Computers & Operations Research 40 (2013) 2724–2739

Contents lists available at SciVerse ScienceDirect

Computers & Operations Research

journal homepage: www.elsevier.com/locate/caor

Master problem approximations in Dantzig–Wolfe decomposition of variational inequality problems with applications to two energy market models

Emre Çelebi [a,*], J. David Fuller [b]

[a] Department of Industrial Engineering, Kadir Has University, Kadir Has Cad., Cibali, Istanbul, Turkey
[b] Department of Management Sciences, University of Waterloo, 200 University Avenue West, Waterloo, Ontario N2L 3G1, Canada

ARTICLE INFO

Available online 30 May 2013

Keywords:
Dantzig–Wolfe decomposition
Equilibrium modeling
Variational inequalities
Approximation of the master problem

ABSTRACT

In this paper, a modification to Dantzig–Wolfe (DW) decomposition algorithm for variational inequality (VI) problems is considered to alleviate the computational burden and to facilitate model management and maintenance. As proposals from DW subproblems are accumulated in the DW master problem, the solution time and memory requirements are increasing for the master problem. Approximation of the DW master problem solution significantly reduces the computational effort required to find the equilibrium. The approximate DW algorithm is applied to a time of use pricing model with realistic network constraints for the Ontario electricity market and to a two-region energy model for Canada. In addition to empirical analysis, theoretical results for the convergence of the approximate DW algorithm are presented.

1. Introduction

Decomposition methods sometimes allow large-scale and complex problems to be solved in a distributed and parallel fashion that helps to overcome computational difficulties. They can reduce the memory requirements and/or increase the speed of calculations. Alternatively they can lead to a drastic simplification of the model development procedure and ease the model management and maintenance [27,29]. Generally, the scope of the complex models (e.g., related to public policy making) expands as addressing one question reveals other related questions. Therefore analyses of such models require continuous re-evaluation of the issues. Decomposition of these models allows different analysts or teams of experts to manage, analyze, re-evaluate and repeatedly run sub-models. Expected run time and errors in modeling can be reduced by using decomposition methods [27].

There are several decomposition algorithms (e.g., Dantzig–Wolfe, Benders, Lagrangian) for solving and analyzing large-scale equilibrium problems. Certain models may have a structure that some of the constraints or variables prevent the separability of the problem into subproblems. If these constraints/variables are removed, the resulting subproblems are frequently considerably easier to solve. These constraints/variables are usually referred to as "complicating" (and sometimes referred to as "common" or "linking") constraints/variables [8]. In Dantzig–Wolfe (DW) and Benders decomposition, instead of solving the original problem with complicating constraints or variables, two problems are solved iteratively, a master problem and a subproblem, i.e., original problem without complicating constraints or variables. The solution to the original model is obtained by exchanging price and quantity information among the subproblem(s) and the master problem in an iterative manner. The size of the master problem grows as new solutions (e.g., columns in DW decomposition) from subproblems are passed to master problem and hence, the requirements (e.g., computational time and memory) to solve the master problem increase at each iteration of the decomposition algorithm.

This paper presents modifications to the DW decomposition of variational inequality (VI) problems that allow for the approximation of the master problem to reduce the computational effort required to solve large-scale equilibrium problems and to facilitate the model management and maintenance.

DW decomposition of VI problems has been introduced by Fuller and Chung [16] and Chung et al. [7]. Approximation of the subproblems in DW decomposition of VI problems (single-valued) for decomposition purposes has been presented by Chung and Fuller [6] under useful assumptions. The DW decomposition of VI problems and the approximation of the subproblems have been also studied by Luna et al. [24]. They consider DW decomposition in a more general setting, i.e., for set-valued and maximal monotone VI mappings (in addition to the single-valued, continuous mappings considered by Fuller and Chung [16, Chung et al., [7] and

* Corresponding author. Tel.: +90 212 533 6532x1425; fax: +90 212 533 6515.
E-mail addresses: ecelebi@khas.edu.tr (E. Çelebi), dfuller@uwaterloo.ca (J. David Fuller).

0305-0548/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.cor.2013.05.012

Chung and Fuller [6]) as well as various kinds of subproblem approximations that were not considered by Chung and Fuller [6]. Furthermore, Luna et al. [24] consider some algorithmic enhancements, including inexact solution of the approximate subproblem, and the cheap generation of additional proposals by a projection method.

Related to DW decomposition of VI problems, Fuller and Chung [15] also apply Benders decomposition to VI problems and provide convergence results and proofs for a useful class of VI problems. Their algorithm is mainly based on DW decomposition of VI problems and they apply a DW decomposition procedure to a dual of the given VI. By converting the dual forms of DW master and subproblems to their primal forms, they derive the Benders master and subproblems. Gabriel and Fuller [17] apply Benders decomposition to solve a two-stage stochastic complementarity problem (or VI) for an electricity market equilibrium model. Egging [12] also employs Benders decomposition algorithm for large-scale, stochastic multi-period mixed complementarity problems (MCP) for various multi-stage natural gas market models accounting for market power exertion by traders. Because of the primal-dual relations of the DW and Benders' master problems, the approximation for the DW master problem presented in this paper can also be applied to Benders decomposition of VI problems.

In this paper, we firstly introduce the DW decomposition algorithm for VI problems and the approximations for the solution of the master problem in DW decomposition. Convergence analysis is also presented. Numerical investigations are performed on two models in energy markets. These models are a single-period (month) time-of-use (TOU) pricing electricity market equilibrium model with linearized DC network constraints from Çelebi [4] and a realistic two-region energy equilibrium model for Canada from Fuller and Chung [16].

2. Background

VI problems were first developed in the context of studying a class of partial differential equations that arise in the field of mechanics and defined on infinite dimensional spaces [31]. In contrast, finite dimensional VI problems have been studied for computation of economic and game theoretic equilibria. In general, a finite dimensional VI problem is defined as follows:

$$VI(G,K): \quad \text{find a vector } x^*\in K\subseteq R^n, \text{ such that :}$$
$$G(x^*)^T(x-x^*)\geq 0 \quad \forall\, x\in K \tag{1}$$

where G is a given continuous function from K to R^n, superscript T denotes the transpose, and K is a nonempty, closed and convex set. Standard conditions for existence and uniqueness of solutions to $VI(G,K)$ are provided in Harker and Pang [21], Nagurney [31] or Patriksson [34].

Many mathematical problems (e.g., system of equations, constrained and unconstrained optimization problems, complementarity problems, game theory and saddle point problems, fixed point problems, traffic assignment and network equilibrium problems) can be formulated as VI problems [31,21,2,34]. Unlike an optimization problem which has an objective function, a VI problem has a vector-valued function G, and it is equivalent to an optimization problem only if this vector-valued function is the gradient of an objective function. A necessary and sufficient condition for a differentiable G to satisfy the above condition is that the Jacobian matrix ∇G is symmetric or in other words, that G is integrable, i.e., it can be integrated to define an objective function [31]. Unfortunately, this condition does not hold in many practical problems. In this paper, we consider problems which are non-integrable (asymmetric). See Takayama and Judge [36] and Samuelson [35] for further details on "integrability" conditions.

There are different techniques or algorithms to solve such VI equilibrium models, e.g., by solving a sequence of integrable optimization problems, as in the Project Independence Evaluation System (PIES) algorithm [1], the decoupling algorithm [37], and more general algorithms for VI problems [31]. Alternatively, a VI problem can be converted to an equivalent complementarity problem and solved by Newton methods that solve a sequence of linear complementarity problems [26,25,11,14].

PIES, which was originally developed for energy modeling for US Department of Energy in the 1970s, captures many key features of large-scale equilibrium models. The PIES algorithm approximates the non-integrable equilibrium problem by a sequence of integrable problems which can be converted into equivalent optimization problems. Each iteration solves a linear programming (LP) problem after a proper step function approximation is made on an integrable approximation of the demand function [23]. This algorithm has the characteristics of the nonlinear Jacobi method for solving a system of nonlinear equations. Ahn and Hogan [1] give sufficient conditions under which the PIES algorithm converges. But, as Murphy and Mudrageda [29] point out, although PIES never met these conditions, because of demand function approximations, it usually does not fail to converge.

In our approximations for the solution of the DW master problem, we have also employed the PIES algorithm (as well as another symmetric mapping) to approximate the original mapping in the master problem (see Section 4 for details).

3. Decomposition algorithm and the approximation of the master problem for VI problems

In this section, we summarize the main results of Fuller and Chung [16], using a slightly different notation and following their presentation closely. Then, we present the algorithm with an approximation of the master problem in DW decomposition and its underlying theory of convergence.

3.1. Dantzig–Wolfe decomposition method for VI problems

We consider a VI problem with a feasible set defined by two sets of constraints. We distinguish one of these constraint sets as complicating constraints, e.g., when they are relaxed a VI subproblem is formed (and it may or may not be decomposable, but it is easier to solve or manage). Convex combinations of solutions of the subproblem, together with the complicating constraints, form the feasible set of the master problem. We first define the feasible set for the original VI as follows. All vectors are considered to be column vectors and superscript T denotes the transpose of a vector or matrix. The feasible set is

$$K=\{x\in R^n | g(x)\geq 0,\ h(x)\geq 0\}$$

where g is a mapping from R^n to R^m such that g_i is concave and continuously differentiable for all $i=1,\ldots,m$, and h is a mapping from R^n to R^l such that h_i is concave and continuously differentiable for all $i=1,\ldots,l$. Concavity of g and h ensure convexity of K. The constraints $h(x)\geq 0$ represent the complicating constraints. The vector function G maps R^n to R^n. The original VI is defined as follows:

$$VI(G,K): \quad \text{find } x^*\in K \text{ such that } G(x^*)^T(x-x^*)\geq 0 \quad \forall\, x\in K \tag{2}$$

We assume throughout this paper that (2) has at least one solution.

The feasible set for the subproblem is defined by relaxing the complicating constraints in K and it is represented as: $\overline{K}=\{x\in R^n | g(x)\geq 0\}$. The subproblem at iteration k is defined with ω^{k-1} (the dual variable vector corresponding to the complicating constraints from the previous master problem solved at iteration

$k-1$) and $\nabla h(x_M^{k-1})$ (the matrix of gradients of h evaluated at the master problem solution x_M^{k-1}). The subproblem VI is defined as follows:

$$\text{Sub}-VI^k(G-\nabla h(x_M^{k-1})^T\omega^{k-1},\ \overline{K}):\ \text{find } x_S^k\in\overline{K} \text{ such that}$$

$$(G(x_S^k)-\nabla h(x_M^{k-1})^T\omega^{k-1})^T(x-x_S^k)\geq 0 \quad \forall\ x\in\overline{K} \tag{3}$$

The feasible set for the master problem at iteration k is restricted to all convex combinations of the k solutions (or 'proposals') that have been calculated by the first k solutions of the subproblem. An additional restriction is that the complicating constraints need to be satisfied. We use the notation $X^k=[x_S^1,x_S^2,\ldots,x_S^k]$ to represent matrices whose columns are the k proposals collected from the subproblem at each iteration. The weights on the proposals in the convex combination are contained in the vector $\lambda\in R^k$. The feasible set for the master problem is defined as: $\Lambda^k=\{\lambda\in R^k | h(X^k\lambda)\geq 0,\ e^{kT}\lambda=1,\ \lambda\geq 0\}$, where $e^k\in R^k$ is a vector whose k entries are all one. Since any convex combination $X^k\lambda$ of solutions from the subproblem satisfies the constraint set $g(x)\geq 0$, there is no need to explicitly mention this set of constraints in the feasible set of the master problem. For brevity, we sometimes use the notation x_M^k to denote the solution of the master problem, λ^k, in terms of the original x variables: $x_M^k=X^k\lambda$. Note that $x_M^k\in K\subseteq\overline{K}$ (i.e., the feasible region of the subproblem contains the original problem's feasible region, since it is a relaxed version of the original problem's feasible region, without complicating constraints). Finally, the master problem at iteration k is defined as

$$\text{Master}-VI^k(H^k,\Lambda^k):\quad \text{find } \lambda^k\in\Lambda^k \text{ such that } H^k(\lambda^k)^T(\lambda-\lambda^k)\geq 0 \quad \forall\ \lambda\in\Lambda^k \tag{4}$$

where the mapping H^k from R^k to R^k is defined by $H^k(\lambda^k)^T=G(X^k\lambda)^TX^k$.

As an alternative notation, the feasible set for the master problem is also denoted by K^k

$$K^k=\{x\in R^k | h(x)\geq 0,\ x\in conv(X^k)\} \tag{5}$$

where $conv(X^k)$[1] stands for the convex hull of the points represented by X^k. Hence the master problem is more compactly defined as

$$\text{Master}-VI^k(G,K^k):\quad \text{find } x^*\in K \text{ such that } G(x^*)^T(x-x^*)\geq 0 \quad \forall\ x\in K^k \tag{6}$$

The relationship among the feasible sets can be summarized as

$$K^1\subseteq K^2\subseteq\ldots\subseteq K^k\subseteq K^{k+1}\subseteq\ldots\subseteq K\subseteq\overline{K}.$$

The DW algorithm uses the following information exchange between master problem and subproblem. The subproblem for $k=1$, i.e., Sub-VI^1 is solved with a starting guess of the value of the mapping adjustment, such as $\nabla h(x_M^0)^T\omega^0=0$, to obtain the proposals to be transferred to the matrix X^k of the master problem, thus enlarging the set Λ^k (or K^k). Then Master-VI^1 is solved[2] to estimate a new dual vector ω^1 and x_M^1. Later iterations begin with a subproblem and end with a master problem. After each subproblem is solved, a scalar quantity called the convergence gap (CG^k) is calculated from the solutions of subproblem Sub-VI^{k+1} and master problem Master-VI^k

$$CG^k=(G(x_M^k)-\nabla h(x_M^k)^T\omega^k)^T(x_S^{k+1}-x_M^k) \tag{7}$$

The algorithm terminates when a predetermined convergence tolerance, $\varepsilon>0$, is reached, i.e., $|CG^k|<\varepsilon$. The standard DW algorithm for VIs is as follows [16]:

3.1.1. Standard DW algorithm

Step 0: Set $k=0$. Choose $\nabla h(x_M^0)^T\omega^0$ and $\varepsilon>0$.

