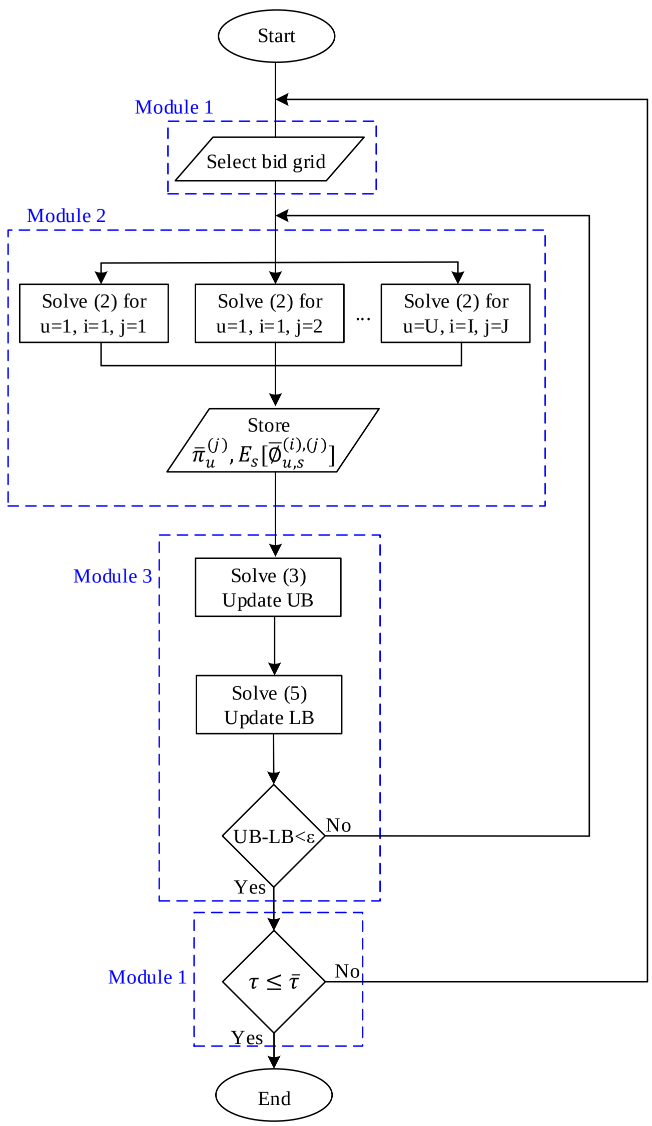

Figure 3: The flowchart of the proposed solution algorithm

4. SPNE-band approach

To compute multiple SPNE, we generalize and modify the equilibria-band concept in [18]. In the generalized version, one would build a SPNE band by placing the SPNE with the worst dispatch cost (WSPNE) into one end and the one with the best dispatch cost (BSPNE) at the other end. However, our SPNE model is much larger than the Nash-equilibrium model in [18]. Accordingly, the MILP models of WSPNE and BSPNE are hard to solve. Hence, we modify the concept. We find a lower bound where the dispatch cost is lower than the one in the BSPNE and an upper bound where the dispatch cost is higher than the one in the WSPNE for building the SPNE band.

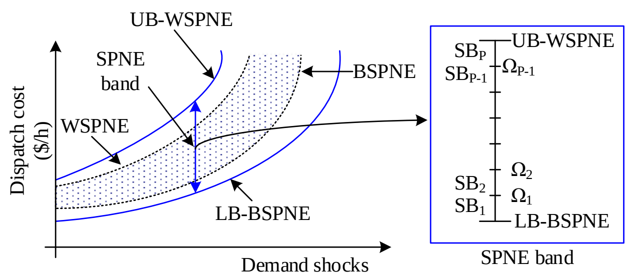

Figure 4: SPNE-band approach, UB: Upper bound, LB: Lower bound, SB_1: Subinterval 1, Ω_1: The upper bound of subinterval 1

Fig. 4 illustrates our proposed SPNE-band concept. All SPNE are located between the BSPNE and the WSPNE (shown in dashed lines). The blue lines represent the lower bound of the BSPNE (LB-SPNE) and the upper bound of the WSPNE (UB-SPNE). We split the SPNE band into several subintervals with a tolerance (ϵ) and we search for a representative SPNE in each subinterval. This is modeled by adding constraint (6) in the stochastic MILP model (1).

$$\Omega_{p-1} \leq \sum_{s,u} \sigma_s(\hat{c}_u^{up} g_{u,s}^{up} - \hat{c}_u^{dn} g_{u,s}^{dn}) + \sum_u \hat{c}_u g_u \leq \Omega_p \tag{6}$$

Constraint (6) ensures that the Total Dispatch Cost (TDC) in the found SPNE in subinterval p is between the upper and lower limits of subinterval p (Ω_p and Ω_{p-1} are parameters) as in Fig. 4. If there is no SPNE in subinterval p, the resulting stochastic MILP model becomes either infeasible or returns an objective value which is strictly greater than zero ($\sum_u \zeta_u > 0$).

