Electrical Power and Energy Systems 33 (2011) 55–60

Contents lists available at ScienceDirect

Electrical Power and Energy Systems

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

Decomposition–coordination interior point method and its application to multi-area optimal reactive power flow

Wei Yan [a,*], Lili Wen [b], W. Li [c], C.Y. Chung [a,d], K.P. Wong [d]

[a] State Key Laboratory of Power Transmission Equipment & System Security and New Technology, College of Electrical Engineering, Chongqing University, Chongqing 400030, China
[b] Test & Research Institute of Chongqing Electric Power, Chongqing 401123, China
[c] British Columbia Transmission Corporation, Vancouver, BC, Canada V7X 1V5
[d] Department of Electrical Engineering, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong

ARTICLE INFO

Article history:
Received 17 February 2009
Received in revised form 3 March 2010
Accepted 13 August 2010

Keywords:
Distributed calculations
Interior point method
Multi-area problems
Optimal reactive power flow
Power system decomposition

ABSTRACT

A decomposition–coordination interior point method (DIPM) is presented and applied to the multi-area optimal reactive power flow (ORPF) problem in this paper. In the method, the area distributed ORPF problem is first formed by introducing duplicated border variables. Then the nonlinear primal dual interior point method (IPM) is directly applied to the distributed ORPF problem in which a Newton system with border-matrix-blocks is formulated. Finally the overall ORPF problem is solved in decomposition iterations with the Newton system being decoupled. The proposed DIPM inherits the good performance of the traditional IPM with a feature appropriate for distributed calculations among multiple areas. It can be easily extended to other distributed optimization problems of power systems. Numeric results of five IEEE Test Systems are demonstrated and comparisons are made with those obtained using the traditional auxiliary problem principle (APP) method. The results show that the DIPM for the multi-area OPRF problem requires less iterations and CPU time, has better stability in convergence, and reaches better optimality compared to the traditional auxiliary problem principle method.

1. Introduction

A modern power system is an interconnected grid and contains widely distributed sub-networks each of which represents an independent utility. Generally, each utility company has its own operation criteria and separate EMS. Local optimal simulations in the individual EMSs will produce great errors. Therefore, there is a need of coordinating solutions of the individual EMSs to achieve the overall optimal solution. Decomposition techniques in a distributed computing environment have attracted great attention.

Many different decomposition techniques have been proposed during the past forty years, such as Dantzig-Wolfe technique [1,2], Lagrangian relaxation technique [3], augmented Lagrangian technique [4–7], and approximate Newton directions [9,10]. Particularly remarkable is the theoretical decomposition framework based on the auxiliary problem principle (APP). It has been used widely in solving many operation problems of power systems, such as the daily generation scheduling [4], distributed state estimation [5], multi-area optimal power flow [6,7], and multi-area ORPF [8]. But the APP method presents only modest speed-ups and efficiencies even in ideal situations [6], and sometimes leads to poor convergence when related parameters are selected improperly [10].

This paper proposes a new decomposition method based on the nonlinear primal dual interior point method (IPM) [11–15]. Similar to the APP method, the proposed method also uses the concept of duplicated border variables to implement an area decomposition of the original overall problem. Differently, in the proposed method, the objective function, variables and constraints are structured in such a way that a Newton system with border-matrix-blocks is created. This Newton system is then decoupled into a set of linear subsystems so that the overall problem can be solved through separate computations of linear area subsystems in the Newton iteration process of IPM. The proposed method only computes the linear subsystems corresponding to individual areas and avoids resolutions of optimization sub-problems with respective iteration processes. Thus, the proposed method has an advantage in computational efficiency superior to the APP and other Lagrangian relaxation or augmented Lagrangian decomposition algorithms while it inherits good performance of the IPM in fast convergence. This is because the proposed method only needs the decoupling implementation of the Newton system in the IPM.

The proposed decomposition–coordination interior point method (DIPM) is applied to solve the multi-area optimal reactive power flow (ORPF) in the paper. It should be pointed out that it can also be extended to other distributed computing problems of power sys-

* Corresponding author.
E-mail addresses: cquyanwei@cqu.edu.cn (W. Yan), wll563003@sina.com (L. Wen), wen.yuan.li@bctc.com (W. Li), eecychun@polyu.edu.hk (C.Y. Chung), eekpwong@polyu.edu.hk (K.P. Wong).

0142-0615/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ijepes.2010.08.004

tems. The rest of the paper is organized as follows. Section 2 provides the formulation of area decentralization model of ORPF. Section 3 describes the proposed decomposition method. In Section 4, numerical examples are given to demonstrate the effectiveness of the proposed method, followed by Section 5 for conclusions.