Step 1: Increment $k\leftarrow k+1$. Solve $\text{Sub}-VI^k(G-\nabla h(x_M^{k-1})^T\omega^{k-1},\overline{K})$ and place the solution x_S^k in the matrix $X^k=[X^{k-1},x_S^k]$.

If $k=1$ then go to Step 2; else
If $CG^{k-1}\geq-\varepsilon$, then STOP; else
go to Step 2.

Step 2: Solve Master-$VI^k(H^k,\Lambda^k)$. Record $\nabla h(x_M^k)^T\omega^k$. Go to Step 1.

Fuller and Chung [16] have provided useful convergence results under the following assumptions.

3.1.2. Assumptions for standard DW algorithm

1. $\overline{K}$ is bounded, and $\{\omega^k\}_{k=1}^{\infty}$ is contained in a bounded set.
2. Each component of $h(x)$ and $g(x)$ is concave and continuously differentiable.
3. G is continuous.
4. The subproblem and master problem are feasible at each iteration.
5. Either G is strictly monotone,[3] or $G=\begin{pmatrix}-p(d)\\ \nabla c(z)\end{pmatrix}$ where $x=\begin{pmatrix}d\\ z\end{pmatrix}$, $-p(d)$ has the same dimension as d and is strictly monotone, and $c(z)$ is a convex function.[4]

Under these assumptions, Fuller and Chung [16] prove several results, including: if $CG^k\geq 0$, then x_M^k solves $VI(G,K)$; before convergence, $CG^k<0$, and $\lim_{k\to\infty} CG^k=0$. Furthermore, if in Assumption 5, "strictly monotone" is replaced by "strongly monotone,[5] " then $\lim_{k\to\infty}|x_S^{k+1}-x_M^k|=0$ (if G is strongly monotone), or $\lim_{k\to\infty}|d_S^{k+1}-d_M^k|=0$ (if $-p$ is strongly monotone). We have provided the theorems of Fuller and Chung [16] in Appendix A and the proofs can be found in Fuller and Chung [16] or Chung and Fuller [6].

The boundedness of $\overline{K}$ in Assumption 1 can be achieved by imposing bounds on x variables in the subproblems. For Assumption 4, infeasibility of the master problem can be avoided by introducing artificial variables in the complicating constraints with high cost coefficients, i.e., $h_i(x)+a_i\geq 0\ \forall\ i$ with $a_i\geq 0$ [7]; this also imposes bounds on each ω^k, satisfying the second requirement of Assumption 1. However, determining the value of these high cost coefficients (i.e., 'big-M' values) for artificial variables may cause some problems. If they are too small, positive artificial variables may be observed in the solution, and if they are too large, numerical problems due to poor scaling may arise. In practice, the modeler's insight is important in determining the 'big-M' values, using the fact that they are bounds on the dual variables of the complicating constraints [6].

3.2. Approximate solution of the master problem

In this subsection we modify the standard DW algorithm by allowing the master problem to be solved approximately. First, we define the approximation method and its properties in a general setting.

At each iteration k, the master problem mapping $G(x)$ is approximated by $\tilde{G}^k(x)$, and we solve a modified master problem denoted by $\text{Master}-VI^k(\tilde{H}^k,\Lambda^k)$, where $\tilde{H}^k(\lambda^k)^T=\tilde{G}^k(X^k\lambda)^TX^k$. We assume that the accuracy of the approximate mapping $\tilde{G}^k$ can be controlled by the modeler to ensure that $|\tilde{G}^k(x_M^k)-G(x_M^k)|\to 0$ as $k\to\infty$. For example, the second set of numerical results in Section 4 are based on the same equilibrium model used by Fuller and Chung [16], who applied standard DW decomposition, with the PIES algorithm

[1] $conv\ (X^k)=\{x\in R^n | x=X^k$, for some $\lambda\in R^k$ such that $e^{kT}\lambda=1,\ \lambda\geq 0\}$.

[2] Unless x_s^1 is feasible with respect to $h(x)\geq 0$, $\text{Master}-VI^1(K^1,G)$ will be infeasible. Later, we discuss the use of artificial variables in the master problem, which avoids this difficulty.

[3] F is strictly monotone on K if $[F(x)-F(y)]^T(x-y)>0,\quad \forall x,y\in K,\quad x\neq y$.

[4] The distinction between subvectors d and z, with corresponding mapping subvectors $-p(d)$ and $\nabla c(z)$, is found in many models, including the two models of Section 4. For example, in the first model of Section 4, $p(d)$ is a vector of inverse demand functions, with cross-commodity dependence, and $c(z)$ is the cost of supply activities z to meet the demands d.

[5] F is *strongly monotone* on K if there exists $\alpha>0$ such that $|x-y|\leq[F(x)-F(y)]^T(x-y),\ \forall x,y\in K$.

solving the master problem very accurately at each iteration; in Section 4, we solve the master problem less accurately at each iteration. The PIES algorithm, at each DW iteration in Fuller and Chung [16], solves a sequence of approximate equilibrium problems, with the approximate mapping getting closer to the exact mapping G at each step in the sequence. In Section 4, we take just one PIES step at each DW iteration, but if it were necessary to have a more accurate approximate mapping, two or more PIES steps could be taken. The first set of numerical results in Section 4, using a large-scale electricity market equilibrium model, employs two different algorithms based on two approximations $\tilde{G}^k$ at each DW iteration, but again, the accuracy of the mapping can be controlled by taking more steps, when necessary, in a sequence of approximate problems.

To gain some computational advantage, $\tilde{G}^k(x)$ is chosen so that the approximate master problem VI is easy to solve. For example, in Section 4, in both sets of results, the approximate master problem is a nonlinear programming (NLP) problem, which can be efficiently solved with less computational effort by an NLP solver; the approximate mapping $\tilde{G}^k(x)$ is the gradient of the objective function of the *last* NLP in the sequence.

In Chung and Fuller [6], and in Luna et al. [24], the *subproblem* mapping is approximated, while the master problem mapping is the exact $G(x)$. In contrast, here the master problem mapping is approximated, but the subproblem mapping employs the exact $G(x)$.

It is required that the approximate $\tilde{G}^k(x)$ should be continuous. The master problem with the approximate mapping is as follows:

Master $-$ $VI^k(\tilde{H}^k, \Lambda^k)$: find $\lambda^k \in \Lambda^k$
such that $\tilde{H}^k(\lambda^k)^T(\lambda - \lambda^k) \geq 0 \quad \forall\ \lambda \in \Lambda^k$,
or equivalently,

Master $-$ $VI^k(\tilde{G}^k, K^k)$: find $x^* \in K^k$
such that $\tilde{G}^k(x^*)^T(x - x^*) \geq 0 \quad \forall\ x \in K^k$ (8)

The algorithm begins by solving the subproblem for $k=1$, then choosing an initial approximation $\tilde{G}^1(x)$ and solving Master $-$ $VI^1(\tilde{H}^1, \Lambda^1)$ (normally with artificial variables ensuring feasibility of the complicating constraints, as for the standard DW algorithm). Later iterations begin with a subproblem and end with a master problem. The convergence gap, CG^k, is calculated, using the exact mapping G. The detailed statement of the algorithm follows. Note that there are many ways to ensure that $|\tilde{G}^k(x_M^k) - G(x_M^k)| \to 0$ as $k \to \infty$; for concreteness, we present one such method, which is to require that $\tilde{G}^k$ be chosen such that $|\tilde{G}^k(x_M^k) - G(x_M^k)| < \alpha |\tilde{G}^{k-1}(x_M^{k-1}) - G(x_M^{k-1})|$, for some $\alpha \in (0,1)$.

The approximate algorithm combines two convergent processes —the Dantzig–Wolfe procedure of using subproblem proposals to approximate the master problem's feasible region, and a steadily improving approximation of the master problem's mapping G. The stopping condition therefore has two parts, as detailed in the following statement of the algorithm.

3.2.1. Approximate DW algorithm

Step 0: Set $k=0$. Choose $\nabla h(x_M^0)^T \omega^0$, $\varepsilon > 0$ and $\alpha \in (0,1)$. Set X^0 to the null matrix and $\tilde{CG}^0 = -\infty$.

Step 1: Increment $k \leftarrow k+1$. Solve Sub-VI^k with $\nabla h(x_M^{k-1})^T \omega^{k-1}$ from the previous master problem (or Step 0) and place the solution x_S^k in the matrix $X^k = [X^{k-1}, x_S^k]$.
If $k=1$ then go to Step 2; else
If $CG^{k-1} \geq -\varepsilon$, and $(\tilde{G}^{k-1}(x_M^{k-1}) - G(x_M^{k-1}))^T (x_S^k - x_M^{k-1}) < \varepsilon$ then STOP; else go to Step 2.

Step 2: Choose $\tilde{G}^k$ and calculate x_M^k such that x_M^k solves Master $-$ $VI^k(\tilde{H}^k, \Lambda^k)$ and (for $k > 1$) $|\tilde{G}^k(x_M^k) - G(x_M^k)| < \alpha |\tilde{G}^{k-1}(x_M^{k-1}) - G(x_M^{k-1})|$. Record $\nabla h(x_M^k)^T \omega^k$. Go to Step 1.

The convergence theory for the approximate algorithm (see Appendix A) requires the following assumptions about $\tilde{G}^k$, as well as the five assumptions required for the standard DW algorithm.

3.2.2. Extra assumptions for approximate DW algorithm

6. $\tilde{G}^k$ is continuous.
7. $\tilde{G}^k$ is chosen such that the solution x_M^k to Master $-$ $VI^k(\tilde{H}^k, \Lambda^k)$ satisfies $\lim_{k\to\infty} |\tilde{G}^k(x_M^k) - G(x_M^k)| = 0$.

Under Assumptions 1–7, most of the 10 theorems and proofs in Fuller and Chung [16] hold for this approximation. There are three exceptions: Theorems 3, 5 and 8, which rely on the exact mapping G being used in the master problem. Below, we provide New Theorems 3, 5 and 8, with proofs. For the reader's convenience, Appendix A contains statements of all the original theorems from Fuller and Chung [16], where proofs may be found.

New Theorem 3. λ^k solves Master $-$ $VI^k(\tilde{H}^k, \Lambda^k)$ iff there exists $\lambda^k \in R_+^k$, $\omega^k \in R_+^l$ and $\psi^k \in R$ such that all the following conditions are satisfied:

$$X^{kT}\tilde{G}^k(X^k\lambda^k) - X^{kT}\nabla h(X^k\lambda^k)^T\omega^k + e^k\psi^k \geq 0 \tag{9}$$

$$h(X^k\lambda^k) \geq 0 \tag{10}$$

$$e^{kT}\lambda^k = 1 \tag{11}$$

$$\lambda^{kT}(X^{kT}\tilde{G}^k(X^k\lambda^k) - X^{kT}\nabla h(X^k\lambda^k)^T\omega^k + e^k\psi^k) = 0 \tag{12}$$

$$\omega^{kT}h(X^k\lambda^k) = 0 \tag{13}$$

Proof. This is a standard result for VI problems; see, e.g., Harker and Pang [21].

New Theorem 5. If $CG^k + (\tilde{G}^k(x_M^k) - G(x_M^k))^T(x_S^{k+1} - x_M^k) < 0$, then $conv(X^k) \subset conv(X^{k+1})$ (strict inclusion).

Proof. We prove the converse. Suppose that $conv(X^k) = conv(X^{k+1})$, i.e. that $x_S^{k+1} \in conv(X^k)$. Then there is a vector $\bar{\lambda} \in R^k$ such that $x_S^{k+1} = X^k\bar{\lambda}$ with $e^{kT}\bar{\lambda} = 1$. In the conditions of New Theorem 3, (12) may be written as $x_M^{kT}(\tilde{G}^k(x_M^k) - \nabla h(x_M^k)^T\omega^k) + \psi^k = 0$, which may be used to substitute for ψ^k in the first k inequalities, (9), of the conditions in New Theorem 3: $X^{kT}(\tilde{G}^k(x_M^k) - \nabla h(x_M^k)^T\omega^k) - e^{kT}x_M^{kT}(\tilde{G}^k(x_M^k) - \nabla h(x_M^k)^T\omega^k) \geq 0$. Multiply each of these k inequalities by the corresponding element of $\bar{\lambda}$, and sum the results, to produce: $(\tilde{G}^k(x_M^k) - \nabla h(x_M^k)^T\omega^k)^T(x_S^{k+1} - x_M^k) \geq 0$. Finally, add and subtract $(\tilde{G}^k(x_M^k) - G(x_M^k))^T(x_S^{k+1} - x_M^k)$ on the left of this inequality to produce $CG^k + (\tilde{G}^k(x_M^k) - G(x_M^k))^T(x_S^{k+1} - x_M^k) \geq 0$.

New Theorem 8. Assume that (a) $\tilde{G}^k$ is continuous, (b) any infinite subsequence of $\{(x_M^k, \omega^k, x_S^{k+1})\}_{k=1}^{\infty}$ has at least one limit point, and (c) $\lim_{k\to\infty} |\tilde{G}^k(x_M^k) - G(x_M^k)| = 0$. Either $CG^k \geq 0$ at a finite iteration number k, or $CG^k < 0$ for all iterations k. In the latter case, $\lim_{k\to\infty} CG^k = 0$.