The BSPNE[1] can be found by solving MILP model (7).

$$\underset{\Phi\setminus\{\zeta_u\}}{\text{Minimize}} \quad \sum_u \hat{c}_u g_u + \sum_s \sigma_s(\sum_u \hat{c}_u^{up} g_{u,s}^{up} - \hat{c}_u^{dn} g_{u,s}^{dn}) \tag{7a}$$

Subject to:

$$\text{Constraints } (1b) - (1am) \tag{7b}$$

[1] We should note that we do not find BSPNE or WSPNE in this study. BSPNE model (7) is introduced for formulating Lemma 1 and Lemma 2.

$$\pi_u + \mathbb{E}_s[\phi_{u,s}] - \pi_u^{(j)} - \mathbb{E}_s[\phi_{u,s}^{(i),(j)}] \geq 0,\ \forall u, i, j \tag{7c}$$

Lemma 1 (Lower bound of the BSPNE)
If the integrality constraints on the binary variables are relaxed in problem (7), then the solution of the resulting model gives the lower bound of the BSPNE.

Proof. If the integrality constraints on the binary variables are relaxed in (7), the resulting model becomes a relaxation of the BSPNE model and it finds a lower bound of the BSPNE. □

For finding the WSPNE[1], the minimization problem (7) should be changed to the maximization problem.

Lemma 2 (Upper bound of the WSPNE)
If the integrality constraints on the binary variables are relaxed in WSPNE model, then the solution of the resulting model gives the upper bound of the WSPNE.

Proof. If the integrality constraints on the binary variables are relaxed in the WSPNE model, the resulting model becomes a relaxation of the WSPNE model and it finds an upper bound on the WSPNE. □

Lemma 1 and Lemma 2 provide two easy-to-solve linear programs to find the lower and upper bounds within which all SPNE lie. At this stage, we divide the interval $[TDC_{LB-BSPNE}$, $TDC_{UB-WSPNE}]$ to equal subintervals SB_1, SB_2,...,SB_P. Each subinterval may or may not include a representative SPNE. We design a pre-feasibility test which detects the subintervals which has no SPNE. We start with the MILP model (1). The linear constraints related to McCormick reformulation technique are replaced by the original bilinear terms. This converts the MILP model (1) to its equivalent Mixed-Integer Bilinear Program (MIBLP)[2]. Then, we relax the binary variables in the MIBLP mode and arrives at Bilinear Program (BLP) model (8). If this BLP model is infeasible, the MIBLP is also infeasible. Accordingly, the interval in question has no SPNE.

$$\underset{\hat{\Theta}}{\text{Maximize}} \quad \sum_u \hat{c}_u g_u + \sum_s \sigma_s (\sum_u \hat{c}_u^{up} g_{u,s}^{up} - \hat{c}_u^{dn} g_{u,s}^{dn}) \tag{8a}$$

Subject to:

$$\text{Constraints } (1b) - (1ag) \text{ and constraints } (1b) - (1ag)\ \forall i, j \tag{8b}$$

$$\underline{c}_u \leq \hat{c}_u \leq \overline{c}_u : (\delta_u^{(58)}, \delta_u^{(59)}),\ \forall u \tag{8c}$$

$$\underline{c}_u^{up} \leq \hat{c}_u^{up} \leq \overline{c}_u^{up} : (\delta_u^{(60)}, \delta_u^{(61)}),\ \forall u \tag{8d}$$

$$\underline{c}_u^{dn} \leq \hat{c}_u^{dn} \leq \overline{c}_u^{dn} : (\delta_u^{(62)}, \delta_u^{(63)}),\ \forall u \tag{8e}$$

$$\mathbb{E}_s[\phi_{u,s}] \geq \mathbb{E}_s[\phi_{u,s}^{(i),(j)}] : (\delta_{u,i,j}^{(64)}),\ \forall u, i, j \tag{8f}$$

$$\pi_u + \mathbb{E}_s[\phi_{u,s}] - \bar{\pi}_u^{(i),(j)} - \mathbb{E}_s[\bar{\phi}_{u,s}^{(i),(j)}] \geq 0 : (\delta_{u,i,j}^{(65)}),\ \forall u, i, j \tag{8g}$$

$$\Omega_{p-1} \leq \sum_{s,u} \sigma_s (\hat{c}_u^{up} g_{u,s}^{up} - \hat{c}_u^{dn} g_{u,s}^{dn}) + \sum_u \hat{c}_u g_u \leq \Omega_p : (\delta^{(66)}, \delta^{(67)}) \tag{8h}$$

[2] The reason for using MIBLP is that we do not need to tune the disjunctive parameters in the McCormick-related constraints.