2. Area decentralization model of ORPF

2.1. Centralized ORPF formulation

The formulation of the original ORPF problem can be mathematically expressed as the following minimization nonlinear programming problem:

$$\begin{aligned} \min \quad & f(\boldsymbol{x}) \\ s.t. \quad & \boldsymbol{g}\boldsymbol{x} = 0 \\ & \underline{\boldsymbol{x}} \leqslant \boldsymbol{x} \leqslant \bar{\boldsymbol{x}} \end{aligned} \tag{1}$$

where the objective function $f(\boldsymbol{x})$ is total active power losses; $\boldsymbol{g}(\boldsymbol{x})$ is the nonlinear vector function representing power flow equations; $\boldsymbol{x}$ is the vector of state and control variables, including voltage magnitudes and angles at load buses, injected reactive powers of generators, voltage magnitudes at generator buses, reactive powers of shunt capacitors/reactors and transformer tap ratios; $\bar{\boldsymbol{x}}$ and $\underline{\boldsymbol{x}}$ are the vectors representing operational limits on state and control variables.

2.2. Area-based decentralization for ORPF

The decomposition idea has been described in [6]. The basic approach is to divide the overall system into geographical areas. Any transmission line that crosses between two adjacent areas is divided into two sections by adding a "dummy bus" at the border between the two areas. Real and reactive power flow variables and voltage magnitude and angle variables are defined for the dummy bus and these four border variables are duplicated, with one copy assigned to each area. To be consistent with the overall formulation, any pair of corresponding duplicated variables in the two areas has the exactly same value.

The decomposition method of an interconnected power system can be explained using Fig. 1. The system consists of two areas as shown in Fig. 1a. Area-1 and area-2 are connected by the border bus B. $\boldsymbol{x}_{I1}$ and $\boldsymbol{x}_{I2}$ denote the internal variables belonging to each area. $\boldsymbol{x}_B$ is the border variables of bus B, which includes its voltage magnitude, voltage angle, active and reactive powers transferred along the tie-line. By splitting bus B into two duplicated dummy buses B_1 and B_2, as shown in Fig. 1b, two separated systems are obtained with each one having a dummy bus. At the same time, $\boldsymbol{x}_B$ is also duplicated as $\boldsymbol{x}_{B1}$ and $\boldsymbol{x}_{B2}$ which are assigned to area-1 and area-2 respectively.

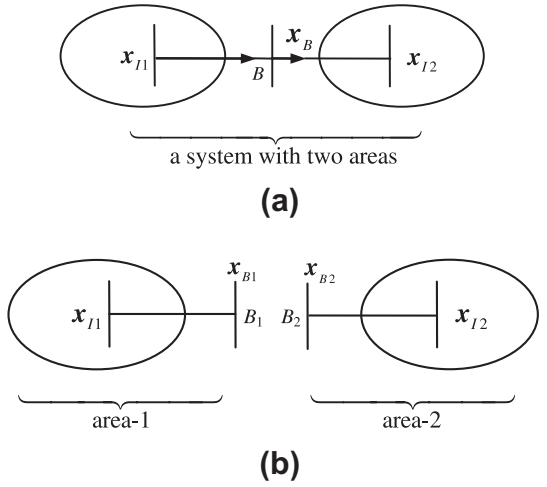

Fig. 1. Decomposition of an interconnected power system.

With such duplication of border variables, problem (1) is fully equivalent to the following expression:

$$\begin{aligned} \min \quad & f_1(\boldsymbol{x}_{I1}, \boldsymbol{x}_{B1}) + f_2(\boldsymbol{x}_{I2}, \boldsymbol{x}_{B2}) \\ s.t. \quad & \boldsymbol{g}_1(\boldsymbol{x}_{I1}, \boldsymbol{x}_{B1}) = 0 \\ & \boldsymbol{g}_2(\boldsymbol{x}_{I2}, \boldsymbol{x}_{B2}) = 0 \\ & \underline{\boldsymbol{x}}_{I1} \leqslant \boldsymbol{x}_{I1} \leqslant \bar{\boldsymbol{x}}_{I1} \\ & \underline{\boldsymbol{x}}_{I2} \leqslant \boldsymbol{x}_{I2} \leqslant \bar{\boldsymbol{x}}_{I2} \\ & \underline{\boldsymbol{x}}_{B1} \leqslant \boldsymbol{x}_{B1} \leqslant \bar{\boldsymbol{x}}_{B1} \\ & \underline{\boldsymbol{x}}_{B2} \leqslant \boldsymbol{x}_{B2} \leqslant \bar{\boldsymbol{x}}_{B2} \\ & \boldsymbol{x}_{B1} - \boldsymbol{x}_{B2} = 0 \end{aligned} \tag{2}$$

where $f_i(\boldsymbol{x}_{Ii}, \boldsymbol{x}_{Bi})$ represents the objective function of each subsystem or area-i, whose equality constraints are denoted by $\boldsymbol{g}_i(\boldsymbol{x}_{Ii}, \boldsymbol{x}_{Bi}) = 0$. Equation $\boldsymbol{x}_{B1} - \boldsymbol{x}_{B2} = 0$ represents the condition that any pair of duplicated variables in the two areas has the exactly same value.