Proof. Suppose that $CG^k = (G(x_M^k) - \nabla h(x_M^k)^T\omega^k)^T(x_S^{k+1} - x_M^k) < 0$ for all k and suppose, contrary to our desired conclusion, that there exists an $\varepsilon > 0$ and infinite set of iteration numbers, $\mathcal{T}$, such that

$(G(x_M^k)-\nabla h(x_M^k)^T\omega^k)^T(x_S^{k+1}-x_M^k) < -\varepsilon$ for all $k\in\mathcal{T}$. Adding $(\tilde{G}^k(x_M^k)-G(x_M^k))^T(x_S^{k+1}-x_M^k)$ to both sides yields: $(\tilde{G}^k(x_M^k) -\nabla h(x_M^k)^T\omega^k)^T(x_S^{k+1}-x_M^k) < -\varepsilon+(\tilde{G}^k(x_M^k)-G(x_M^k))^T(x_S^{k+1}-x_M^k)$ for all $k\in\mathcal{T}$. For any k and j with $j>k$, x_S^{k+1} is one of the proposals available to Master – $VI^j(\tilde{H}^j,\Lambda^j)$, so we may use the complementarity conditions in New Theorem 3 to derive an inequality. We do this by examining the dual feasibility constraint (12) in Master – $VI^j(\tilde{H}^j,\Lambda^j)$ associated with the primal variable λ_{k+1}^j which is the weight associated with the proposal x_S^{k+1}: $x_S^{k+1T}\tilde{G}^j(x_M^j)- x_S^{k+1T}\nabla h(x_M^j)^T\omega^j+\psi^j\geq 0$. We may eliminate the variable ψ^j by using the complementarity slackness condition $\sum_{i=1}^j(x_S^{iT}\tilde{G}_j(x_M^j)- x_S^{iT}\nabla h(x_M^j)^T\omega^j+\psi^j)^T\lambda_i^j=0$, and using the constraint $\sum_{i=1}^j\lambda_i^j=1$, and the fact that $x_M^j=\sum_{i=1}^j\lambda_i^j x_S^i$: $\psi^j=-x_M^{jT}\tilde{G}^j(x_M^j) +x_M^{jT}\nabla h(x_M^j)^T\omega^j$. This allows us to rewrite the constraint associated with λ_{k+1}^j: $(\tilde{G}^j(x_M^j)-\nabla h(x_M^j)^T\omega^j)^T(x_S^{k+1}-x_M^j)\geq 0$, for all k and j with $j>k$. Subtracting this from the strict inequality derived earlier, yields $(\tilde{G}^k(x_M^k)-\nabla h(x_M^k)^T\omega^k)^T(x_S^{k+1}-x_M^k)-(\tilde{G}^j(x_M^j)-\nabla h(x_M^j)^T\omega^j)^T(x_S^{k+1}-x_M^j) < -\varepsilon + (\tilde{G}^k(x_M^k)- G(x_M^k))^T(x_S^{k+1}-x_M^k)$, for all $k,j\in\mathcal{T}$ with $j>k$. Note that continuity of $\tilde{G}^k$ (and $\tilde{G}^j$) makes the left of the inequality continuous in (x_M,ω,x_S). By the property that any infinite subsequence of $\{(x_M^k,\omega^k,x_S^{k+1})\}_{k=1}^{\infty}$ has at least one limit point, there exists a subset $\hat{\mathcal{T}}\subset\mathcal{T}$ such that $\lim_{k\to\infty,\,k\in\hat{\mathcal{T}}}(x_M^k,\omega^k,x_S^{k+1})=(\hat{x}_M,\hat{\omega},\hat{x}_S)$, a limit point. Also note that $\lim_{k\to\infty,\,k\in\hat{\mathcal{T}}}\{(\tilde{G}^k(x_M^k)-G(x_M^k))^T(x_S^{k+1}-x_M^k)\}=0$, because $\lim_{k\to\infty}|\tilde{G}^k(x_M^k)-G(x_M^k)|=0$. Finally, we let $j\to\infty$ through values $j\in\hat{\mathcal{T}}$, in the inequality, and then let $k\to\infty$ through values $k\in\hat{\mathcal{T}}$ (this order of limits ensures that $j>k$ throughout the limiting process), and note that $\tilde{G}^k(x_M^k)\to G(\hat{x}_M)$ and $\tilde{G}^j(x_M^j)\to G(\hat{x}_M)$. This gives the contradiction $0=(G(\hat{x}_M)-\nabla h(\hat{x}_M)^T\hat{\omega})^T(\hat{x}_S-\hat{x}_M)-(G(\hat{x}_M)-\nabla h(\hat{x}_M)^T\hat{\omega})^T(\hat{x}_S-\hat{x}_M)<-\varepsilon<0$. □

Remark 1. The requirement (b) in the original Theorem 8 and in New Theorem 8 – that any infinite subsequence of $\{(x_M^k,\omega^k,x_S^{k+1})\}_{k=1}^{\infty}$ has at least one limit point – is justified in Fuller and Chung [16] by noting that Assumption 1 ($\overline{K}$ is bounded) ensures that x_M^k and x_S^{k+1} have limit points, while the practice of introducing artificial variables in the complicating constraints creates bounds on the dual variables ω^k, which ensures that the dual variables have a limit point.

Remark 2. New Theorem 5 shows that if the convergence gap plus an additional term is less than zero, $CG^k+(\tilde{G}^k(x_M^k)-G(x_M^k))^T(x_S^{k+1}-x_M^k)<0$, then the next master problem feasible region strictly includes the previous iteration's feasible region; i.e., the algorithm continues to make progress, and it is worthwhile to continue. However, since the stopping condition in Step 1 is different from this requirement, it is important to show that, if the stopping condition is violated, a useful proposal is generated. Because there are two convergent processes in the algorithm, a proposal can be useful either because it enlarges the master feasible region, or because it brings $\tilde{G}^k(x_M^k)$ closer to $G(x_M^k)$. The stopping condition can be violated in one or both of two ways; in particular, if $(\tilde{G}^k(x_M^k)-G(x_M^k))^T(x_S^{k+1}-x_M^k)\geq\varepsilon$, then the algorithm continues, $|\tilde{G}^{k+1}(x_M^{k+1})-G(x_M^{k+1})|<\alpha|\tilde{G}^k(x_M^k)-G(x_M^k)|$ with $\alpha\in(0,1)$ (see Step 2), but the master feasible region may or may not be enlarged. If instead, $(\tilde{G}^k(x_M^k)-G(x_M^k))^T(x_S^{k+1}-x_M^k)<\varepsilon$ and the stopping condition is violated because $CG^k<-\varepsilon$, then the requirement of New Theorem 5 is satisfied, and so the master feasible region is enlarged, because

$$CG^k+(\tilde{G}^k(x_M^k)-G(x_M^k))^T(x_S^{k+1}-x_M^k)<-\varepsilon+\varepsilon=0.$$

To summarize, as the algorithm proceeds, either $\tilde{G}^k$ becomes more accurate, or the new proposal enlarges the master feasible region (or both).

3.3. Alternative forms of the master problem

In Section 4, we employ slight variations on DW decomposition as described above. For the first test model of Section 4, we form a subproblem which uses a subset of the variables in the vector x, while the master problem formulation uses the remaining variables and convex combinations of proposals from the subproblem. The test model fits the form of $VI(G,K)$, and the convergence theory as found in Gabriel and Fuller [17]. In that paper, $x=(x_1^T,x_2^T)^T$, $G(x)=(G_1(x_1)^T,G_2(x_2)^T)^T$, $g(x)=g(x_2)$ and $h(x)=h_1(x_1)+h_2(x_2)$, where x_2 is a vector containing the variables that appear in the subproblem. The convergence gap CG^k is defined over the subset of variables that appear in the subproblem, i.e., (7) above is modified as $CG^k=(G_2(x_{2M}^k)-\nabla h(x_{2M}^k)^T\omega^k)^T(x_{2S}^{k+1}-x_{2M}^k)$. When master problem approximation is applied, the expression $(\tilde{G}^k(x_M^k)-G(x_M^k))^T(x_S^{k+1}-x_M^k)$ that appears in Section 3.2 above should be replaced by $(\tilde{G}_2^{\,k}(x_{2M}^k)-G(x_{2M}^k))^T(x_{2S}^{k+1}-x_{2M}^k)$, and if $G_1(x_1)$ is calculated exactly in the master problem (i.e., only G_2 is approximated) as is true of the first test model in Section 4, then the convergence theory for the modified algorithm of Section 3.2 is valid.

For the second test model of Section 4, the subproblem decomposes into two separate models, each providing distinct sets of proposals to the master problem in matrices X_A^k and X_B^k. As is common in DW decomposition applied to optimization models, we employ two distinct sets of weights in the master problem, λ_A^k and λ_B^k, so that $x_M^k=(x_{AM}^{kT},x_{BM}^{kT})^T$, with $x_{AM}^k=X_A^k\lambda_A^k$, $x_{BM}^k=X_B^k\lambda_B^k$. The convergence theory for the standard and modified DW algorithms of Sections 3.1 and 3.2 is unaffected by this variation.

4. Numerical results

In this section, we illustrate the approximate solution of the DW master problem on two energy market equilibrium models. They are briefly described in the subsequent sections. All models are coded in GAMS using a Windows 2003 Server, Dual Core AMD Opteron 2.6 GHz computer with 32 GB memory. The VI problems are solved by the GAMS/EMP framework and the PATH solver, and the convex optimization problems are solved by the NLP solvers CONOPT3 (Section 4.1) or MINOS5 (Section 4.2). The MCP models are also solved by the PATH solver in GAMS.

4.1. TOU pricing models with transmission network constraints

We illustrate the algorithms of Sections 3.1 and 3.2 using the TOU pricing models with linearized DC network constraints, as detailed and illustrated in Chapter 3 of Çelebi [4] (also see Çelebi and Fuller [5] for similar models without network constraints). These models use the Hobbs' [22] framework to determine TOU prices in Nash–Cournot game setting in electricity markets on a linearized DC network with line limits. The models were calibrated to represent the Ontario electricity system, with a 66-bus approximation of the transmission system. See Appendix B for model description, VI formulation and the DW algorithms.

We first solved the original models (perfect competition and Nash–Cournot cases) using the MCP and VI formulations, without decomposition. Computational results showed that there is not much difference in solution times and number of PATH iterations for MCP and VI formulations without line limits. However, with line limits the VI solved by GAMS/EMP with PATH is considerably faster than the MCP solved by PATH. The MCP formulation takes 18,240 (~5)–22,672 (~7) seconds (hours) and 163,876 to 256,281 PATH iterations to reach the equilibrium solution for perfect competition and Nash–Cournot models, respectively. On the other hand, the VI formulation takes 54.5–69.9 seconds and 735–483 PATH iterations to reach the same equilibrium solution for perfect competition and Nash–Cournot models, respectively. The GAMS/EMP framework, in fact, converts the VI formulation into an equivalent MCP formulation and with this conversion process some additional equation/variable pairs are added to the new MCP formulation. We have noted that the MCP (VI) formulation has 16,500 (16,662) and 16,824 (17,310) equation/variable pairs for perfect competition and Nash–Cournot models, respectively. These additional equation/variable pairs are most probably the reason for a faster computation of the equilibrium. Since there is no technical documentation about the GAMS/EMP framework yet (other than a general guide in [13]), at this point, we can presume that these added equation/variable pairs make the VI formulation better for PATH than its pure MCP counterpart. As pointed out in Garcia et al. [19], an increase in dimension can improve the formulation for integer programs and for certain structured LPs and NLPs; perhaps something similar happens in the EMP conversion of a VI to an MCP. Hence we have used the VI formulation and GAMS/EMP framework for models in the rest of our computational results.

We next applied standard DW decomposition of Section 3.1, with the complicating constraints $h(x)$ as the market clearing conditions at the transmission buses, i.e., for each hour and bus in the model, total demand at the bus minus total generation at the bus equals net injection from transmission lines into the bus. Once they are eliminated the ISO's problem can be separated from the firms' problems but since prices at a bus and demand block depends on all firms' sales, the firms' problems cannot be decomposed by firm. In our computational illustrations, the subproblem was left as one large subproblem and it was not split into two separate subproblems for the ISO and the firms; this had no significant consequences for the computing times in our experiments because, in all our experiments, almost all the computation times are spent for the master problem. In the master problem for standard DW decomposition, we used separate λ weights for the ISO and firms subproblem proposals, as discussed in Section 3.3.

For all variations of the DW algorithms, computations for subproblem and master problem start from their equilibrium solution found in the previous iteration. This may be computationally advantageous for the iterations that use slightly modified data from the previous iteration (e.g., it may reduce the number of iterations required for convergence) [30]. The convergence gap, CG^k, at each iteration of all the DW algorithms is an economically meaningful term—e.g., for the perfect competition model, the absolute value of CG^k measures the increase in the overall producers' surplus, comparing the subproblem solution with the previous master problem solution. Therefore, the convergence requirement that $CG^{k-1} \geq -\varepsilon$ ensures that, at the market clearing bus prices calculated by the master problem, the next subproblem can find a solution that increases total producers' surplus by no more than the small amount ε.

For all algorithms in this subsection, we have set $\varepsilon = 0.1$ and the dual variables of the market clearing constraints $\beta^0_{njh} = 0$ (β^0_{njh} is this model's symbol for components of the vector ω^k of Section 3). For approximations of the master problem (see Appendix B, approximate DW algorithm), we have set φ^0 = the demand quantities in x^* (where x^* is the equilibrium solution for the perfect competition model without line limits). We did not implement the requirement $|\tilde{G}^k(x^k_M) - G(x^k_M)| < \alpha |\tilde{G}^{k-1}(x^{k-1}_M) - G(x^{k-1}_M)|$, and so there was no need to choose a value for α; see below for further details on this point. In the master problem, artificial variables are added to the constraint set with large cost coefficients (e.g., big-M). After some experimentation, big-M values are set to 325. Also in the subproblem, large upper/lower bounds are set for all x variables that do not already have any upper/lower bounds.

We were not able to reach an equilibrium solution for the standard DW algorithm for the TOU pricing models with 66-bus network (with or without line limits) within a reasonable time framework and we terminated the algorithm after 48 hours of computation time. This is expected, because once the market clearing constraints at buses are relaxed in the subproblem, the links between the ISO's problem and the firms' problems are also disconnected. Proposals from the subproblem are so far from satisfying the relaxed constraints that artificial variables become positive in most of the iterations (i.e., no feasible solution is found). The convergence theory is not violated—for a smaller 3-bus test system, the standard DW algorithm found a feasible solution after 30 iterations and converged to the equilibrium solution after 84 iterations. The problem with the 66-bus system is that there are many complicating constraints $-24 \times 66 = 1584$ of these constraints, versus only 3 in the smaller test problem.