The set of decision variables in (8) is $\hat{\Theta}$=$\Phi \setminus \{x_{u,a}, x_{u,a}^{up}, x_{u,a}^{dn}, y_{u,r}\}$. Since the BLP model (8) is hard to solve we solve its dual program. The Lagrange multipliers related to the constraints of (8) are given in parenthesis. The dual program of nonconvex BLP (8) is formulated in (9).

$$\begin{aligned}
\underset{\tilde{\Theta}}{\text{Minimize}} \quad & \sum_{s} \delta_s^{(1)}(\sum_{n}(\Delta W_{n,s} - D_n) + \sum_{u,s} \delta_{u,s}^{(4)} G_u - \sum_{n,s} \delta_s^{(24)} \Delta W_{n,s} + \sum_{k,s} \delta_{k,s}^{(2)}(F_k - \\
& \sum_{n} H_{k,n}(\Delta W_{n,s} - D_n)) + \sum_{n,s} \delta_{n,s}^{(8)}(\Delta W_{n,s} + W_n) - \sum_{n,s} \delta_{n,s}^{(31)}(\Delta W_{n,s} + W_n) + \\
& \sum_{k,s} \delta_{k,s}^{(25)}(-F_k + \sum_{n} H_{k,n}(\Delta W_{n,s} - D_n)) - \delta^{(49)} \sum_{n} D_n - \sum_{u,s} \delta_{u,s}^{(27)} G_u + \\
& \sum_{l} \delta_l^{(50)}(\bar{F}_l + H'_{l,z}(\sum_{z:n} D_n)) + \sum_{u}(\delta_u^{(52)} G_u - \delta_u^{(58)} \underline{c}_u + \delta_u^{(59)} \bar{c}_u - \delta_u^{(60)} \underline{c}_u^{up} + \\
& \delta_u^{(61)} \bar{c}_u^{up} - \delta_u^{(62)} \underline{c}_u^{dn} + \delta_u^{(63)} \bar{c}_u^{dn}) + \delta^{(66)} \Omega_p - \delta^{(67)} \Omega_{p-1} \qquad (9a)
\end{aligned}$$

Subject to:

$$-\delta_u^{(53)} + \delta_u^{(58)} - \delta_u^{(59)} = 0, \forall u \qquad (9b)$$

$$-\sum_{s} \sigma_s(\delta_{u,s}^{(9)} + \delta_{u,s}^{(21)}) + \delta_u^{(60)} - \delta_u^{(61)} = 0, \forall u \qquad (9c)$$

$$\sum_{s} \sigma_s(\delta_{u,s}^{(10)} + \delta_{u,s}^{(22)}) + \delta_u^{(62)} - \delta_u^{(63)} = 0, \forall u \qquad (9d)$$

$$\begin{aligned}
\delta_s^{(1)} - & \sum_{n:u,k,s} H_{k,n}(\delta_{k,s}^{(2)} + \delta_{k,s}^{(25)}) \sum_{s}(\delta_{u,s}^{(4)} + \delta_{u,s}^{(6)} + \delta_{u,s}^{(27)} - \delta_{u,s}^{(29)}) + \delta^{(49)} - \sum_{u:z,l} H'_{l,z} \delta_l^{(50)} + \\
& \delta_u^{(51)} - \delta_u^{(52)} - \sum_{i,j} \delta_{u,i,j}^{(65)} C_u = 0, \ \forall u \qquad (9e)
\end{aligned}$$

$$\delta_s^{(1)} - \sum_{u:n,k} H_{k,n} \delta_{k,s}^{(2)} + \delta_{u,s}^{(3)} - \delta_{u,s}^{(4)} - C_u^{up} \delta_{u,s}^{(20)} - \sum_{i} C_u^{up} \delta_{u,i,j}^{(64)} \sigma_s - \sum_{i,j} C_u^{up} \delta_{i,j,s}^{(65)} \sigma_s = 0, \ \forall u, s \qquad (9f)$$

$$-\delta_s^{(1)} + \sum_{u:n,k} H_{k,n} \delta_{k,s}^{(2)} + \delta_{u,s}^{(5)} - \delta_{u,s}^{(6)} + C_u^{dn} \delta_{u,s}^{(20)} + \sum_{i} C_u^{dn} \delta_{u,i,j}^{(64)} \sigma_s + \sum_{i,j} C_u^{dn} \delta_{i,j,s}^{(65)} \sigma_s = 0, \ \forall u, s \qquad (9g)$$

$$-\delta_s^{(1)} + \sum_{k} H_{k,n} \delta_{k,s}^{(2)} + \delta_{n,s}^{(7)} - \delta_{n,s}^{(8)} = 0, \ \forall n, s \qquad (9h)$$

$$\sum_{u}(\delta_{u,s}^{(9)} - \delta_{u,s}^{(10)}) - \sum_{n} \delta_{n,s}^{(11)} = 0, \ \forall s \qquad (9i)$$

$$\sum_{n} H_{k,n}(\sum_{n:u}(\delta_{u,s}^{(10)} - \delta_{u,s}^{(9)}) + \delta_{n,s}^{(11)}) + \delta_{k,s}^{(12)} = 0, \ \forall k, s \qquad (9j)$$

$$\delta_{u,s}^{(9)} + \delta_{u,s}^{(13)} = 0, \ \forall u, s \qquad (9k)$$

$$\delta_{u,s}^{(15)} + \delta_{u,s}^{(16)} = 0, \ \forall u, s \qquad (9l)$$

$$\delta_{n,s}^{(17)} + \delta_{n,s}^{(18)} = 0, \ \forall n, s \qquad (9m)$$

$$-\delta^{(9)}_{u,s}+\delta^{(14)}_{u,s}+(\sum_{z:u}\overline{R}_{z,s})G_u(\delta^{(20)}_{u,s}/\sigma_s+\sum_{i}\delta^{(64)}_{u,i,j}+\sum_{i,j}\delta^{(65)}_{u,i,j})=0,\ \forall u,s \tag{9n}$$