Formulation (2) is the area-based decentralization model of ORPF. Apparently, formulation (2) can be extended to a system with N areas. By denoting $\boldsymbol{x}_i$ for $(\boldsymbol{x}_{Ii}, \boldsymbol{x}_{Bi})^T$ $(i = 1, 2)$ and $\boldsymbol{A}_i$ for the coefficient matrix representing the coupled border condition, a general formulation for the multiple-area model of ORPF can be expressed as follows:

$$\begin{aligned} \min \quad & F = \sum_{i=1}^{N} f_i(\boldsymbol{x}_i) \\ s.t. \quad & \boldsymbol{g}_i(\boldsymbol{x}_i) = 0 \\ & \underline{\boldsymbol{x}}_i \leqslant \boldsymbol{x}_i \leqslant \bar{\boldsymbol{x}}_i \\ & \sum_i \boldsymbol{A}_i \boldsymbol{x}_i = 0, \quad i = 1, \ldots, N \end{aligned} \tag{3}$$

It can be seen from formulation (3) that the objective function and all the constraints except for the coupled border condition can be divided into N independent sub-problems, and the variables in each sub-problem are associated with ones in other sub-problems only through the coupled border constraints.

3. Decomposition–coordination IPM

In this section, the decomposition–coordination interior point method (DIPM) for the distributed ORPF problem is presented. The key idea is the construction of a border-matrix-block and the decoupling of the Newton system in applying the IPM to problem (3).

3.1. IPM for ORPF

The ORPF problem shown in formulation (3) can be solved directly using the IPM. In this method, slack variables and Lagrange multipliers are introduced to deal with inequality and equality constraints, and logarithmic barrier functions are used to guarantee the non-negativity condition of slack variables. Then the ORPF problem given in the formulation (3) is transformed into one non-constrained optimization problem with the following Lagrange function:

$$\begin{aligned} L = & \sum_{i=1}^{N} f_i(\boldsymbol{x}_i) - \sum_{i=1}^{N} \boldsymbol{y}_i^T \boldsymbol{g}_i(\boldsymbol{x}_i) - \sum_{i=1}^{N} \boldsymbol{z}_i^T (\boldsymbol{x}_i - \boldsymbol{l}_i - \underline{\boldsymbol{x}}_i) + \sum_{i=1}^{N} \boldsymbol{w}_i^T (\boldsymbol{x}_i \\ & + \boldsymbol{u}_i - \bar{\boldsymbol{x}}_i) - \boldsymbol{y}_d^T \left[\sum_{i=1}^{N} \boldsymbol{A}_i \boldsymbol{x}_i \right] \\ & - \sum_{i=1}^{N} \mu_i \left[\sum_{j=1}^{n_i} \ln(l_{i(j)}) + \sum_{j=1}^{n_i} \ln(u_{i(j)}) \right] \end{aligned} \tag{4}$$

where both $\boldsymbol{y}_i$ and $\boldsymbol{y}_d$ are the vectors of Lagrange multipliers, in which $\boldsymbol{y}_i$ is related to the original equality constraints for each subsystem i and $\boldsymbol{y}_d$ to the coupled equality constraints; $\boldsymbol{l}_i, \boldsymbol{u}_i \geqslant 0, \boldsymbol{z}_i$, and $\boldsymbol{w}_i \geqslant 0$ are the vectors of slack variables and Lagrangian multipliers respectively; μ_i is a barrier parameter.

Based on the Karush–Kuhn–Tucker (KKT) first-order conditions of the problem (4), a set of nonlinear algebraic equations is formed as following:

$$\nabla_{x_i} L = \nabla f_i(\boldsymbol{x}_i) - \nabla \boldsymbol{g}_i(\boldsymbol{x}_i)^T \boldsymbol{y}_i - z_i + \boldsymbol{w}_i - \boldsymbol{A}_i^T \boldsymbol{y}_d = 0 \tag{5}$$

$$\nabla_{y_i} L = \boldsymbol{g}_i(\boldsymbol{x}_i) = 0 \tag{6}$$

$$\nabla_{l_i} L = \boldsymbol{Z}_i \boldsymbol{L}_i \boldsymbol{e} - \mu_i \boldsymbol{e} = 0 \tag{7}$$

$$\nabla_{u_i} L = \boldsymbol{W}_i \boldsymbol{U}_i \boldsymbol{e} - \mu_i \boldsymbol{e} = 0 \tag{8}$$

$$\nabla_{z_i} L = \boldsymbol{x}_i - \boldsymbol{l}_i - \underline{\boldsymbol{x}}_i = 0 \tag{9}$$

$$\nabla_{w_i} L = \boldsymbol{x}_i + \boldsymbol{u}_i - \bar{\boldsymbol{x}}_i = 0 \tag{10}$$

$$\nabla_{y_d} L = \sum_{i=1}^{N} \boldsymbol{A}_i \boldsymbol{x}_i = 0 \tag{11}$$

where $i = 1,\ldots, N$.$\boldsymbol{Z}_i$, $\boldsymbol{L}_i$, $\boldsymbol{W}_i$ and $\boldsymbol{U}_i$ are the diagonal matrices defined by the elements in the vectors $\boldsymbol{z}_i$, $\boldsymbol{l}_i$, $\boldsymbol{w}_i$, $\boldsymbol{u}_i$ respectively; $\boldsymbol{e}$ is a unit vector.