For the standard DW algorithm, we also observed that the master problem weights λ for the ISO's variables (see Section 3.3) were all positive at least once during the algorithm. This suggests the idea to include the ISO's problem in the master problem instead of the subproblem, as discussed in Section 3.3. Since the market clearing conditions are eliminated from the subproblem's feasible set $\overline{K}$, in the standard DW algorithm, the ISO's problem can be moved into the master problem, leaving only the firms represented in the subproblem. Moreover, to ensure feasibility for the master problem and to produce better proposals from subproblems, we add extra constraints to the subproblem's feasible set $\overline{K}$. These extra subproblem constraints are obtained by summing the market clearing constraints over all buses, to produce constraints that require that total generation equals total demand during every hour (without mentioning the ISO's variables). This algorithm, with the ISO's problem in the master problem and the extra subproblem constraints, is denoted the 'modified' DW algorithm; details can be found in Appendix B. As mentioned in Section 3.3, this modified DW algorithm fits the form of $VI(G,K)$, and the convergence theory as found in Gabriel and Fuller [17]. Although the extra subproblem's constraints change the subproblem feasible set $\overline{K}$, the original model's feasible set is still contained in the subproblem's feasible set, $K \subseteq \overline{K}$, which is crucial property needed in the convergence theory.

Without line limits, the modified DW decomposition method converges in only one iteration. This is not surprising, because without line limits, the congestion based wheeling fees would be zero and using a starting guess of zero for the duals of linking constraints (i.e., $\beta^0_{njh} = 0$) would cause the subproblem to provide the equilibrium solution.

With line limits, for the perfect competition model (Nash–Cournot model), the modified DW algorithm converged in 125 (307) iterations, taking more than 1.2 hours (4.2 hours), with the master problem calculations taking most of the time, see Tables 1 and 2. Therefore, we seek better computational results with master problem approximation as in Section 3.2 (see Appendix B for details for this model). We test two types of approximations: a symmetrization of the Jacobian of G (labeled 'Approximate' in the tables) and a diagonalization of G as in the PIES algorithm (labeled 'Approximate (PIES)' in the tables).

These approximations for the inverse demand functions (perfect competition model) or marginal revenue functions (Nash–Cournot model) in the mapping $\tilde{G}^k(x)$ satisfy the approximation properties. Since $\tilde{G}^k(x)$ becomes the gradient of a convex function, it can be integrated to form a convex optimization problem, e.g., NLP.

As mentioned above, we did not implement the requirement $|\tilde{G}^k(x_M^k)-G(x_M^k)| < \alpha|\tilde{G}^{k-1}(x_M^{k-1})-G(x_M^{k-1})|$ for the master problem approximation algorithms; instead, we solved only two NLP for the approximate master problem at each iteration, and calculated the implied value $\alpha = |\tilde{G}^k(x_M^k)-G(x_M^k)|/|\tilde{G}^{k-1}(x_M^{k-1})-G(x_M^{k-1})|$, as reported in Tables 3 and 4 (α columns). These computed values of α are all less than 0.125.

In the approximate DW algorithm, we have the two-part stopping condition $CG^{k-1} \geq -\varepsilon = -0.1$ and $(\tilde{G}^k(x_M^k)-G(x_M^k))^T(x_S^{k+1}-x_M^k) < \varepsilon = 0.1$. We note that the latter condition has been satisfied at every iteration of the DW algorithm, except for some very early iterations in the perfect competition case (see Tables 3 and 4, $(\tilde{G}^k-G)^T(x_S^{k+1}-x_M^k)$ columns). As mentioned in Section 3.2, Remark 2, accuracy of the approximation $\tilde{G}^k$ is enough (usually $(\tilde{G}^k-G)^T(x_S^{k+1}-x_M^k)$ term equals to zero or less than the tolerance set, $\varepsilon = 0.1$, for most of the iterations, and becomes much smaller as the algorithm proceeds) to ensure that whenever $CG^{k-1} > -\varepsilon$, the next proposal enlarges the master feasible region.

Computational results and progress of DW iterations for all algorithms are summarized in Tables 1–5, for the cases with line limits.

For the cases without line limits, the approximate DW algorithms converge, again, in only one iteration. With line limits, it takes about 42 minutes (3.3 hours) to converge to the equilibrium solution for the perfect competition (Nash–Cournot) model. The computational results in Tables 1 and 2 show that the approximate DW algorithms outperform the standard and modified DW algorithms in terms of computation time. However, the number of DW iterations is sometimes larger for the approximate DW algorithm than for the modified DW algorithm: the NLP computation at each approximate master problem is faster than the EMP/PATH master calculation of the modified DW algorithm, for less computation time overall.

The approximate (PIES) DW algorithm converges in the least time of all tested algorithms, and with fewer iterations than the approximate DW algorithm.

In Tables 3 and 4, we have presented the progress of iterations for DW algorithms for perfect competition (PC) and Nash–Cournot (NC) TOU pricing models, respectively. On the other hand, Table 5 displays the accuracy of prices at each DW iteration for approximate DW algorithms for perfect competition and Nash–Cournot TOU pricing models, i.e., accuracy measured as maximum of the relative percentage of differences in prices between solutions of master problem (denoted by subscript M) and subproblem (denoted by subscript S)/original problem (denoted by superscript $*$). After convergence, the accuracy of most decomposition algorithms, by either measure, is 0.01%, but for the approximate (PIES) algorithm in perfect competition TOU pricing case, the accuracy is worse, at 0.1%. The difference is mainly due to fewer DW iterations required for the approximate (PIES) algorithm. If this algorithm is run for several more DW iterations (e.g., by imposing a tighter convergence tolerance ε), the accuracy is improved.

This approach of approximation of the master problem allows one team of analysts to manage/maintain generation companies' problems (the subproblem) and another team to manage/maintain the system operator's problem (the master problem). In practice, this could be especially useful for the system operator, since network constraints are constantly monitored and updated by the system operator. Although the standard DW decomposition algorithm (with the system operator's problem in the subproblem) takes impractically long time to converge, the modified DW and

Table 1
Computational results for the DW algorithms for perfect competition TOU pricing models (with line limits).

DW algorithm	***Computation time (s)***			***DW iterations***
	Sub	*Master*	*Total*	
Standard	*N/A*	*N/A*	*> 48 hours*	*> 500*
Modified	42.1	4336.7	4378.8	125
Approximate	51.6	2469.4	2521.0	145
Approximate (PIES)	43.1	1891.6	1934.7	133

Table 2
Computational results for the DW algorithms for Nash–Cournot TOU pricing models (with line limits).

DW algorithm	***Computation time (s)***			***DW iterations***
	Sub	*Master*	*Total*	
Standard	*N/A*	*N/A*	*> 48 hours*	*> 500*
Modified	138.1	14,798.1	14,936.2	307
Approximate	124.5	11,674.5	11,799.0	317
Approximate (PIES)	117.8	9,992.6	10,110.4	295

Table 3
Progress of iterations, approximate DW algorithms (perfect competition, line limits).

DW iteration	**PC-APPROX**			**PC-APPROX-PIES**		
	CG^k	$(\tilde{G}^k-G)^T(x_S^{k+1}-x_M^k)$	α	CG^k	$(\tilde{G}^k-G)^T(x_S^{k+1}-x_M^k)$	α
2	−1.95E+07	1.37E+03	5.29E−03	−1.95E+07	−1.86E+03	2.86E−03
10	−6.47E+05	2.81E−13	0	−6.47E+05	4.27E−08	0
20	−3.67E+05	−5.02E−09	0	−3.67E+05	−2.69E−07	0
30	−6.42E+04	3.07E−11	0	−6.42E+04	1.71E−10	0
40	−8.62E+03	−2.33E−11	0	−7,006.29	−6.91E−12	0
50	−5.39E+03	2.03E−03	2.23E−03	−1.14E+04	1.34E−11	0
60	−1.88E+03	−5.58E−11	0	−1.82E+03	−1.06E−02	2.26E−02
70	−1.51E+03	1.18E−03	1.16E−02	−1.68E+03	−4.87E−11	0
80	−9.22E+02	2.22E−03	6.92E−03	−2.34E+03	−1.40E−02	1.20E−02
90	−1.41E+02	7.70E−05	7.30E−03	−1.97E+02	1.93E−11	0
100	−4.48E+01	−5.87E−06	3.73E−03	−3.47E+01	−9.03E−04	2.76E−02
110	−1.11E+01	1.37E−06	3.21E−03	−1.04E+01	5.42E−05	6.31E−02
120	−5.42E+00	4.19E−07	1.23E−02	−6.50E−01	−3.83E−07	1.23E−01
130	−9.91E−01	8.93E−08	3.59E−03	−1.44E−01	−5.36E−07	4.08E−02
133	−6.52E−01	8.92E−08	8.27E−03	**−8.74E−02**	1.61E−08	**5.28E−02**
145	**−1.99E−02**	**5.32E−08**	**6.26E−03**	*N/A*	*N/A*	*N/A*

Table 4
Progress of iterations, approximate DW algorithms (Nash–Cournot, line limits).

DW iteration	NC-APPROX			NC-APPROX-PIES		
	CG^k	$(\tilde{G}^k - G)^T\ (x_S^{k+1} - x_M^k)$	α^∞	CG^k	$(\tilde{G}^k - G)^T\ (x_S^{k+1} - x_M^k)$	α
2	−4.96E+07	0	0	−4.96E+07	0	0
20	−8.17E+05	−6.86E−11	0	−8.10E+05	1.23E−10	0
40	−4.26E+05	−2.61E−05	2.47E−05	−3.94E+05	3.86E−11	0
60	−2.77E+05	−2.80E−04	7.50E−05	−1.43E+05	−7.78E−01	2.40E−02
80	−1.05E+05	−1.64E−03	2.10E−03	−1.30E+05	−8.83E−04	1.90E−02
100	−1.08E+05	5.72E−03	3.02E−04	−6.94E+04	−4.87E−01	2.61E−02
120	−8.22E+04	3.74E−06	1.96E−03	−6.11E+04	5.71E−03	7.98E−03
140	−4.07E+04	−1.15E−04	1.23E−03	−3.40E+04	−2.94E−02	1.24E−02
160	−2.19E+04	3.41E−04	5.79E−04	−1.45E+04	9.32E−03	7.85E−03
180	−1.25E+04	1.49E−04	1.71E−03	−8.34E+03	−2.35E−02	4.09E−02
200	−6.65E+03	6.80E−04	3.31E−03	−3.91E+03	−1.39E−02	1.27E−02
220	−2.62E+03	−2.16E−05	1.05E−03	−1.23E+03	3.51E−03	2.24E−02
240	−1.19E+03	5.30E−04	7.83E−03	−1.08E+02	1.91E−05	2.93E−02
260	−4.09E+01	9.53E−06	6.29E−03	−4.41E+00	−3.71E−06	1.73E−02
280	−8.14E+00	1.86E−06	9.34E−03	−9.04E−01	−5.10E−07	1.30E−02
295	−1.37E+00	1.10E−06	6.74E−03	**−4.01E−02**	**−3.09E−07**	**1.39E−02**
317	**−9.13E−02**	**5.39E−08**	**9.32E−03**	*N/A*	*N/A*	*N/A*

Table 5
Maximum of relative error in prices (in percentages) for approximate DW algorithms (perfect competition and Nash–Cournot TOU pricing models with line limits).

DW iter.	PC-APPROX		PC-APPROX-PIES		DW iter.	NC-APPROX		NC-APPROX-PIES	
	$\left\lvert\frac{(p_M-p_S)}{p_M}\right\rvert$	$\left\lvert\frac{(p_M-p^*)}{p_M}\right\rvert$	$\left\lvert\frac{(p_M-p_S)}{p_M}\right\rvert$	$\left\lvert\frac{(p_M-p^*)}{p_M}\right\rvert$		$\left\lvert\frac{(p_M-p_S)}{p_M}\right\rvert$	$\left\lvert\frac{(p_M-p^*)}{p_M}\right\rvert$	$\left\lvert\frac{(p_M-p_S)}{p_M}\right\rvert$	$\left\lvert\frac{(p_M-p^*)}{p_M}\right\rvert$
2	100.00	17.72	812.66	135.12	**2**	156.75	34.26	343.24	50.77
10	57.34	4.67	57.31	4.67	**20**	23.25	28.54	23.44	28.54
20	55.93	4.65	55.97	4.65	**40**	8.31	22.45	6.07	22.17
30	42.61	3.93	42.16	3.93	**60**	3.83	20.01	8.64	19.92
40	4.63	1.68	18.32	1.31	**80**	9.05	17.47	5.07	17.50
50	18.61	1.41	12.29	0.99	**100**	5.88	14.85	8.49	14.17
60	15.64	1.72	4.62	1.64	**120**	5.06	12.63	5.51	13.44
70	13.37	1.19	13.08	1.29	**140**	6.32	12.03	6.86	9.97
80	3.98	0.82	9.30	1.22	**160**	4.52	8.68	4.80	6.85
90	1.52	0.54	2.58	0.82	**180**	5.01	6.77	3.77	5.24
100	0.39	0.35	1.01	0.69	**200**	3.69	4.81	2.52	3.33
110	0.35	0.34	0.59	0.59	**220**	2.02	2.62	0.87	1.28
120	0.34	0.33	0.13	0.11	**240**	0.94	1.33	0.45	0.48
130	0.16	0.13	0.11	0.09	**260**	0.38	0.37	0.11	0.11
133	0.15	0.12	**0.10**	**0.09**	**279**	0.13	0.13	0.03	0.03
145	**0.01**	**0.01**	*N/A*	*N/A*	**295**	0.05	0.05	**0.01**	**0.01**
					317	**0.01**	**0.01**	*N/A*	*N/A*

the approximate DW algorithms improve the computation time substantially, especially for the perfect competition model.