$$\delta^{(24)}_{s}-\sum_{u:n,k}H_{k,s}\delta^{(25)}_{k,s}+\delta^{(26)}_{u,s}-\delta^{(27)}_{u,s}=0,\ \forall u,s \tag{9o}$$

$$-\delta^{(24)}_{s}+\sum_{u:n,k}H_{k,s}\delta^{(25)}_{k,s}+\delta^{(28)}_{u,s}-\delta^{(29)}_{u,s}=0,\ \forall u,s \tag{9p}$$

$$-\delta^{(24)}_{s}+\sum_{k}H_{k,s}\delta^{(25)}_{k,s}-\delta^{(31)}_{n,s}-\delta^{(30)}_{n,s}=0,\ \forall n,s \tag{9q}$$

$$\sum_{k}H_{k,s}\delta^{(25)}_{k,s}/\sigma_s+\delta^{(32)}_{n,s}-\delta^{(24)}_{s}/\sigma_s=0,\ \forall n,s \tag{9r}$$

$$(\sum_{z:u}\overline{R}_{z,s})G_u(\delta^{(27)}_{u,s}/\sigma_s+\delta^{(40)}_{u,s}=0,\ \forall u,s \tag{9s}$$

$$\delta^{(25)}_{k,s}+\delta^{(41)}_{k,s}=0,\ \forall k,s \tag{9t}$$

$$\delta^{(26)}_{u,s}+\delta^{(42)}_{u,s}=0,\ \forall u,s \tag{9u}$$

$$\delta^{(27)}_{u,s}+\delta^{(43)}_{u,s}=0,\ \forall u,s \tag{9v}$$

$$\delta^{(28)}_{u,s}+\delta^{(44)}_{u,s}=0,\ \forall u,s \tag{9w}$$

$$\delta^{(29)}_{u,s}+\delta^{(45)}_{u,s}=0,\ \forall u,s \tag{9x}$$

$$\delta^{(30)}_{n,s}+\delta^{(46)}_{n,s}=0,\ \forall n,s \tag{9y}$$

$$\delta^{(31)}_{u,s}+\delta^{(47)}_{u,s}=0,\ \forall u,s \tag{9z}$$

$$\delta^{(53)}_{u}+\delta^{(55)}_{u}=0,\ \forall u \tag{9aa}$$

$$\sum_{u}\delta^{(53)}_{u}-\delta^{(57)}\sum_{n}D_n=0 \tag{9ab}$$

$$-\sum_{z}H'_{l,z}(\sum_{u:z}\delta^{(53)}_{u})+\delta^{(54)}_{l}-\delta^{(57)}\sum_{l}(\bar{F}_l+\sum_{z}H'_{l,z}(\sum_{n:z}D_n))=0,\ \forall l \tag{9ac}$$

$$-\delta^{(53)}_{u}+\delta^{(56)}_{u}-\delta^{(57)}\sum_{u}G_u=0,\ \forall u \tag{9ad}$$

$$1+\sum_{i,j}\delta^{(65)}_{u,i,j}-\delta^{(66)}+\delta^{(67)}-\delta^{(57)}=0,\ \forall u \tag{9ae}$$

$$\sigma_s+\delta^{(20)}_{u,s}+\sigma_s\sum_{i}\delta^{(64)}_{u,i,j}+\sum_{i,j}\delta^{(65)}_{u,i,j}\sigma_s-\delta^{(66)}\sigma_s+\delta^{(67)}\sigma_s=0,\ \forall u,s \tag{9af}$$

$$\sum_{z:u}\overline{R}_{z,s}\delta^{(20)}_{u,s}/\sigma_s+\sum_{z:u}\overline{R}_{z,s}(\sum_{i,j}()\delta^{(64)}_{u,i,j}+\delta^{(65)}_{u,i,j})=0,\ \forall u,s \tag{9ag}$$

$$\sum_{z:u}\underline{R}_{z,s}(\delta^{(20)}_{u,s}/\sigma_s+\sum_{i,j}(\delta^{(64)}_{u,i,j}+\delta^{(65)}_{u,i,j}))=0,\ \forall u,s \tag{9ah}$$

$$-\sum_{z:u}\overline{R}_{z,s}\delta^{(27)}_{u,s}/\sigma_s+\sum_{z:u}\underline{R}_{z,s}\delta^{(29)}_{u,s}/\sigma_s=0,\ \forall u,s \tag{9ai}$$