The nonlinear KKT equations can be solved using the Newton–Raphson algorithm. The Newton system is formed using the first order derivatives of the KKT equations, and the optimal solution can be obtained by means of alternate computations of the Newton system. By denoting $\boldsymbol{\rho}_i = [\boldsymbol{x}_i, \boldsymbol{y}_i, \boldsymbol{l}_i, \boldsymbol{u}_i, \boldsymbol{z}_i, \boldsymbol{w}_i]^T$ as the internal variables for each area i and $\boldsymbol{y}_d$ as the coupled border variables among subsystems, and having the variables organized in the structure of one area by one area, the Newton system can be expressed as the following system with border-matrix-blocks:

$$\begin{bmatrix} \boldsymbol{M}_1 & 0 & 0 & \boldsymbol{E}_1^T \\ 0 & \ddots & 0 & \vdots \\ 0 & 0 & \boldsymbol{M}_N & \boldsymbol{E}_N^T \\ \boldsymbol{E}_1 & \cdots & \boldsymbol{E}_N & 0 \end{bmatrix} \begin{bmatrix} \Delta\boldsymbol{\rho}_1 \\ \vdots \\ \Delta\boldsymbol{\rho}_N \\ \Delta\boldsymbol{y}_d \end{bmatrix} = - \begin{bmatrix} \boldsymbol{B}_1 \\ \vdots \\ \boldsymbol{B}_N \\ \nabla_{y_d} L_0 \end{bmatrix} \tag{12}$$

where,

$$M_i = \begin{bmatrix} \boldsymbol{H}_i & -\nabla g_i(x_i)^T & 0_{(n_i)} & 0_{(n_i)} & -\boldsymbol{I}_{(n_i)} & \boldsymbol{I}_{(n_i)} \\ \nabla \boldsymbol{g}_i(x_i) & 0_{(p_i)} & 0_{(p_i \times n_i)} & 0_{(p_i \times n_i)} & 0_{(p_i \times n_i)} & 0_{(p_i \times n_i)} \\ 0_{(n_i)} & 0_{(n_i \times p_i)} & \boldsymbol{Z}_i & 0_{(n_i)} & \boldsymbol{L}_i & 0_{(n_i)} \\ 0_{(n_i)} & 0_{(n_i \times p_i)} & 0_{(n_i)} & \boldsymbol{W}_i & 0_{(n_i)} & U_i \\ \boldsymbol{I}_{(n_i)} & 0_{(n_i \times p_i)} & -\boldsymbol{I}_{(n_i)} & 0_{(n_i)} & 0_{(n_i)} & 0_{(n_i)} \\ \boldsymbol{I}_{(n_i)} & 0_{(n_i \times p_i)} & 0_{(n_i)} & \boldsymbol{I}_{(n_i)} & 0_{(n_i)} & 0_{(n_i)} \end{bmatrix} \tag{13}$$

$$\boldsymbol{H}_i = \nabla^2 f_i(\boldsymbol{x}_i) - \sum_{j=1}^{p_i} y_{i(j)} \nabla^2 g_{i(j)}(\boldsymbol{x}_i) \tag{14}$$

$$\boldsymbol{B}_i = [\nabla_{x_i} L_0, \nabla_{y_i} L_0, \nabla_{l_i} L_0, \nabla_{u_i} L_0, \nabla_{z_i} L_0, \nabla_{w_i} L_0]^T \tag{15}$$

$$\boldsymbol{E}_i = [A_i \quad 0_{(q \times p_i)} \quad 0_{(q \times n_i)} \quad 0_{(q \times n_i)} \quad 0_{(q \times n_i)} \quad 0_{(q \times n_i)}] \tag{16}$$

where, n_i and p_i are the number of variables and equality constraints for subsystem i respectively, q is the number of the coupled border equality constraints, $0_{(n_i \times p_i)}$ and $0_{(n_i)}$ are $n_i \times p_i$ and $n_i \times n_i$ zero-element matrices, $\boldsymbol{I}_{(n_i)}$ is a $n_i \times n_i$ unit matrix. $\nabla_{x_i} L_0, \nabla_{y_i} L_0, \nabla_{l_i} L_0, \nabla_{u_i} L_0, \nabla_{z_i} L_0, \nabla_{w_i} L_0$ and $\nabla_{y_d} L_0$ are the residual vectors of the KKT equations as shown in Eqs. (5)–(11), where the index 0 means the initial condition of optimal variables $\boldsymbol{\rho}_i^0$ and $\boldsymbol{y}_d^0$.