4.2. Two region Canadian energy model

In this subsection, we provide numerical results on the performance of the approximation of the master problem in the DW decomposition algorithm using a realistic two-region energy equilibrium model for Canada. The complicating constraints, relaxed in the subproblem, are the import/export balances between two regions. See Fuller and Chung [16] for the model description. The potential model management advantage of decomposition is to allow different teams to manage/maintain the two regional models.

Fuller and Chung [16] illustrated their numerical results using the PIES algorithm (with several PIES steps until a very precise solution was found) to solve for the master problem and the subproblem at each iteration of standard DW decomposition. The approximate DW algorithm of Section 3.2 proposes that instead of using the exact mapping $G(x)$ for the equilibrium master problem, an integrable approximation of it ($\tilde{G}^k(x)$) can be used. The PIES algorithm also approximates the original mapping of the equilibrium problem by an integrable one—see Ahn and Hogan [1] for further details on PIES algorithm. It solves the equilibrium problem iteratively until there is not much change in the solution of two consecutive iterations. The approximate DW algorithm also proposes to solve the master problem as a series of approximate master problems. Instead of an exact solution, even an approximate solution (i.e., with one or more steps of PIES) within the DW algorithm is sufficient for convergence, as we illustrate below. We have tested the approximate DW algorithm on a two-region energy equilibrium model for Canada and provide the results for three cases (where ε set to 10^{-4} for all cases):

(A) PIES with up to 25 steps for both the subproblem and master problem, i.e., standard DW decomposition.
(B) PIES with up to 25 steps for the subproblem and only one PIES step for the master problem, i.e., approximate DW decomposition.
(C) PIES with only 1 step for both the subproblem and master problem.

Table 6
Progress of iterations for the three DW algorithms for the two-region Canadian energy equilibrium model (Cases A, B and C).

DW iter.	Case A			Case B			Case C		
	CG^k	$\left\lvert\frac{(p_M-p_S)}{p_M}\right\rvert$	*Extra term*	CG^k	$\left\lvert\frac{(p_M-p_S)}{p_M}\right\rvert$	*Extra term*	CG^k	$\left\lvert\frac{(p_M-p_S)}{p_M}\right\rvert$	*Extra term*
1	−11262.157	944.243	3.53E−05	−11267.200	944.243	1.87E+00	−13699.659	944.250	−1.68E+01
5	−20317.036	37.006	1.16E−04	−505.152	22.131	−6.05E−01	−47332.506	322.908	1.88E+00
10	−49.441	2.753	9.17E−06	−41.205	8.063	−3.13E−01	−30.493	8.779	7.58E−02
15	−7.538	1.171	9.65E−06	−1.329	0.483	−1.55E−02	−25.336	5.355	3.48E−03
20	−0.199	0.140	−1.53E−06	−0.106	0.124	−1.15E−03	−1.210	0.665	−4.36E−03
25	−0.014	0.033	8.71E−07	−0.002	0.015	−2.44E−05	−0.183	0.246	7.46E−04
31	**1.04E−05**	**7.03E−03**	**−3.03E−07**	1.19E−05	6.58E−03	2.18E−06	−2.08E−03	1.89E−02	−1.82E−05
33	*N/A*	*N/A*	*N/A*	**1.31E−05**	**6.23E−03**	**5.86E−07**	−2.05E−03	1.84E−02	1.01E−05
38	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*	**−7.02E−05**	**1.35E−02**	**−3.75E−07**

Case C mixes approximation of the master problem, as in Section 3.2, with approximation of the subproblem as in Chung and Fuller [6]. The results on progress of the iterations for these three cases are summarized in Table 6.

In Table 6, the "CG^k" column shows the convergence gap at each decomposition iteration as in Fuller and Chung [16] and the "$|(p_M-p_S)/p_M|$" column is the maximum, overall price values in vector p, of the absolute difference between the master problem solution and the subproblem solution, expressed as a percent of the master problem solution. The column "*Extra Term*" is the $(\tilde{G}^k(x_M^k)-G(x_M^k))^T(x_S^{k+1}-x_M^k)$ term at each DW iteration. Similar to Section 4.1 results, the accuracy of the approximation $\tilde{G}^k$ is enough here (i.e., $(\tilde{G}^k(x_M^k)-G(x_M^k))^T(x_S^{k+1}-x_M^k)$ approaches to zero as the algorithm proceeds) to ensure that whenever $CG^{k-1} > -\varepsilon$, the next proposal enlarges the master feasible region (see Section 3.2, Remark 2).

The numbers of DW iterations required for the approximate DW algorithms in Cases A, B and C are 31, 33 and 38, respectively. For the master problem, the maximum of the number of PIES steps required[6] at each DW iteration for the whole DW algorithm for each case are 17, 1 and 1, respectively. Similarly, for the subproblem of region 1, the maximum of the number of PIES steps required for each case are 6, 7 and 1, respectively. For the subproblem of region 2, in each case, they are 7, 22 and 1, respectively.

In Table 7, we have also compared the reference solution (i.e., original model solution without decomposition and up to 25 PIES steps) with the solutions of the DW algorithms in each case. This can be regarded as a measure of accuracy for each case. In this respect, the measure "$|(q_M-q^*)/q^*|$" is the maximum, overall demand quantities in vector q, of the absolute difference between master problem's solution and the reference solution, expressed as a percent of the reference solution. Similarly, "$|(p_M-p^*)/p^*|$" is an accuracy measure for the price vector p. Table 7 presents these measures.

Case B has the greatest accuracy after 33 DW iterations and for Cases A and C, we can have a better degree of accuracy with further DW decomposition iterations. Moreover, as Table 8 shows, the extra time to compute subproblem solutions in Case B compared with A is more than offset by the computational improvement for the master problem in Case B. Case C is the least accurate, but also the fastest of the three cases. Although it takes the most DW iterations (Table 8), Case C takes the least time for both the master and subproblem calculations, due to the NLP approximations for both master and subproblem.

[6] PIES method with up to 25 steps has the convergence condition that there should be no more than 0.1% change in the price of any demand commodity from one PIES iteration to the next.

Table 7
Accuracy of the three DW algorithms for Canadian energy equilibrium model.

DW iter.	Case A		Case B		Case C	
	$\left\lvert\frac{(q_M-q^*)}{q^*}\right\rvert$	$\left\lvert\frac{(p_M-p^*)}{p^*}\right\rvert$	$\left\lvert\frac{(q_M-q^*)}{q^*}\right\rvert$	$\left\lvert\frac{(p_M-p^*)}{p^*}\right\rvert$	$\left\lvert\frac{(q_M-q^*)}{q^*}\right\rvert$	$\left\lvert\frac{(p_M-p^*)}{p^*}\right\rvert$
1	288.87%	94.52%	288.87%	330.77%	380.70%	94.52%
5	69.07%	3295.58%	69.09%	3296.18%	67.38%	5532.41%
10	18.76%	103.15%	34.28%	175.85%	17.53%	189.06%
15	7.62%	15.18%	3.93%	9.78%	13.45%	59.63%
20	2.30%	7.09%	3.11%	9.45%	4.06%	25.95%
25	1.15%	2.29%	0.50%	0.95%	1.91%	13.06%
31	0.17%	0.42%	0.12%	0.44%	0.29%	1.23%
33	N/A	N/A	0.12%	0.42%	0.47%	1.11%
38	N/A	N/A	N/A	N/A	0.18%	0.99%

Table 8
Computation times (in seconds) for the three DW algorithms and reference case (original model) for energy equilibrium model of Fuller and Chung [16].

Cases	***Master problem***	***Subproblem for region 1***	***Subproblem for region 2***	***Total***	***% of Case A***
Case A	0.924	0.312	0.425	***1.661***	*100*
Case B	0.142	0.377	0.641	***1.16***	*69.84*
Case C	0.047	0.359	0.167	***0.573***	*34.50*
Reference case	–	–	–	***0.045***	*2.71*

The computational results are quite different from those of Section 4.1, where the master problem approximations greatly improve the computational times compared with the standard (or modified) DW algorithm. In Section 4.1, the total computational time is dominated by the master problem calculations, but for the Canadian energy model, the subproblem computational time is comparable to that of the master problem. Therefore, if the master approximation causes more iterations to be required, then the total computational time can increase unless the subproblem calculations can be done more quickly (Case C).

Note that without any decomposition and using the PIES algorithm for the original model, it takes only 0.045 seconds and nine PIES iterations to reach the equilibrium solution. Decomposition is not useful here to improve computational speed; its value is potentially for model management/maintenance.

In this illustration, note that both the subproblem and the master problem solutions are approximated in Case C (i.e., only one

PIES iteration is allowed for subproblem/master problem). Chung and Fuller [6] study approximations of the subproblem for DW decomposition of VI problems and this illustration combines their ideas with the approximation of the master problem. We did not provide any theorems for this algorithm and leave it for future research. However, we note that any real implementation necessarily has some degree of error in the solutions of the master problem and the subproblem. With approximate solutions proposed in this paper, one can control the amount of computational effort required at each iteration in order to decrease the overall computational burden.

For this illustration of approximation of the master problem, one team of analysts can manage/maintain the subproblem (e.g., regional models represented without linking constraints) and another team of analysts can manage/maintain the master problem (e.g., coordination problem with linking constraints enforced).

5. Conclusions and future research

In this paper, we present DW algorithms with approximation of the master problem and we test them on two models of energy markets, with comparisons to the standard DW algorithm without such approximations. The standard DW algorithm fails to converge within a reasonable time for some of our illustrations. However, the approximate DW algorithms converge to the equilibrium solution with some computational improvements over the standard DW algorithm in all cases.

Although the models without any decomposition can be solved considerably quicker than any DW algorithm, the benefits of managing the subproblem and master problem separately may compensate for the additional time to obtain a solution. For the TOU pricing models, separate teams of analysts can maintain the subproblem (containing only firms' problem) and the master problem (containing convex combinations of the proposals from subproblems and the ISO's problem with network constraints). With further approximation in the subproblems (as in Chung and Fuller [6] or Luna et al. [24]), models with special structure can be decomposed by other dimensions. As an example, energy market models can be decomposed by region (e.g., western and eastern Canada) or by commodity (e.g., electricity, gas, oil). Therefore, the resulting subproblems can be managed and maintained separately. Consistent solution of the whole model may be obtained by the proposed DW algorithms.

For some very large models, memory limits may cause problems (e.g., a stochastic model with many scenarios) and the only practical option may be the decomposition of the problem (see Gabriel and Fuller [17] and Egging [12] for an account).

Although the numerical results are presented for the VI problems, the theoretical results also hold for variety of problems, e.g., NLPs and nonlinear complementarity problems that satisfy the assumptions.

There are several avenues for future research on algorithms and computational efforts for large-scale VI problems. Theoretical work on convergence of algorithms which combine approximations of the mappings in both the subproblem and the master problem could be explored within the general framework laid out for subproblem approximation by Luna et al. [24]. Another possibility is to extend the ideas for approximate DW decomposition algorithms to the Benders decomposition for VI problems. A third possibility is presented by Goffin et al. [20] and Denault and Goffin [10,9] who introduce the analytic center cutting plane method (ACCPM) to solve VI problems. In the context of column generation and cutting planes, ACCPM is a centering concept from interior point methods. A direct application of this method to the algorithms in this paper is that, ACCPM can be used to compute "central prices" (e.g., central prices for congestion based wheeling) at each iteration of the DW decomposition. Instead of solving the master problem at each iteration, central prices can be used to compute a new proposal from the subproblem (e.g., a proposal provided by the ACCPM method). However, computing the analytic center can be computationally very challenging. Therefore, a proximal (i.e., approximate) analytic center can be useful within the Benders decomposition of VI problems in order to reduce the computational effort.

On the other hand, column generation methods (e.g., DW, simplicial decomposition) usually include some column dropping schemes that drops the columns that are no longer believed to be necessary in order to express an optimal solution [28,32,34,33,19,18]. The computational aim is to generate profitable columns in the search process of an optimal solution and, hence, to reduce the number of iterations and to increase efficiency of computations. In DW algorithm for VI problems, computational difficulties may arise when solving the master equilibrium problem, because the problem size grows with added columns. But using the background from optimization problems, these difficulties may sometimes be alleviated with a column dropping method.

Acknowledgments

Financial support for this work came from the Natural Sciences and Engineering Research Council of Canada and we are thankful to Dr. William Chung for providing GAMS models and data for the two-region energy equilibrium model for Canada.

Appendix A

We provide the theorems from Fuller and Chung [16] in this appendix. Proofs can be found in Fuller and Chung [16] and Chung and Fuller [6]. Statements and proofs of Theorems 3, 5 and 8 must be altered for the approximate DW algorithm, due to the master problem's use of the approximate mapping $\tilde{G}^k$; these are found in Section 3.2. All other theorems – numbered 1, 2, 4, 6, 7, 9 and 10 – do not rely at all on the approximate mapping $\tilde{G}^k$, and consequently, they hold true for the approximate DW algorithm, as readers may verify by examining the proofs in Fuller and Chung [16] and Chung and Fuller [6].