$\delta^{(2)}_{k,s},\delta^{(3)}_{u,s},\delta^{(4)}_{u,s},\delta^{(5)}_{u,s},\delta^{(6)}_{u,s},\delta^{(7)}_{n,s},\delta^{(8)}_{n,s},\delta^{(12)}_{k,s},\delta^{(13)}_{u,s},\delta^{(14)}_{u,s},\delta^{(15)}_{u,s},\delta^{(16)}_{u,s},\delta^{(17)}_{n,s},\delta^{(18)}_{n,s},\delta^{(20)}_{u,s},\delta^{(33)}_{k,s},\delta^{(34)}_{u,s},$

$$\delta_{u,s}^{(35)}, \delta_{u,s}^{(36)}, \delta_{u,s}^{(37)}, \delta_{n,s}^{(38)}, \delta_{n,s}^{(39)}, \delta_{u,s}^{(40)}, \delta_{k,s}^{(41)}, \delta_{u,s}^{(42)}, \delta_{u,s}^{(43)}, \delta_{u,s}^{(44)}, \delta_{u,s}^{(45)}, \delta_{n,s}^{(46)}, \delta_{n,s}^{(47)}, \delta_{l}^{(50)}, \delta_{u}^{(51)}, \delta_{u}^{(52)},$$

$$\delta_{l}^{(54)}, \delta_{u}^{(55)}, \delta_{u}^{(56)}, \delta_{u}^{(58)}, \delta_{u}^{(59)}, \delta_{u}^{(60)}, \delta_{u}^{(61)}, \delta_{u}^{(62)}, \delta_{u}^{(63)} \geq 0 \tag{9aj}$$

$$\text{Constraints } (9a) - (9aj),\ \forall i, j \tag{9ak}$$

$$\delta_{u,i,j}^{(64)}, \delta_{u,i,j}^{(65)}, \delta^{(66)}, \delta^{(67)} \geq 0 \tag{9al}$$

In optimization problem (9), $\tilde{\Theta}{=}V \cup V^{(i),(j)} \cup \{\delta_{u,i,j}^{(64)}, \delta_{u,i,j}^{(65)}, \delta^{(66)}, \delta^{(67)}\}$ where $V{=}\{\delta_{s}^{(1)}, \delta_{k,s}^{(2)}, \delta_{u,s}^{(3)}, \delta_{u,s}^{(4)}, \delta_{u,s}^{(5)}, \delta_{u,s}^{(6)}, \delta_{n,s}^{(7)}, \delta_{n,s}^{(8)}, \delta_{u,s}^{(9)}, \delta_{u,s}^{(10)}, \delta_{n,s}^{(11)}, \delta_{k,s}^{(12)}, \delta_{u,s}^{(13)}, \delta_{u,s}^{(14)}, \delta_{u,s}^{(15)}, \delta_{u,s}^{(16)}, \delta_{n,s}^{(17)}, \delta_{n,s}^{(18)}, \delta_{n,s}^{(19)}, \delta_{u,s}^{(20)}, \delta_{u,s}^{(21)}, \delta_{u,s}^{(22)}, \delta_{n,s}^{(23)}, \delta_{s}^{(24)}, \delta_{k,s}^{(25)}, \delta_{u,s}^{(26)}, \delta_{u,s}^{(27)}, \delta_{u,s}^{(28)}, \delta_{u,s}^{(29)}, \delta_{n,s}^{(30)}, \delta_{n,s}^{(31)}, \delta_{z,s}^{(32)}, \delta_{k,s}^{(33)}, \delta_{u,s}^{(34)}, \delta_{u,s}^{(35)}, \delta_{u,s}^{(36)}, \delta_{u,s}^{(37)}, \delta_{n,s}^{(38)}, \delta_{n,s}^{(39)}, \delta_{u,s}^{(40)}, \delta_{k,s}^{(41)}, \delta_{u,s}^{(42)}, \delta_{u,s}^{(43)}, \delta_{u,s}^{(44)}, \delta_{u,s}^{(45)}, \delta_{n,s}^{(46)}, \delta_{n,s}^{(47)}, \delta_{s}^{(48)}, \delta^{(49)}, \delta_{l}^{(50)}, \delta_{u}^{(51)}, \delta_{u}^{(52)}, \delta_{u}^{(53)}, \delta_{l}^{(54)}, \delta_{u}^{(55)}, \delta_{u}^{(56)}, \delta^{(57)}, \delta_{u}^{(58)}, \delta_{u}^{(59)}, \delta_{u}^{(60)}, \delta_{u}^{(61)}, \delta_{u}^{(62)}, \delta_{u}^{(63)}\}$.

Now, we have the following lemma.

Lemma 3 (Pre-feasibility test)
If LP model (9) is unbounded in subinterval SB_p, then there is no SPNE in subinterval SB_p.

Proof. Based on duality theory, we know that (a) the dual of a nonconvex program is always convex and (b) if a primal program is infeasible, the dual program is unbounded [21]. Therefore if LP model (9) is unbounded, then primal BLP problem (8) which is a relaxation of (7) is infeasible. If the relaxation of (7) is infeasible, it is ensured that problem (7) is infeasible. In other words, there is no SPNE in subinterval SB_p. Note that our feasibility test is based on solving an easy-to-solve LP model (9). □

Using the pre-feasibility test in Lemma 3, we first solve LP model (9), to detect the subintervals without SPNE. Accordingly, we solve MILP model (1) in less number of subintervals. This improves the computational tractability for finding the set of representative SPNE. The algorithm for the SPNE-band approach is outlined in Algorithm 1. Steps 1-3 explain how the SPNE band is constructed. In Step 4, the application of pre-feasibility test and the parallelization of the SPNE search is described. After splitting the SPNE band into several subintervals, there is no communication requirement between these subintervals. Therefore the SPNE can be searched in all subintervals in parallel. We use the fork-join parallelization method in Algorithm 1. This parallel architecture is presented in Fig. 5.