The Newton system (12) can be reduced in the scale as follows:

$$\begin{bmatrix} \boldsymbol{M}_{R1} & 0 & 0 & \boldsymbol{E}_{R1}^T \\ 0 & \ddots & 0 & \vdots \\ 0 & 0 & \boldsymbol{M}_{RN} & \boldsymbol{E}_{RN}^T \\ \boldsymbol{E}_{R1} & \cdots & \boldsymbol{E}_{RN} & 0 \end{bmatrix} \begin{bmatrix} \Delta\boldsymbol{\rho}_{R1} \\ \vdots \\ \Delta\boldsymbol{\rho}_{RN} \\ \Delta\boldsymbol{y}_d \end{bmatrix} = - \begin{bmatrix} \boldsymbol{B}_{R1} \\ \vdots \\ \boldsymbol{B}_{RN} \\ \nabla_{y_d} L_0 \end{bmatrix} \tag{17}$$

where

$$\boldsymbol{M}_{Ri} = \begin{bmatrix} \boldsymbol{\Phi}_i & -\nabla \boldsymbol{g}_i(\boldsymbol{x}_i)^T \\ \nabla \boldsymbol{g}_i(\boldsymbol{x}_i) & 0_{(p_i)} \end{bmatrix}, \Delta\boldsymbol{\rho}_{Ri} = \begin{bmatrix} \Delta\boldsymbol{x}_i \\ \Delta\boldsymbol{y}_i \end{bmatrix}$$

$$\boldsymbol{B}_{Ri} = \begin{bmatrix} \boldsymbol{\psi}_i \\ \nabla_{y_i} L_0 \end{bmatrix}, \boldsymbol{E}_{Ri} = \begin{bmatrix} \boldsymbol{A}_i & 0_{(q \times p_i)} \end{bmatrix}$$

$$\boldsymbol{\Phi}_i = \nabla^2 f_i(\boldsymbol{x}_i) - \sum_{j=1}^{p_i} \boldsymbol{y}_{i(j)} \nabla^2 g_{i(j)}(\boldsymbol{x}_i) + \boldsymbol{Z}_i \boldsymbol{L}_i^{-1} + \boldsymbol{W}_i \boldsymbol{U}_i^{-1}$$

$$\boldsymbol{\psi}_i = \nabla_{x_i} L_0 + (\nabla_{l_i} L_0 + \boldsymbol{Z}_i \nabla_{z_i} L_0) \boldsymbol{L}_i^{-1} - (\nabla_{u_i} L_0 - \boldsymbol{W}_i \nabla_{w_i} L_0) U_i^{-1}$$

The following sub-vectors can be computed from Δx_i, Δy_i, which are included in $\Delta\boldsymbol{\rho}_{Ri}$ given in Eq. (17):

$$\begin{cases} \Delta\boldsymbol{l}_i = \Delta\boldsymbol{x}_i + \nabla_{z_i} L_0 \\ \Delta\boldsymbol{u}_i = -(\Delta\boldsymbol{x}_i + \nabla_{w_i} L_0) \\ \Delta z_i = (-\nabla_{l_i} L_0 - \boldsymbol{Z}_i \Delta\boldsymbol{l}_i) \boldsymbol{L}_i^{-1} \\ \Delta\boldsymbol{w}_i = (-\nabla_{u_i} L_0 - \boldsymbol{W}_i \Delta\boldsymbol{u}_i) \boldsymbol{U}_i^{-1} \end{cases} \tag{18}$$

The decomposition format of Eq. (17) is as follows:

$$\begin{aligned} &(-\boldsymbol{E}_{R1} \boldsymbol{M}_{R1}^{-1} \boldsymbol{E}_{R1}^T \cdots - \boldsymbol{E}_{RN} \boldsymbol{M}_{RN}^{-1} \boldsymbol{E}_{RN}^T) \Delta\boldsymbol{y}_d \\ &\quad = \boldsymbol{E}_{R1} \boldsymbol{M}_{R1}^{-1} \boldsymbol{B}_{R1} + \cdots + \boldsymbol{E}_{RN} \boldsymbol{M}_{RN}^{-1} \boldsymbol{B}_{RN} - \nabla_{y_d} L_0 \end{aligned} \tag{19}$$

$$\boldsymbol{M}_{Ri} \Delta\boldsymbol{\rho}_{Ri} = -\boldsymbol{B}_{Ri} - \boldsymbol{E}_{Ri}^T \Delta\boldsymbol{y}_d (i = 1, \ldots, N) \tag{20}$$

3.2. Distributed computations of ORPF

Each diagonal block $\boldsymbol{M}_{Ri}$, in the given system in Eqs. (17), (20) is an individual ORPF subsystem which has its own objective function, constraints and variables for each area. Thanks to the structure of the Newton system with border-matrix-blocks, the decomposition–coordination computations of the ORPF problem can be implemented in the following steps.

- Step 1: Calculate $\boldsymbol{E}_{Ri} \boldsymbol{M}_{Ri}^{-1} \boldsymbol{E}_{Ri}^T$, $\boldsymbol{E}_{Ri} \boldsymbol{M}_{Ri}^{-1} \boldsymbol{B}_{Ri}^T$ and $\boldsymbol{A}_i$ ***for each area on its sub-server processor separately.***
- Step 2: Send solutions obtained in step 1 from all sub-servers to the central coordinator server by internet or any other communication channel and then solve $\Delta\boldsymbol{y}_d$ from Eq. (19).
- Step 3: Send the data of $\Delta\boldsymbol{y}_d$ from the coordinator server to every sub-server located in each area, solve $\Delta\boldsymbol{\rho}_{Ri}$ from Eq. (20), and compute $\Delta\boldsymbol{l}_i$, $\Delta\boldsymbol{u}_i$, Δz_i and $\Delta\boldsymbol{w}_i$ from Eq. (18) on each area's server.