Theorem 1. x^* solves $VI(G,K)$ iff there exists $x^* \in R^n$, $\sigma^* \in R^m_+$ and $\omega^* \in R^l_+$ such that all the following conditions are satisfied:

$$G(x^*) - \nabla g(x^*)^T \sigma^* - \nabla h(x^*)^T \omega^* = 0 \tag{A.1}$$

$$g(x^*) \geq 0 \tag{A.2}$$

$$h(x^*) \geq 0 \tag{A.3}$$

$$\sigma^{*T} g(x^*) = 0 \tag{A.4}$$

$$\omega^{*T} h(x^*) = 0 \tag{A.5}$$

Theorem 2. x_S^k solves $\text{Sub-}VI^k(G - \nabla h(x_M^{k-1})^T \omega^{k-1}, \overline{K})$ iff there exists $x_S^k \in R^n$ and $\sigma_S^k \in R^m_+$ such that all the following conditions are satisfied:

$$G(x_S^k) - \nabla h(x_M^{k-1})^T \omega^{k-1} - \nabla g(x_S^k)^T \sigma_S^k = 0 \tag{A.6}$$

$$g(x_S^k) \geq 0 \tag{A.7}$$

$$\sigma_S^{kT} g(x_S^k) = 0 \tag{A.8}$$

Theorem 3. λ^k solves Master-$VI^k(H^k, \Lambda^k)$ iff there exist $\lambda^k \in R^k_+$, $\omega^k \in R^l_+$ and $\psi^k \in R$ such that all the following conditions are satisfied:

$$X^{kT} G(X^k \lambda^k) - X^{kT} \nabla h(X^k \lambda^k)^T \omega^k + e^k \psi^k \geq 0 \tag{A.9}$$

$$h(X^k\lambda^k)\geq 0 \tag{A.10}$$

$$e^{kT}\lambda^k = 1 \tag{A.11}$$

$$\lambda^{kT}(X^{kT}G(X^k\lambda^k) - X^{kT}\nabla h(X^k\lambda^k)^T\omega^k + e^k\psi^k) = 0 \tag{A.12}$$

$$\omega^{kT}h\left(X^k\lambda^k\right) = 0 \tag{A.13}$$

Theorem 4. If x_M^k solves Sub-$VI^{k+1}(G-\nabla h(x_M^k)^T\omega^k, \overline{K})$, then x_M^k solves $VI(G,K)$.

Theorem 5. If $CG^k < 0$, then $conv(X^k) \subset conv(X^{k+1})$ (strict inclusion).

Theorem 6. Assume that Assumption 5 is true (either G is strictly monotone, or $G(x) = \begin{pmatrix} -p(d) \\ \nabla c(z) \end{pmatrix}$ where $x = \begin{pmatrix} d \\ z \end{pmatrix}$, $-p(d)$ is strictly monotone and $c(z)$ is a convex function). If $CG^k \geq 0$, then x_M^k solves $VI(G,K)$.

Theorem 7. In the special case that $VI(G,K)$ is a LP, CG^k equals the difference between the value of the dual feasible solution provided by the subproblem and the value of the master problem.

Theorem 8. Assume that (a) G is continuous, and (b) any infinite subsequence of $\{(x_M^k, \omega^k, x_S^{k+1})\}_{k=1}^{\infty}$ has at least one limit point. Either $CG^k \geq 0$ at a finite iteration number k, or $CG^k < 0$ for all iterations k. In the latter case, $\lim_{k\to\infty} CG^k = 0$.

Theorem 9. If G is strictly monotone, then the solution to $VI(G,K)$ is unique. If the mapping $G(x) = \begin{pmatrix} -p(d) \\ \nabla c(z) \end{pmatrix}$ where $x = \begin{pmatrix} d \\ z \end{pmatrix}$, $-p(d)$ is strictly monotone and $c(z)$ is a convex function, then the solution is unique in d if $-p(d)$ is strictly monotone; and the solution is unique in d and z if $c(z)$ is strictly convex.

Theorem 10. If G is strongly monotone and continuous, then either $x_M^k = x_S^{k+1}$ for a finite iteration number k, or $\lim_{k\to\infty} |x_S^{k+1} - x_M^k| = 0$. If the mapping $G(x) = \begin{pmatrix} -p(d) \\ \nabla c(z) \end{pmatrix}$ where $x = \begin{pmatrix} d \\ z \end{pmatrix}$, $-p(d)$ is strongly monotone and continuous, and $c(z)$ is a convex function, then $\lim_{k\to\infty} |d_S^{k+1} - d_M^k| = 0$

Appendix B

In this appendix, we have provided the details about the models and DW algorithms used in Section 4.1 (see Çelebi [4] and Çelebi and Fuller [5] for further details).

Hobbs' [22] framework is used to determine TOU prices in Nash–Cournot game setting in electricity markets on a linearized DC network with line limits. In this setting, it is assumed that there are no power losses during transmission, and congestion is the basis for geographical differentiation in pricing. ISO is the owner of the grid and it operates the transmission system (not only the market operator, and not only ensuring that supply equals demand at every hour). ISO charges a congestion based fee (e.g., wheeling fee) for transmitting power from an arbitrary hub bus to any bus. But these fees are exogenous to its problem (i.e., adopting the Nash–Bertrand assumption that it cannot alter the fees it gets). Also, there are no arbitragers in their model and this allows non-cost based price differences to arise. Hence, suppliers can raise prices where competition is weak or demand is inelastic.

The supplier firms have their decision making process based on a bilateral market model. In this process, generation firms bilaterally contract with consumers to deliver electricity and generation firms pay the cost of transmitting power from the point of generation to the point of consumption. The schedule of injections and withdrawals by generation firms are then provided to the ISO, who collects the transmission fees from these firms for their use of the transmission network. The network constraints have already increased the complexity of the model, hence, we suppose only single period (e.g., a month). It is straightforward to include several periods. The model consists of four parts: the ISO's problem, supply side (e.g., firm f's problem), demand side and the market clearing conditions.

Symbols for the ISO's and supply side problems are defined in the following list. Symbols for the demand side are defined in "Demand Side" section of this appendix.

Sets

set of generation facilities: $i=1,\ldots,I$.
set of demand blocks: $j=1,\ldots,J$ (alias index k).
set of buses: $n=1,\ldots,N$ (N_d set of demand buses; N_g: set of generation buses).
set of hours in a demand block j: $h=1,\ldots,H_j$ (defined by the market regulator).
set of firms: $f=1,\ldots,F$.
set of lines: $l=1,..,L$.

Parameters

c_{fni} = operating cost per unit of energy for firm f's facility i at bus n ($/MWh).
κ_{fni} = capacity of firm f's facility i at bus n (MW).
δ_{njh} = fraction of total energy demand at bus n during block j of a month that occurs during hour h.
$PTDF_{ln}$ = power transfer distribution factors.[7]
T_{l-}, T_{l+} = lower and upper bounds on real power flows through line (interface) l (MW).

Decision variables

z_{fnijh} = the energy flowing from firm f's facility i to demand block j for hour h at generation bus n, $n \in N_g$ (MWh).
d_{fnj} = sales by firm f to demand block j at demand bus n, $n \in N_d$ (MWh).
p_{nj} = TOU prices (e.g., uniform block prices) for demand block j at bus n (off-peak, mid-peak and on-peak, $/MWh) (a function of d_{fnj} variables).
y_{njh} = net injections from transmission lines into bus n for demand block j at hour h (conceptually, power from hub bus to bus n) (MW).
β_{njh} = a congestion based fee (e.g., wheeling fee) for transmitting power from an arbitrary hub bus to bus n.

B.1. ISO's problem

In (B.1), the ISO (also the grid owner) maximizes its profit by allocating transmission capacity efficiently. It chooses y_{njh} by naively assuming that it is a price taker for transmission services (i.e., wheeling fees β_{njh}^* are exogenous in its problem). This is equivalent to a competitive market for transmission rights in which suppliers do not exercise market power [22]. In this market setting, congestion (wheeling) charges are sufficient to ration the use of the transmission network (i.e., transmission service is

[7] Power transfer distribution factor for bus n on line l ($PTDF_{ln}$) describes the per megawatt (MW) impact (e.g., increase or decrease) in flow resulting from 1 MW of power injection at hub bus and 1 MW of withdrawal at bus n. Summation of such impacts over all buses gives the total flow on line l.

allocated efficiently and it maximizes social welfare) [3].

$$\begin{aligned}
&\max_{y} \sum_{n=1}^{N}\sum_{j=1}^{J}\sum_{h=1}^{H_j} \beta^*_{njh} y_{njh} \\
&\text{subject to} \qquad\qquad\qquad\qquad [dual] \\
&-\sum_{n=1}^{N} PTDF_{\ln} y_{njh} \le T_{l-} \qquad \forall\ l,j,h \quad [\gamma_{ljh-}] \\
&\sum_{n=1}^{N} PTDF_{\ln} y_{njh} \le T_{l+} \qquad \forall\ l,j,h \quad [\gamma_{ljh+}] \\
&\sum_{n=1}^{N} y_{njh} = 0 \qquad\qquad \forall\ j,h \quad [\eta_{jh}]
\end{aligned} \tag{B.1}$$

where γ_{ljh-}, γ_{ljh+} are the dual variables of the negative and positive power flow through line l for demand block j at hour h ($/MW) and η_{jh} is the dual variable for flow balance equation. The model (B.1) differs from that of Hobbs [22] in the inclusion of hours h and demand blocks j.

Consistent with the linear DC approximation, flows through line l are modeled with PTDFs, which are derived based on Kirchhoff's current law (net flow into a bus equals zero) and Kirchhoff's voltage law (net voltage drop around any loop in the network is zero). The net MW flow through line l is $\sum_{n=1}^{N} PTDF_{\ln} y_{njh}$. The last constraint is the flow balance constraint at the hub bus. Note that y_{njh} variables are not restricted in sign. A positive (negative) y_{njh} means that there is a net flow into (out of) bus n from (to) hub bus. It is trivial to note that $y_{njh}=0$ is always a feasible solution to the ISO's problem, because T_{l-} and T_{l+} are positive scalars [22].

B.2. Supply side: firm f's problem

In the supply side of the model, formulated in (B.2), firm f computes its nodal sales and generation to maximize its profit, π_f, i.e., the total revenues of firm f minus the total operating cost of firm f's hourly generation by different technologies of production (e.g., nuclear, hydro, coal, gas/oil, indexed by i) to meet its sales in different demand blocks (e.g., off-peak, mid-peak, on-peak, indexed by j) minus the ISO's congestion (e.g., wheeling) fees.

$$\max_{d,z} \pi_f = \sum_{n\in N_d}\sum_{j=1}^{J}\left[p_{nj}(\cdot) - \sum_{h=1}^{H_j}\delta_{njh}\beta^*_{njh}\right] d_{fnj} - \sum_{n\in N_g}\sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{h=1}^{H_j}[c_{fni}-\beta^*_{njh}] z_{fnijh}$$

$$\begin{aligned}
&\text{subject to} \qquad\qquad\qquad [dual] \\
&\sum_{n\in N_d} d_{fnj} - \sum_{n\in N_g}\sum_{i=1}^{I}\sum_{h=1}^{H_j} z_{fnijh} \le 0 \qquad \forall\ j \quad [\rho_{fj}] \\
&z_{fnijh} \le \kappa_{fni} \qquad \forall\ n,i,j,h \quad [\mu_{fnijh}] \\
&z_{fnijh} \ge 0 \qquad \forall\ n,i,j,h
\end{aligned} \tag{B.2}$$

The first set of constraints ensures that electricity supply of firm f is sufficient to meet its sales to demand block j overall buses; at an optimal solution, these constraints are binding equalities. The second set of constraints contains the capacity constraint for each generation facility owned by firm f at each bus.

We have examined the Nash–Cournot structure where either all firms or some large firms act à la Cournot. In this structure, firms see their knowledge of the dependence of $p_{nj}(\cdot)$ on total market demand (i.e., in firm f's problem, p_{nj} is a function of $d_{fnj} + \sum_{f\neq g} d^*_{gnj}$, where other firms sales, d^*_{gnj}, are exogenous—denoted by superscript *). It is also presumed that the ISO's wheeling fees, β^*_{njh}, are exogenous to firm f's problem (i.e., all firms are "price taker" for transmission services) and yet endogenous in the overall equilibrium model (i.e., they will become endogenous in the MCP or VI formulation of the whole model).

The $[c_{fni}-\beta^*_{njh}]$ term in the objective function is denoting the per unit cost of the firm f to transmit power to the hub bus. The $[p_{nj}(\cdot)-\sum_{h=1}^{H_j}\delta_{njh}\beta^*_{njh}]$ term is denoting the per unit revenue of the firm f for conveying energy from hub bus to the sales bus for demand block j.[8] Note that d_{fnj} is a variable for demand block j (for several hours), and $\beta^*_{njh}$is an hourly wheeling fee. δ_{njh} parameters are the connection between these different time scales (e.g., d_{fnj}: sales to demand blocks j at demand bus n; z_{fnijh} :hourly generation output at generation bus n). Wheeling fee for sales in demand block j at demand bus n is represented for several hours, i.e., $\sum_{h=1}^{H_j}\delta_{njh}\beta^*_{njh}d_{fnj}$.

B.3. Demand side

The demand side is represented by demand equations that use only the prices as independent variables. $\boldsymbol{D}_n^{(0)}$ is the lagged demand term and it is a parameter in the single period model. A multi-commodity case where each commodity is the electricity demand in different times of day (e.g., demand blocks: on-peak, mid-peak, off-peak) at each bus n

$$\boldsymbol{D}_n = \boldsymbol{A}_n + \boldsymbol{B}_n\boldsymbol{P}_n + \boldsymbol{E}_n\boldsymbol{D}_n^{(0)} \tag{B.3}$$

where

$\boldsymbol{A}_n$ = vector of the factors representing non-price effects at bus n.
$\boldsymbol{D}_n$ = vector of all demands for electricity at bus n (i.e., on-peak, mid-peak, off-peak demand at bus n) where $\boldsymbol{D}_n = [d_{nj}]$ and $d_{nj} = \sum_{f=1}^{F} d_{fnj}$.
$\boldsymbol{D}_n^{(0)}$ = vector of all lagged demands for electricity at bus n.
$\boldsymbol{P}_n$ = vector of TOU electricity prices at bus n (i.e., on-peak, mid-peak, off-peak prices).
$\boldsymbol{B}_n$ = a square matrix of the price coefficients (i.e., own-price and cross-price) for bus n.
$\boldsymbol{E}_n$ = a square diagonal matrix of the lag coefficients for bus n.

$\boldsymbol{B}_n$ is assumed to be invertible for our analyses (i.e., inverse demand functions are well defined).