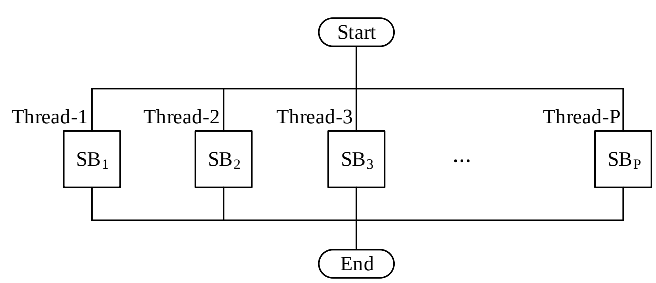

Figure 5: Application of fork-join method to our proposed SPNE-band approach

Algorithm 1: Proposed SPNE-band approach

Step 1: Calculate the lower bound of the SPNE band.
Step 2: Calculate the upper bound of the SPNE band.
Step 3: Construct SPNE band by:
(a) Decide number of subintervals by $P = log_{(1+\epsilon)} \frac{UB}{LB}$
(b) Calculate upper bound of subinterval SB_p by $\Omega_p = (1+\epsilon)\Omega_{p-1}$.
Step 4: Search a SPNE in subinterval SB_p by the steps below:
for *p=1 to P* **do**
    Solve LP model (9) for subinterval SB_p;
    if *LP model (9) is unbounded* **then**
        Report no SPNE found for subinterval SB_p;
    else
        Solve MILP model (1) for subinterval SB_p using the proposed solution algorithm.;
        if *optimal solution found* $(\sum_u \zeta_u = 0)$ **then**
            Report SPNE found for subinterval SB_p;
        else
            Report no SPNE found for SB_p;
        end
    end
end

5. Numerical Results

We use the same 6-node and modified IEEE 30-node examples presented in part I of this paper [1] to demonstrate our proposed solution algorithm and the SPNE-band approach.

Algorithm 1 is employed in 6-node system and the results are illustrated in Fig. 6. The SPNE band is split into 14 subintervals by setting ϵ to 10%. The pre-feasibility test returns unbounded solution in subintervals SB_1, SB_2 and SB_8-SB_{14}. So there is no SPNE in these subintervals. We search for a representative SPNE in SB_3, SB_4, SB_5, SB_6 and SB_7. We find a SPNE only in SB_5.

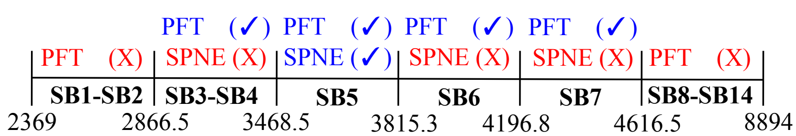

Figure 6: The SPNE band in the 6-node system, PFT: Pre-feasibility test, (X): No solution found, (✓): Optimal solution found

First, the price grid is set by the values used in part I of this paper [1] with a tolerance of $\tau^{(1)}$=20%. Then, it is reduced to $\tau^{(2)}$=10% and $\tau^{(3)}$=5% with the grid-refinement technique. We see that there is no change in the SPNE between the grid-refinement iterations for the 6-node system.

We employ Algorithm 1 to the modified IEEE 30-node system and the results are shown in Fig. 7. The SPNE band is split into 8 subintervals by setting the tolerance to ϵ=10%. The pre-feasibility test returns unbounded solutions in SB_1, SB_2, SB_7 and SB_8. A representative SPNE is searched in SB_3, SB_4, SB_5 and SB_6. The SPNE is only found in SB_4.

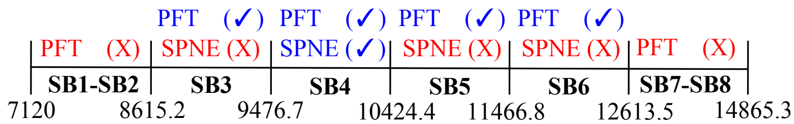

Figure 7: The SPNE band in the modified IEEE 30-node system, PFT: Pre-feasibility test, (X): No solution found, (✓): Optimal solution found

We iteratively refine the price grid. In the first iteration, the grid tolerance is set to $\tau^{(1)}$=20%. Then it is set to $\tau^{(2)}$=10% and $\tau^{(3)}$=5% in iterations 2 and 3, respectively. We see that the bids of u_1, u_2, u_3 and u_4 in the SPNE in the iterations of the grid refinement technique remain the same. Producer u_5, however, changes its up-regulation bid from 39.05 \$/MWh to 39.938 \$/MWh and this change increases its profit from 250.8 \$/h to 310.9 \$/h. We observe that improving the accuracy of the price grid of u_5 affects its profit by 24%.

We consider IEEE 118-node system [22] to further examine the impact of the grid-refinement technique in a large system. Table 2 shows the data of the producers considered in the IEEE 118-node system.