It can be seen that the Newton system and thus the original ORPF problem can be solved in a distributed mode.

3.3. Estimating step length and the barrier parameter

Usually the step length α and barrier parameter μ for each area can be selected as one unified value, i.e.,

$$\mu_1 = \mu_2 = \cdots = \mu_N = \mu = \sigma \frac{\sum_{i=1}^{N} (\boldsymbol{l}_i^T \boldsymbol{z}_i + \boldsymbol{u}_i^T \boldsymbol{w}_i)}{\sum_{i=1}^{N} n_i} \tag{21}$$

$$\alpha_1 = \alpha_2 = \cdots = \alpha_N = \alpha = 0.9995 \min_{i=1,\cdots,N} \{\alpha_{pi}, \alpha_{di}, 1\} \tag{22}$$

$$\alpha_{di} = \min_{(\Delta l_i)_j < 0, (\Delta u_i)_j < 0} \left\{ -\frac{(\boldsymbol{l}_i)_j}{(\Delta\boldsymbol{l}_i)_j}, -\frac{(\boldsymbol{u}_i)_j}{(\Delta\boldsymbol{u}_i)_j}, 1 \right\} (j = 1, \ldots, n_i) \tag{23}$$

$$\alpha_{pi} = \min_{(\Delta z_i)_j < 0, (\Delta w_i)_j < 0} \left\{ -\frac{(\boldsymbol{z}_i)_j}{(\Delta\boldsymbol{z}_i)_j}, -\frac{(\boldsymbol{w}_i)_j}{(\Delta\boldsymbol{w}_i)_j}, 1 \right\} (j = 1, \cdots, n_i) \tag{24}$$

where σ is the centering-parameter ranging from 0 to 1. The parameters α_{di} and α_{pi} are step lengths for primal variables and dual variables respectively.

3.4. Stopping criteria

The general stopping criteria in the IPM are: both the complementary gap G and mismatches D of the KKT equations are sufficiently small. In the proposed DIPM for the ORPF problem, each area has its own complementary gaps and mismatches. The stopping criteria are defined as follows:

$$G = \max([Gap_1, \cdots Gap_N]) < \varepsilon_1 \tag{25}$$

$$Gap_i = \boldsymbol{l}_i^T \boldsymbol{z}_i + \boldsymbol{u}_i^T \boldsymbol{w}_i \tag{26}$$

$$D = \max([D_1, \cdots D_N, D_d]) < \varepsilon_2 \tag{27}$$

$$D_i = ||[\nabla_{x_i} L_0, \nabla_{y_i} L_0, \nabla_{l_i} L_0, \nabla_{u_i} L_0, \nabla_{z_i} L_0, \nabla_{w_i} L_0]|| \tag{28}$$

$$D_d = ||\nabla_{y_d} L_0|| \tag{29}$$

where ε_1 and ε_2 are small positive values; G and D are the overall gap and mismatch for the central coordinator; and G_i and D_i are the gaps and mismatches for each area.

3.5. Overall procedure of DIPM for multi-area ORPF

The overall procedure of DIPM for multi-area ORPF can be summarized as follows:

- Step 1: Initialization. Choose an initial point satisfying non-negativity conditions as done in a general IPM [12], in which the initial multipliers are set as $\boldsymbol{y}_i^0 = \boldsymbol{1}$, $\boldsymbol{y}_d^0 = \boldsymbol{1}$, $z_i^0 = \boldsymbol{10}$ and $\boldsymbol{w}_i^0 = \boldsymbol{10}$; Set the tolerant errors in the stopping criteria ε_1, ε_2, initial iteration count $k = 0$ and maximum iteration count *MaxIter*.
- Step 2: Compute D_i and Gap_i on sub-servers in each area using Eqs. (26) and (28), and then transfer them to the central coordinator server with the border variables that are needed to compute D_d in Eq. (29). If the convergence criteria shown in Eqs. (25) and (27) are met, an optimal solution is reached and the calculation process is terminated; otherwise go to Step 3.
- Step 3: Set $k = k + 1$. If $k >$ *MaxIter*, it indicates that the method fails to converge and the calculation process is terminated; otherwise go to Step 4.
- Step 4: Implement the decomposition computations of the Newton subsystems as described in Part B of Section 3.
- Step 5: Determine the step length and barrier parameter using Eqs. (21)–(24) and update primal and dual variables. Go back to Step 2.