B.4. Market clearing conditions

The total transmission service demanded by generators and consumers from the hub to any bus n, demand block j, hour h must equal the transmission service the grid provides between these buses

$$\delta_{njh}\sum_{f=1}^{F} d_{fnj} - \sum_{f=1}^{F}\sum_{i=1}^{I} z_{fnijh} = y_{njh} \quad \forall\ n,j,h\ [dual] : [\beta_{njh}] \tag{B.4}$$

where β_{njh} is the dual variable for the market clearing condition (i.e., wheeling fees that clear the markets). Note that total generation by all firms for hour h at bus n in demand block j is a fraction (δ_{njh}) of total energy sales by all firms to demand block j at bus n.

[8] A firm pays $-\beta^*_{njh}$ to transmit power to the hub bus from a generator at bus n and it pays $\sum_{h=1}^{H_j}\delta_{njh}\beta^*_{n'jh}$ to convey power to sales bus n' from hub bus. Generation is charged straightforwardly at the hourly wheeling fee, but consumption is more complicated because it is measured only in a block of hours, so it is charged at the weighted average of the hourly wheeling fees for the hours within the demand block j.

Also note that in the special case of $\delta_{njh}=\delta_{jh}$ at all buses, we can add up (B.4) over all buses to derive (together with the last constraint of (B.1))

$$\delta_{jh}\sum_{n\in N_d}\sum_{f=1}^{F}d_{fnj}-\sum_{n\in N_g}\sum_{f=1}^{F}\sum_{i=1}^{I}z_{fnijh}=0 \quad \forall j,h. \tag{B.5}$$

This set of constraints states that the hourly generation at all generation buses for all different facilities and all firms should meet the total sales of all firms overall demand buses for every hour h. With this condition, the ISO imposes the historical shape of the load duration curve within the hours of demand block j over all buses and β_{njh} can be additionally interpreted as a penalty/payment for deviations from the historical shape of the load duration curve.

Notice that demand variations over hours within the demand block j at each bus n is modeled with δ_{njh} parameters (when enough data are available for each bus). This pattern of variation in demand at bus n, δ_{njh}, is imposed on total sales of all firms, not individually for each firm. A firm may produce more or less than $\delta_{njh}\sum_{n\in N_d}d_{fnj}$ in an hour, but its total production for demand block j must meet its sales in demand block j, $\sum_{n\in N_d}d_{fnj}$.

B.5. MCP and VI Formulations for the TOU pricing models on a linearized DC network

Firstly, we formulate the perfect competition model as a MCP, by writing out the necessary KKT conditions for the ISO's and firm f's problems along with the demand equation and the market clearing conditions

MCP: find $d_{fnj}, z_{fnijh}, p_{nj}, \rho_{fj}, \mu_{fnijh}, y_{njh}, \gamma_{ljh-}, \gamma_{ljh+}, \eta_{jh}, \beta_{njh}$ that satisfy

$$\begin{aligned}
&d_{fnj}\geq 0\perp && -p_{nj}+\sum_{h=1}^{H_j}\delta_{njh}\beta_{njh}+\rho_{fj}\geq 0 && \forall f,n,j\\
&z_{fnijh}\geq 0\perp && c_{fni}-\rho_{fj}+\mu_{fnijh}-\beta_{njh}\geq 0 && \forall f,n,i,j,h\\
&\rho_{fj}\geq 0\perp && \sum_{n\in N_d}d_{fnj}-\sum_{n\in N_d}\sum_{i=1}^{I}\sum_{h=1}^{H_j}z_{fnijh}\leq 0 && \forall f,j\\
&\mu_{fnijh}\geq 0\perp && z_{fnijh}\leq \kappa_{fni} && \forall f,n,i,j,h\\
&\mu_{fnijh}\geq 0\perp && -\beta_{njh}+\sum_{l=1}^{L}PTDF_{ln}(\gamma_{ljh+}-\gamma_{ljh-})+\eta_{jh}=0 && \forall n,j,h\\
&\gamma_{ljh-}\geq 0\perp && -\sum_{n=1}^{N}PTDF_{ln}y_{njh}\leq T_{l-} && \forall l,j,h\\
&\gamma_{ljh+}\geq 0\perp && \sum_{n=1}^{N}PTDF_{ln}y_{njh}\leq T_{l+} && \forall l,j,h\\
&\eta_{jh}\ free\perp && \sum_{n=1}^{N}y_{njh}=0 && \forall j,h\\
&\beta_{njh}\ free\perp && \delta_{njh}\sum_{f=1}^{F}d_{fnj}-\sum_{f=1}^{F}\sum_{i=1}^{I}z_{fnijh}=y_{njh} && \forall n,j,h\\
& && \sum_{f=1}^{F}d_{fnj}=a_{nj}+\sum_{k=1}^{K}b_{njk}p_{nk}+e_{njj}\sum_{f=1}^{F}d_{fnj}^{(0)} && \forall n,j
\end{aligned} \tag{B.6}$$

where a_{nj} and p_{nj} are the jth elements of vectors $\boldsymbol{A}_n$ and $\boldsymbol{P}_n$, respectively. Similarly, b_{njk} and e_{njj} are the elements of matrices $\boldsymbol{B}_n$ and $\boldsymbol{E}_n$, respectively. Note that p_{nj} variables are implicitly defined by the d_{fnj} variables. Instead, an explicit inverse demand function can be used for a more compact formulation without p_{nj} variables, but for ease of readability of the formulation, the p_{nj} variables are used.

We can also formulate (B.6) as a VI problem. The feasible set for the VI problem is defined as follows:

$$K=\left\{d_{fnj},z_{fnijh},p_{nj},y_{njh}\ \middle|\ \begin{aligned}
&\sum_{n\in N_d}d_{fnj}-\sum_{n\in N_d}\sum_{i=1}^{I}\sum_{h=1}^{H_j}z_{fnijh}\leq 0 && \forall f,j\\
&z_{fnijh}\leq\kappa_{fni} && \forall f,n,i,j,h\\
&z_{fnijh}\geq 0 && \forall f,n,i,j,h\\
&-\sum_{n=1}^{N}PTDF_{ln}y_{njh}\leq T_{l-} && \forall l,j,h\\
&\sum_{n=1}^{N}PTDF_{ln}y_{njh}\leq T_{l+} && \forall l,j,h\\
&\sum_{n=1}^{N}y_{njh}=0 && \forall j,h\\
&\delta_{njh}\sum_{f=1}^{F}d_{fnj}-\sum_{f=1}^{F}\sum_{i=1}^{I}z_{fnijh}=y_{njh} && \forall n,j,h\\
&\sum_{f=1}^{F}d_{fnj}=a_{nj}+\sum_{k=1}^{K}b_{njk}p_{nk}+e_{njj}\sum_{f=1}^{F}d_{fnj}^{(0)} && \forall n,j
\end{aligned}\right\}$$

In the feasible set K, the first six constraints are from the firm f's and ISO's problems and the last two equations are the market clearing condition and the linear distributed lagged demand equation, respectively.

The VI problem for the perfect competition model is as in (B.7). Note that for d_{fnj}, z_{fnijh} and y_{njh} the corresponding elements of G are the partial derivatives of the objective functions of firm f's problem (B.2) and the ISO's problem (B.1)

Find $(d^*_{fnj},z^*_{fnijh},p^*_{nj},y^*_{njh})\in K$ such that

$$\begin{aligned}
&-\sum_{f=1}^{F}\sum_{n\in N_d}\sum_{j=1}^{J}\left(p^*_{nj}-\sum_{h=1}^{H_j}\delta_{njh}\beta_{njh}\right)(d_{fnj}-d^*_{fnj})\\
&+\sum_{f=1}^{F}\sum_{n\in N_g}\sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{h=1}^{H_j}(c_{fni}-\beta_{njh})(z_{fnijh}-z^*_{fnijh})\\
&-\sum_{n=1}^{N}\sum_{j=1}^{J}\sum_{h=1}^{H_j}\beta_{njh}(y_{njh}-y^*_{njh})\geq 0\\
&\forall (d_{fnj},z_{fnijh},p_{nj},y_{njh})\in K
\end{aligned} \tag{B.7}$$

Also note that the β_{njh} terms are canceled out in (B.7) (i.e., due to market clearing condition), and a more compact form is derived as follows:

Find $(d^*_{fnj},z^*_{fnijh},p^*_{nj})\in K$ such that

$$\begin{aligned}
&-\sum_{f=1}^{F}\sum_{n\in N_d}\sum_{j=1}^{J}p^*_{nj}(d_{fnj}-d^*_{fnj})\\
&+\sum_{f=1}^{F}\sum_{n\in N_g}\sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{h=1}^{H_j}c_{fni}(z_{fnijh}-z^*_{fnijh})\geq 0\\
&\forall (d_{fnj},z_{fnijh},p_{nj})\in K
\end{aligned} \tag{B.8}$$

The VI problem (B.8) in primal variables has the KKT conditions listed in (B.6) and hence is equivalent to the MCP (B.6).

We only provide the VI formulation for Nash–Cournot market structure, for ease of representation. It is straightforward to derive its equivalent MCP formulations as in the perfect competition case. The Nash–Cournot market structure has the same feasible set K. The VI problem for the Nash–Cournot model is formulated as in the following equation:

Find $(d^*_{fnj},z^*_{fnijh},p^*_{nj})\in K$ such that

$$\begin{aligned}
&-\sum_{f=1}^{F}\sum_{n\in N_d}\sum_{j=1}^{J}(p^*_{nj}+\theta^*_{fnj})(d_{fnj}-d^*_{fnj})\\
&+\sum_{f=1}^{F}\sum_{n\in N_g}\sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{h=1}^{H_j}c_{fni}(z_{fnijh}-z^*_{fnijh})\geq 0\\
&\forall (d_{fnj},z_{fnijh},p_{nj})\in K
\end{aligned} \tag{B.9}$$

where the term $p^*_{nj}+\theta^*_{fnj}$ is the marginal revenue for firm f at bus n and demand block j, and θ^*_{fnj} is the "extra" marginal revenue term. This marginal revenue term is derived from the partial derivative of

the objective function in (B.2), with respect to d_{fnj}, when the firm is aware of the price–quantity relation of the distributed lagged demand equation

$$\frac{\partial \pi_f}{\partial d_{fnj}} = p_{nj} + \frac{\partial p_{nj}}{\partial d_{fnj}} d_{fnj} = p_{nj} + \theta_{fnj} \tag{B.10}$$

where $[\partial p_{nj}/\partial d_{fnj}] = \boldsymbol{B}_n^{-1}$.

Note that the congestion fees charged by the ISO are neither included in the VI formulation (B.9) nor in the marginal revenue term (B.10), because they are canceled out in the overall formulation.

B.6. Standard DW algorithm

In this part of the Appendix B, we relate the standard DW algorithm to the VI formulation of perfect competition TOU pricing models in Chapter 3 of Çelebi [4]. For brevity, Nash–Cournot extension is not provided in this appendix, but it is a straightforward extension of perfect competition TOU pricing models. To relate general VI form (B.7) to the subproblem (3) and master problem (6), we have

$$x = \begin{bmatrix} d_{fnj} \\ z_{fnijh} \\ y_{njh} \\ p_{nj} \end{bmatrix} \ \forall f,n,i,j,h;\ G(x) = \begin{bmatrix} -p_{nj} \\ c_{fni} \\ 0 \\ 0 \end{bmatrix} \ \forall f,n,i;\ \omega^k = [\beta^k_{njh}] \quad \forall\ n,j,h;$$

$$-\nabla h(x_M^k)^T \omega^k = \begin{bmatrix} \sum_{h=1}^{H_j} \delta_{njh}\beta^k_{njh} \\ -\beta^k_{njh} \\ -\beta^k_{njh} \\ 0 \end{bmatrix} \quad \forall\ n,j,h;$$

$$X^k = \begin{bmatrix} D^k_{fnj} \\ Z^k_{fnijh} \\ Y^k_{njh} \\ P^k_{nj} \end{bmatrix} = \begin{bmatrix} d^1_{fnj}, d^2_{fnj}, \ldots, d^k_{fnj} \\ z^1_{fnijh}, z^2_{fnijh}, \ldots, z^k_{fnijh} \\ y^1_{njh}, y^2_{njh}, \ldots, y^k_{njh} \\ p^1_{nj}, p^2_{nj}, \ldots, p^k_{nj} \end{bmatrix} \quad \forall f,n,i,j,h.$$

The vector x contains the variables d_{fnj}, z_{fnijh}, y_{njh} and p_{nj} for all f, n, i, j and h, and the elements of the vector-valued mapping $G(x)$ are as follows: $-p_{nj}$ is the element of G that corresponds to d_{fnj}; c_{fni} is the element of G that corresponds to z_{fnijh}; and the elements of G that corresponds to y_{njh} and p_{nj} are zero. Note that for d_{fnj} and z_{fnijh}, the corresponding elements of G are the partial derivatives of the objective function of firm f's problem (B.2), but the terms involving the dual variables β^k_{njh} are left out. Because, these dual terms β^k_{njh} are canceled out in the original VI formulation and therefore, they do not appear in the mapping G. The feasible set for the subproblem and the master problem are as follows:

$$\overline{K} = \left\{ d_{fnj}, z_{fnijh}, p_{nj}, y_{njh} \;\middle|\; \begin{array}{ll} \sum_{n\in N_d} d_{fnj} - \sum_{n\in N_g} \sum_{i=1}^{I} \sum_{h=1}^{H_j} z_{fnijh} \le 0 & \forall\ f,j \\ z_{fnijh} \le \kappa_{fni} & \forall\ f,n,i,j,h \\ z_{fnijh} \ge 0 & \forall\ f,n,i,j,h \\ -\sum_{n=1}^{N} PTDF_{ln} y_{njh} \le T_{l-} & \forall\ l,j,h \\ \sum_{n=1}^{N} PTDF_{ln} y_{njh} \le T_{l+} & \forall\ l,j,h \\ \sum_{n=1}^{N} y_{njh} = 0 & \forall\ j,h \\ \sum_{f=1}^{F} d_{fnj} = a_{nj} + \sum_{k=1}^{K} b_{njk} p_{nk} + e_{nj} \sum_{f=1}^{F} d^{(0)}_{fnj} & \forall\ n,j \end{array} \right\}$$

$$K^k = \{ d_{fnj}, z_{fnijh}, p_{nj}, y_{njh} \in conv(X^k) |\ \delta_{njh} \sum_{f=1}^{F} d_{fnj} - \sum_{f=1}^{F} \sum_{i=1}^{I} z_{fnijh} = y_{njh} \quad \forall\ n,j,h \}$$