Table 2: Producer Data for IEEE 118-node system

	Node	C_u (\$/MWh)	C_u^{up} (\$/MWh)	C_u^{dn} (\$/MWh)	G_u (MW)
u_1	30	23	31.5	12.5	1200
u_2	32	22.5	30.5	11.5	1200
u_3	66	24.5	33.5	14.5	1000
u_4	92	23.5	34.5	15.5	400
u_5	77	25	35.5	16.5	1000
u_6	110	25.5	35	17	1000

In the first iteration of the grid-refinement technique the tolerance is set to $\tau^{(1)}$=40%. The profit of each producer in each iteration of the grid refinement technique is shown in Table 3. The percentages which written in bold fonts in Table 3 shows how many percent the profit changes as compared to the previous iteration. We observe that, except for u_2, each producer's total profit is affected by the improved accuracy of the price grid. The grid-refinement technique improves the tolerance of price grid from $\tau^{(1)}$=40% to $\tau^{(4)}$=5% through 4 iterations and we see that total profit of all producers are affected by 17.3%, 12.4% and 5.5% in iterations 2, 3 and 4, respectively.

To examine the computational performance of our proposed solution algorithm, we consider IEEE 300-node systems. The simulations are performed on a computer with 18 cores with Intel Xeon E5-2699 CPU and 128 GB of RAM. We use the same producers' data as in the IEEE 118-node system. Table 4 illustrates the size and solution time of each example considered in this study.

Table 3: The total expected profits of each producer in each iteration of grid-refinement technique, it: Iteration

	it. 1 ($\tau^{(1)}$=40%)	it. 2 ($\tau^{(2)}$=20%)	it. 3 ($\tau^{(3)}$=10%)	it. 4 ($\tau^{(4)}$=5%)
u_1	9658.8	12851.2 (**33.1%**)	14321.2 (**11.4%**)	15056.2 (**5.1%**)
u_2	17711.5	13545.8 (**-23.5%**)	13545.8 (0%)	13545.8 (0%)
u_3	500	980 (**96%**)	1470 (**50%**)	1715 (**16.7%**)
u_4	600	1380 (**130%**)	1870 (**35.5%**)	2115 (**13.1%**)
u_5	0	1950	3175 (**62.8%**)	3787 (**19.3%**)
u_6	0	2687.6	3135.5 (**16.7%**)	3359.5 (**7.1%**)
Total	28470.3	33394.6 (**17.3%**)	37517.5 (**12.4%**)	39578.5 (**5.5%**)

Table 4: The required time for finding one SPNE by solving MILP model (1) in all example systems, (*): No solution found after 24 hours, Var: Variables, BD: Benders decomposition, SA: Solution algorithm

	6-node	30-node	118-node	300-node
# of constraints	297,578	1,845,098	8,509,754	18,801,882
# of binary var.	591	985	1,182	1,182
# of continuous var.	116,081	647,247	2,646,900	5,784,804
BD embedded in CPLEX	55m	18h25m	*	*
Proposed SA	40m	15h45m	18h15m	22h35m

We see that Benders decomposition embedded in CPLEX can find a SPNE in the 6-node and the IEEE 30-node systems. For the IEEE 118-node and the IEEE 300-node examples, the CPLEX solver did not return any solution after 24 hours of running. This is while our proposed solution algorithm can find the optimal solution (i.e. SPNE of the game).

6. Conclusion

For large-scale examples, we propose (i) a solution algorithm to solve the proposed MILP model in part I and (ii) a method to tackle multiple SPNE situation. Our proposed solution algorithm decomposes the MILP model in part I into several subproblems and solves them iteratively. Moreover, it improves the accuracy of the price grid. For a given price grid, our proposed solution algorithm decomposes the MILP model into several subproblems and it solves them iteratively. After finding a SPNE, it improves the accuracy of the price grid and it searches for a new SPNE.

To tackle multiple SPNE, this paper proposes the SPNE-band approach. A band with upper and lower bounds is designed and it is split into several subintervals and a SPNE is searched in each subintervals using parallel computing. We also design a pre-feasibility test to find the subintervals which have no SPNE. This pre-feasibility test enables us to avoid searching for a SPNE in those intervals. The proposed solution algorithm and the SPNE-band approach is applied on the 6-node, the modified 30-node, the IEEE 118-node and the IEEE 300-node example systems. Our proposed solution algorithm can improve the tractability of the optimal solution in 6-node and the modified IEEE 30-node example systems. Moreover, it can find the optimal solution in IEEE 118-node and the IEEE 300-node example systems where the CPLEX solver cannot find it after running for 24 hours.

Acknowledgments

Mahir Sarfati was sponsored by the Swedish Centre for Smart Grids and Energy Storage (SweGRIDS). Pär Holmberg and Mahir Sarfati have been financially supported by the Jan Wallander and Tom Hedelius stiftelse and the Swedish Energy Agency.

References

[1] M. Sarfati, M. R. Hesamzadeh, P. Holmberg, Increase-decrease game under imperfect competition in two-stage zonal power markets—part i, IFN Working Papers 1253, Research Institute Of Industrial Economics (IFN) (2018).
URL `https://www.ifn.se/publikationer/working_papers/2018/1253`

[2] V. DeMiguel, H. Xu, A stochastic multiple-leader stackelberg model: Analysis, computation, and application, Operations Research 57 (5) (2009) 1220–1235. `doi:10.1287/opre.1080.0686`.