3.6. Amount of data communication

For a distributed algorithm, the amount of data communication is one of the most concerned factors. Large amount of data communication will cause communication bottlenecks and can not satisfy the need of the distributed computation. As discussed in Part B of Section 3, the transferred data between the sub-servers and the central coordinator is mainly involved in two $q \times q$ square matrices, i.e., $\boldsymbol{E}_{Ri}\boldsymbol{M}_{Ri}^{-1}\boldsymbol{E}_{Ri}^T$ and $\boldsymbol{E}_{Ri}\boldsymbol{M}_{Ri}^{-1}\boldsymbol{B}_{Ri}^T$, as well as two vectors of q dimensions, i.e., $\boldsymbol{A}_i\boldsymbol{x}_i$ and $\Delta\boldsymbol{y}_d$. The q is the number of the coupled border equality constraints as mentioned in Part A of Section 3. Note that we did not include D_i and Gap_i since they have a much smaller size than q. Therefore, the total amount of data communication between two regions in each iteration can be expressed as follows:

$$C_{DIPM} = 2(q_i^2 + q_i) \tag{30}$$

It can be seen from (30) that the amount of transferred data only relies on the size of coupled variables rather than the size of the whole system or any subsystem. In most cases, tie-lines between areas are really few and the q is a small figure. Thus the amount of data communication is quite small compared to centralized calculations and more importantly, most of the information on each subsystem would be kept undisclosed.

4. Simulation results

Five IEEE systems were used in simulations to demonstrate feasibility and effectiveness of the proposed method. The IEEE 14-bus, 30-bus and 118-bus systems are the original IEEE systems and each of them was divided into two areas. The IEEE-30 × 2 and IEEE-118 × 2 in the tables indicate that they are composed by two IEEE-30 systems or two IEEE-118 systems with each representing an area in the study.

Table 1 shows the basic information of each test system. The first column gives the case name. The second column shows the total number of buses in the system without dummy buses at the border. The basic information of the two subsystems including the duplicated dummy buses is shown in the third to seventh columns. The third column presents the number of buses in each area. The fourth column shows the number of interconnecting lines (tie-lines) between the two areas. The fifth, sixth, seventh and eighth columns present the total number of branches, generators, compensators and LTC Transformers for the two subsystems, respectively.

All the cases given in Table 1 were solved using two decomposition methods for the multi-area ORPF problem: (1) the traditional APP method; (2) the proposed DIPM. To validate the proposed distributed algorithm, we also used the centralized IPM (CIPM) for the cases. For the APP method, the optimization sub-problem for each area was also iteratively solved using the traditional IPM [12,14]. All the methods were implemented in the MATLAB that runs on an Intel Core2 E4300 processor. Several stopping tolerant errors 10^{-3}, 10^{-4}, 10^{-5} and 10^{-6} were applied in the simulations.

Table 2 shows the numerical results for the case studies. The second column indicates the method with different stopping tolerant errors. The third column shows the total number of iterations required to reach the optimum. The fourth column presents the total CPU time (in seconds) required to solve the overall problem. The fifth column shows active power losses (in per unit) as the objective function in the ORPF problem.

It is observed from the third column of Table 2 that the total number of iterations for the proposed DIPM is much smaller than

Table 1
Basic information of the case studies.

Name	Buses	Basic information of the subsystems					
		Bus	Tie	Branch	Generator	Compensator	LTC
IEEE-14	14	8.8	2	9.8	4.1	0.1	2.1
IEEE-30	30	15.18	3	17.20	5.1	1.1	3.1
IEEE-30 × 2	58	30.30	2	37.37	6.6	2.2	4.4
IEEE-118	118	84.38	4	129.18	14.5	13.1	6.3
IEEE-118 × 2	232	118.118	4	147.147	19.19	14.14	9.9

Table 2
Comparison of optimization results obtained using the three methods.

Case	Method	Iterations	CPU time (s)	PL (pu)
IEEE-14	APP (10^{-3})	8	2.013	0.1309
	APP (10^{-4})	18	4.698	0.1316
	APP (10^{-5})	23	6.466	0.1316
	APP (10^{-6})	28	7.211	0.1316
	DIPM (10^{-4})	12	0.258	0.1297
	DIPM (10^{-5})	15	0.313	0.1297
	DIPM (10^{-6})	15	0.313	0.1297
	CIPM (10^{-6})	15	0.267	0.1297
IEEE-30	APP (10^{-3})	13	4.556	0.0727
	APP (10^{-4})	22	8.573	0.0716
	APP (10^{-5})	30	13.883	0.0715
	APP (10^{-6})	39	18.5938	0.0715
	DIPM (10^{-4})	14	0.627	0.0681
	DIPM (10^{-5})	14	0.627	0.0681
	DIPM (10^{-6})	15	0.678	0.0681
	CIPM (10^{-6})	14	0.608	0.0681
IEEE-30 × 2	APP (10^{-3})	9	11.4594	0.1381
	APP (10^{-4})	12	15.281	0.1385
	APP (10^{-5})	19	24.453	0.1386
	APP (10^{-6})	30	39.681	0.1386
	DIPM (10^{-4})	12	1.183	0.1361
	DIPM (10^{-5})	13	1.272	0.1361
	DIPM (10^{-6})	13	1.272	0.1361
	CIPM (10^{-6})	15	1.845	0.1361
IEEE-118	APP (10^{-3})	11	58.2408	1.2316
	APP (10^{-4})	58	345.375	1.2063
	APP (10^{-5})	Fail to converge with maximal iteration 100		
	APP (10^{-6})	Fail to converge with maximal iteration 100		
	DIPM (10^{-4})	20	6.924	1.1706
	DIPM (10^{-5})	20	6.924	1.1706
	DIPM (10^{-6})	21	7.302	1.1706
	CIPM (10^{-6})	12	5.558	1.1706
IEEE-118 × 2	APP (10^{-3})	20	523.015	2.4693
	APP (10^{-4})	42	1116.187	2.4594
	APP (10^{-5})	Fail to converge with maximal iteration 100		
	APP (10^{-6})	Fail to converge with maximal iteration 100		
	DIPM (10^{-4})	19	22.256	2.3417
	DIPM (10^{-5})	19	22.256	2.3417
	DIPM (10^{-6})	21	24.715	2.3417
	CIPM (10^{-6})	19	31.568	2.3417