B.7. Modified DW algorithm

For the 'modified' DW decomposition algorithm, we first present the relation of the mapping and variables for the subproblem to the general form of subproblem in (3) as follows:

$$x = \begin{bmatrix} d_{fnj} \\ z_{fnijh} \\ p_{nj} \end{bmatrix} \quad \forall f,n,i,j,h; \quad G(x) = \begin{bmatrix} -p_{nj} \\ c_{fni} \\ 0 \end{bmatrix} \quad \forall f,n,i; \quad \omega^k = [\beta^k_{njh}] \quad \forall\ n,j,h;$$

$$-\nabla h(x_M^k)^T \omega^k = \begin{bmatrix} \sum_{h=1}^{H_j} \delta_{njh}\beta^k_{njh} \\ -\beta^k_{njh} \\ 0 \end{bmatrix} \quad \forall\ n,j,h.$$

The feasible set for the subproblem is as follows:

$$\overline{K} = \left\{ d_{fnj}, z_{fnijh}, p_{nj} \;\middle|\; \begin{array}{ll} \sum_{n\in N_d} d_{fnj} - \sum_{n\in N_g} \sum_{i=1}^{I} \sum_{h=1}^{H_j} z_{fnijh} \le 0 & \forall\ f,j \\ z_{fnijh} \le \kappa_{fni} & \forall\ f,n,i,j,h \\ z_{fnijh} \ge 0 & \forall\ f,n,i,j,h \\ \delta_{jh} \sum_{n\in N_d} \sum_{f=1}^{F} d_{fnj} - \sum_{n\in N_g} \sum_{f=1}^{F} \sum_{i=1}^{I} z_{fnijh} = 0 & \forall\ j,h \\ \sum_{f=1}^{F} d_{fnj} = a_{nj} + \sum_{k=1}^{K} b_{njk} p_{nk} + e_{njj} \sum_{f=1}^{F} d^{(0)}_{fnj} & \forall\ n,j \end{array} \right\}$$

In this subproblem, the vector x contains the variables d_{fnj}, z_{fnijh} and p_{nj} for all f, n, i, j and h, and the elements of the vector-valued mapping $G(x)$ are as follows: $-p_{nj}$ is the element of G that corresponds to d_{fnj}; c_{fni} is the element of G that corresponds to z_{fnijh}; and the element of G that corresponds to p_{nj} is zero. Note that the feasible set for the subproblem does not include any constraints of the ISO's problem. These constraints are included in the master problem. Also the subproblem does not include any y_{njh} variables in the VI formulation (i.e., the subproblem has no variables for the linearized DC network constraints). Moreover, to enforce feasibility for the master problem and to produce better proposals from subproblems, we add extra constraints to the subproblem's feasible set $\overline{K}$, the fourth set of constraints. These extra constraints are obtained by summing the complicating constraints (market clearing conditions) over all buses.

For the 'modified' DW decomposition algorithm, we relate the mapping, variables and set for the master problem to the general form of the master problem in (6) as follows:

$$x = \begin{bmatrix} d_{fnj} \\ z_{fnijh} \\ y_{njh} \\ p_{nj} \end{bmatrix} \quad \forall \quad f,n,i,j,h; \quad G(x) = \begin{bmatrix} -p_{nj} \\ c_{fni} \\ 0 \\ 0 \end{bmatrix} \quad \forall\ f,\ n,\ i;$$

$$X^k = \begin{bmatrix} D^k_{fnj} \\ Z^k_{fnijh} \\ P^k_{nj} \end{bmatrix} = \begin{bmatrix} d^1_{fnj}, d^2_{fnj}, \ldots, d^k_{fnj} \\ z^1_{fnijh}, z^2_{fnijh}, \ldots, z^k_{fnijh} \\ p^1_{nj}, p^2_{nj}, \ldots, p^k_{nj} \end{bmatrix} \quad \forall\ f,\ n,\ i,\ j,\ h.$$

The feasible set for the master problem is as follows:

$$K^k = \left\{ d_{fnj}, z_{fnijh}, p_{nj} \in conv(X^k), y_{njh} \;\middle|\; \begin{array}{ll} -\sum_{n=1}^{N} PTDF_{ln} y_{njh} \le T_{l-} & \forall\, l,j,h \\ \sum_{n=1}^{N} PTDF_{ln} y_{njh} \le T_{l+} & \forall\, l,j,h \\ \sum_{n=1}^{N} y_{njh} = 0 & \forall\, j,h \\ \delta_{njh} \sum_{f=1}^{F} d_{fnj} - \sum_{f=1i=1}^{F} \sum_{i=1}^{I} z_{fnijh} = y_{njh} & \forall\ n,j,h \end{array} \right\}$$

For the master problem, the vector x contains the variables d_{fnj}, z_{fnijh}, y_{njh} and p_{nj} for all f, n, i, j and h, and the elements of the vector-valued mapping $G(x)$ are as follows: $-p_{nj}$ is the element of G that corresponds to d_{fnj}; c_{fni} is the element of G that corresponds to z_{fnijh}; and the element of G that corresponds to p_{nj} and y_{njh} are zero. Note that, in the feasible set of the master problem d_{fnj}, z_{fnijh}, $p_{nj} \in conv(X^k)$ and y_{njh} are the variables of the master problem only (e.g., subproblem only provides proposals for d_{fnj}, z_{fnijh}, p_{nj} variables).

B.8. Approximate DW algorithms

In this part of Appendix B, we present the approximation of the master problem by different mappings $\tilde{G}^k(x)$ and relation of the mappings and variables for the subproblem and the approximate master problem to the general form of subproblem in (3) and approximate master problem in (8).

Instead of using the asymmetric inverse demand function, $p_{nj}(d)$ (where $d = [d_{fnj}]$), we define a symmetric inverse demand function, $\tilde{p}_{nj}(d;\varphi^k)$, where φ^k equals the d variables from the most recent solution of the master problem, d_M^{k-1}

$$\tilde{p}_{nj}(d;\varphi^k) = p_{nj}(\varphi^k) + \tilde{B}^{-1}(d-\varphi^k) = p_{nj}(d) + (\tilde{B}^{-1} - B^{-1})(d-\varphi^k),$$

where $\tilde{B} = (1/2)(B + B^T)$. We denote this algorithm as 'approximate'.

Another approximation in the master problem of the DW algorithm is to apply the PIES algorithm to the master problem, i.e., $\tilde{B} = Diag(B)$. This approximation allows for a diagonal inverse demand function, $\tilde{p}_{nj}(d;\varphi^k)$, (e.g., with cross-demand variables fixed at the φ^k). We call this approximation, 'approximate-PIES'. The difference between these two approximations is the difference between symmetrization (approximate) and diagonalization (approximate-PIES) of the mapping $G(x)$.

To relate TOU pricing models to the approximation of the master problem in (8), we have

$$x = \begin{bmatrix} d_{fnj} \\ z_{fnijh} \\ y_{njh} \\ p_{nj} \end{bmatrix} \quad \forall\, f,n,i,j,h; \qquad \tilde{G}^k(x) = \begin{bmatrix} -\tilde{p}_{nj}(d;\varphi^k) \\ c_{fni} \\ 0 \\ 0 \end{bmatrix} \quad \forall\ f,n,i;$$

$$X^k = \begin{bmatrix} D^k_{fnj} \\ Z^k_{fnijh} \\ P^k_{nj} \end{bmatrix} = \begin{bmatrix} d^1_{fnj}, d^2_{fnj}, \ldots, d^k_{fnj} \\ z^1_{fnijh}, z^2_{fnijh}, \ldots, z^k_{fnijh} \\ p^1_{nj}, p^2_{nj}, \ldots, p^k_{nj} \end{bmatrix} \quad \forall\ f,n,i,j,h.$$

The subproblem in (3), however, uses the exact mapping $G(x) = \begin{bmatrix} -p_{nj} \\ c_{fni} \\ 0 \end{bmatrix}$ as in the 'modified' DW algorithm. The feasible set for the master problem and subproblem for the approximate DW algorithms are same as the 'modified' DW algorithm.

References

[1] Ahn B, Hogan WW. On convergence of the PIES algorithm for computing equilibria. Operations Research 1982;30(2):281–300.
[2] Bertsekas DP, Tsitsiklis JN. Parallel and distributed computation: numerical methods. New Jersey: Prentice Hall; 1989.
[3] Boucher J, Smeers Y. Alternative models of restructured electricity systems, part I: no market power. Operations Research 2001;49(6):821–38.
[4] Çelebi, E. Decomposition of variational inequalities with applications to Nash–Cournot models in time of use electricity markets. PhD thesis, University of Waterloo; 2011.
[5] Çelebi E, Fuller JD. Time-of-use pricing in electricity markets under different market structures. IEEE Transactions on Power Systems 2012;27 (3):1170–81.
[6] Chung W, Fuller JD. Subproblem approximation in Dantzig–Wolfe decomposition of variational inequality models with an application to a multicommodity economic equilibrium model. Operations Research 2010;58(5):1318–27.
[7] Chung W, Fuller JD, Wu Y. A new decomposition method for multiregional economic equilibrium models. Operations Research 2006;54(4):643–55.
[8] Conejo AJ, Castillo E, Minguez R, Garcia-Bertrand R. Decomposition techniques in mathematical programming: engineering and science applications. Heidelberg: Springer; 2006.
[9] Denault M, Goffin J. The analytic-center cutting-plane method for variational inequalities: a quadratic-cut approach. INFORMS Journal on Computing 2005;17(2):192–206.
[10] Denault M, Goffin JL. On a primal-dual analytic center cutting plane method for variational inequalities. Computational Optimization and Applications 1999;12 (1):127–55.
[11] Dirkse SP, Ferris MC. A pathsearch damped newton method for computing general equilibria. Annals of Operations Research 1996;68(2):211–32.
[12] Egging R. Benders decomposition for multi-stage stochastic mixed complementarity problems—applied to a global natural gas market model. European Journal of Operational Research 2013;226(2):341–53.
[13] Ferris MC, Dirkse SP, Jagla JH, Meeraus A. An extended mathematical programming framework. Computers & Chemical Engineering 2009;33 (12):1973–82.
[14] Ferris MC, Mangasarian OL, Pang JS. Complementarity: applications, algorithms and extensions. Dordrecht: Kluwer Academic Pub.; 2001.
[15] Fuller JD, Chung W. Benders decomposition for a class of variational inequalities. European Journal of Operational Research 2008;185(1):76–91.
[16] Fuller JD, Chung W. Dantzig–Wolfe decomposition of variational inequalities. Computational Economics 2005;25(4):303–26.
[17] Gabriel SA, Fuller JD. A Benders decomposition method for solving stochastic complementarity problems with an application in energy. Computational Economics 2010;35:301–29.
[18] Garcia-Rodenas R, Marin A, Patriksson M. Column generation algorithms for nonlinear optimization, II: numerical investigations. Computers and Operations Research, 38; 591–604.
[19] Garcia R, Marin A, Patriksson M. Column generation algorithms for nonlinear optimization, I: convergence analysis. Optimization 2003;52 (2):171–200.
[20] Goffin JL, Marcotte P, Zhu D. An analytic center cutting plane method for pseudomonotone variational inequalities. Operations Research Letters 1997;20 (1):1–6.
[21] Harker PT, Pang JS. Finite-dimensional variational inequality and nonlinear complementarity problems: a survey of theory, algorithms and applications. Mathematical Programming 1990;48(1):161–220.
[22] Hobbs BF. Linear complementarity models of Nash–Cournot competition in bilateral and POOLCO power markets. IEEE Transactions on Power Systems 2001;16(2):194–202.
[23] Hogan WW. Energy policy models for project independence. Computers & Operations Research 1975;2(3–4):251–71.
[24] Luna J, Sagastizábal C, Solodov M. A class of Dantzig–Wolfe type decomposition methods for variational inequality problems. Mathematical Programming 2012:1–33, http://dx.doi.org/10.1007/s10107-012-0599-7.
[25] Manne AS. On the formulation and solution of economic equilibrium models. Economic Equilibrium: Model Formulation and Solution 1985:1–22.
[26] Mathiesen L. Computation of economic equilibria by a sequence of linear complementarity problems. Economic Equilibrium: Model Formulation and Solution 1985:144–62.
[27] Murphy FH. Making large-scale models manageable: modeling from an operations management perspective. Operations Research 1993;41(2):241–52.
[28] Murphy FH. Column dropping procedures for the generalized programming algorithm. Management Science 1973;19(11):1310–21.
[29] Murphy FH, Mudrageda MV. A decomposition approach for a class of economic equilibrium models. Operations Research 1998;46(3):368–77.
[30] Murphy FH, Conti JJ, Shaw SH, Sanders R. Modeling and forecasting energy markets with the intermediate future forecasting system. Operations Research 1988(3):36–420.
[31] Nagurney A. Network economics: a variational inequality approach. Dordrecht: Kluwer Academic Pub.; 1993.
[32] O'Neill RP. Technical note—column dropping in the Dantzig–Wolfe convex programming algorithm: computational experience. Operations Research 1977;25(1):148–55.

[33] Patriksson M. Nonlinear programming and variational inequality problems: a unified approach, vol. 23. Dordrecht: Kluwer Academic Pub.; 1999.
[34] Patriksson M. The traffic assignment problem: models and methods. Utrecht: VSP Intl Science; 1994.
[35] Samuelson PA. Spatial price equilibrium and linear programming. The American Economic Review 1952;42(3):283–303.
[36] Takayama T, Judge GG. Spatial and temporal price and allocation models. Amsterdam: North-Holland Pub. Co; 1971.
[37] Wu Y, Fuller JD. Introduction of geometric, distributed lag demand into energy-process models. Energy 1995;20(7):647–56.