[3] V. V. Shanbhag, Decomposition and sampling methods for stochastic equilibrium problems, Ph.D. thesis, Stanford University (2005).
URL `https://web.stanford.edu/group/SOL/dissertations/uday-thesis.pdf`

[4] S. Gabriel, A. Conejo, J. Fuller, B. Hobbs, C. Ruiz, Complementarity Modeling in Energy Markets, International Series in Operations Research & Management Science, Springer New York, 2012.

[5] D. Zhang, H. Xu, Y. Wu, A two stage stochastic equilibrium model for electricity markets with two way contracts, Mathematical Methods of Operations Research 71 (1) (2010) 1–45. `doi:10.1007/s00186-009-0283-8`.

[6] C.-L. Su, Analysis on the forward market equilibrium model, Operations Research Letters 35 (1) (2007) 74 – 82. `doi:https://doi.org/10.1016/j.orl.2006.01.006`.

[7] J. F. Benders, Partitioning procedures for solving mixed-variables programming problems, Numerische Mathematik 4 (1) (1962) 238–252. `doi:10.1007/BF01386316`.

[8] L. Baringo, A. J. Conejo, Wind power investment: A benders decomposition approach, IEEE Transactions on Power Systems 27 (1) (2012) 433–441. `doi:10.1109/TPWRS.2011.2167764`.

[9] J. Soares, B. Canizes, M. A. F. Ghazvini, Z. Vale, G. K. Venayagamoorthy, Two-stage stochastic model using benders decomposition for large-scale energy resource management in smart grids, IEEE Transactions on Industry Applications 53 (6) (2017) 5905–5914. `doi:10.1109/TIA.2017.2723339`.

[10] T. Dai, W. Qiao, Finding equilibria in the pool-based electricity market with strategic wind power producers and network constraints, IEEE Transactions on Power Systems 32 (1) (2017) 389–399. doi:10.1109/TPWRS.2016.2549003.

[11] C. Ruiz, A. J. Conejo, Y. Smeers, Equilibria in an oligopolistic electricity pool with stepwise offer curves, IEEE Transactions on Power Systems 27 (2) (2012) 752–761. doi:10.1109/TPWRS.2011.2170439.

[12] D. Pozo, J. Contreras, Finding multiple nash equilibria in pool-based markets: A stochastic epec approach, IEEE Transactions on Power Systems 26 (3) (2011) 1744–1752. doi:10.1109/TPWRS.2010.2098425.

[13] M. R. Hesamzadeh, D. R. Biggar, Computation of extremal-nash equilibria in a wholesale power market using a single-stage MILP, Power Systems, IEEE Transactions on 27 (3) (2012) 1706–1707. doi:10.1109/TPWRS.2012.2187120.

[14] M. R. Hesamzadeh, D. R. Biggar, N. Hosseinzadeh, P. J. Wolfs, Transmission augmentation with mathematical modeling of market power and strategic generation expansion—part i, IEEE Transactions on Power Systems 26 (4) (2011) 2040–2048. doi:10.1109/TPWRS.2011.2145008.

[15] Y. Tohidi, M. R. Hesamzadeh, Multi-regional transmission planning as a non-cooperative decision-making, IEEE Transactions on Power Systems 29 (6) (2014) 2662–2671. doi:10.1109/TPWRS.2014.2312021.

[16] M. R. Hesamzadeh, M. Yazdani, Transmission capacity expansion in imperfectly competitive power markets, IEEE Transactions on Power Systems 29 (1) (2014) 62–71. doi:10.1109/TPWRS.2013.2280345.

[17] M. R. Hesamzadeh, N. Hosseinzadeh, P. J. Wolfs, Transmission system augmentation based on the concepts of quantity withheld and monopoly rent for reducing market power, IEEE Transactions on Power Systems 25 (1) (2010) 167–180. doi:10.1109/TPWRS.2009.2036823.

[18] M. R. Hesamzadeh, D. R. Biggar, Merger analysis in wholesale power markets using the equilibria-band methodology, IEEE Transactions on Power Systems 28 (2) (2013) 819–827. doi:10.1109/TPWRS.2012.2218134.

[19] E. Moiseeva, M. R. Hesamzadeh, D. R. Biggar, Exercise of market power on ramp rate in wind-integrated power systems, IEEE Transactions on Power Systems 30 (3) (2015) 1614–1623. doi:10.1109/TPWRS.2014.2356255.

[20] M. McCool, A. Robison, J. Reinders, Structured Parallel Programming: Patterns for Efficient Computation, Morgan Kaufmann, 2012.

[21] S. Boyd, L. Vandenberghe, Convex Optimization, Cambridge University Press, New York, NY, USA, 2004.

[22] A. Khodaei, M. Shahidehpour, S. Kamalinia, Transmission switching in expansion planning, IEEE Transactions on Power Systems 25 (3) (2010) 1722–1733. doi:10.1109/TPWRS.2009.2039946.