that for the APP in all the cases with each having the same stopping tolerance. The proposed method also shows better stability in convergence. It can be seen from the third and fifth columns that the iterations and objective function values of the proposed DIPM remain almost the same when the stopping tolerance is changed from 10^{-4} to 10^{-6} for each test system whereas the iteration numbers of the APP method increase greatly with the stopping tolerance decreasing. Particularly, in the cases of IEEE 118-bus and 118 × 2-bus systems, the APP method could not converge when the stopping tolerance is set at 10^{-5} or 10^{-6}. By comparing the results obtained using the DIPM and CIPM, it can be seen that the two methods require almost the same number of iterations to reach convergence except for the case of IEEE 118-bus where the proposed DIPM requires more iterations than the CIPM. The two methods achieved the exactly same optimal values of active power losses for the same stopping tolerance 10^{-6}. The results indicate that the effectiveness of the proposed DIPM can be ensured regardless of the size of the cases.

It can be observed from the fourth column of Table 2 that the total CPU time required by the proposed DIPM is much less than that required by the APP method in all the cases. This is due to the fact that the APP method is a two-stage algorithm, in which individual sub-problems for each area need to be solved by using the IPM in each iteration, whereas the proposed DIPM is a single-stage algorithm, in which only the decoupled Newton subsystems for each area need to be solved in each iteration. The number of iterations required for the IPM to solve a sub-problem is about 10 or even more and the total iterations for the IPM to solve two sub-problems corresponding to two areas have to be doubled. On the other hand, the proposed DIPM avoids solving sub-problems and only needs to solve the decoupled Newton system that corresponds to one overall problem. Additionally, the CPU times required by the DIPM and CIPM are comparable. In some of the given five cases, the DIPM requires more CPU than CIPM whereas in other cases, the DIPM requires less CPU than CIPM.

5. Conclusions

A decomposition–coordination interior point method for the multi-area ORPF problem is presented in this paper. The proposed method is based on a single-stage algorithm. It avoids sub-problems in each area of the multi-area ORPF and only needs to solve a decoupled Newton system in the IPM, which corresponds to the original overall ORPF problem. This feature enables the proposed method to have a much higher computing speed than the traditional APP method that has to solve sub-problems for each area.

Another feature of the proposed method lies in the fact that it is a direct implementation of the decomposition–coordination IPM and therefore inherits good performance of the traditional IPM. Mathematically, the proposed method can be also easily extended to other distributed optimization problems in power systems.

The paper also presents the distributed implementation process of the multi-area ORPF, which requires coordination between the central coordinator server and local servers in each area. The method only needs a transfer of the information associated with two matrices and two vectors with a greatly reduced size, which are only related to dummy buses at the border between areas, rather than the information associated with whole subsystems. The proposed method helps maintain autonomy of each area in a global system by means of a coordinated but decentralized procedure.

Numerical results demonstrate that the proposed method is superior to the traditional APP method in iterations and CPU time, stability in convergence and optimality.

Acknowledgement

This work was supported by The National Natural Science Foundation of China under Contract 50577073 and Scientific Research Foundation of State Key Lab. of Power Transmission Equipment and System Security (2007DA10512710204).

References

[1] Deeb NI, Shahidehpour SM. Linear reactive power optimization in a large power network using the decomposition approach. IEEE Trans Power Syst 1990;5(2):428–38.
[2] Deeb NI, Shahidehpour SM. Decomposition approach for minimizing real power losses in power systems. Proc IEEE, Part C 1991;138(1):27–38.
[3] Zhuang F, Galiana FD. Toward a more rigorous and practical unit commitment by Lagrangian relaxation. IEEE Trans Power Syst 1988;3(2):763–73.
[4] Batut J, Renaud A. Daily generation scheduling optimization with transmission constraints: a new class of algorithms. IEEE Trans Power Syst 1992;7(3):982–9.
[5] Ebrahimian R, Baldick R. State estimation distributed processing. IEEE Trans Power Syst 2000;15(4):1240–6.
[6] Kim BH, Baldick R. Coarse-grained distributed optimal power flow. IEEE Trans Power Syst 1997;12(2):932–9.
[7] Kim BH, Baldick R. A comparison of distributed optimal power flow algorithms. IEEE Trans Power Syst 2000;15(2):599–604.
[8] Cheng X, Li J, Cao L, Liu X. Multi-objective distributed parallel reactive power optimization based on sub-area division of the power systems. Proc CSEE 2003;23(10):109–13.
[9] Nogales FJ, Prieto FJ, Conejo AJ. A decomposition methodology applied to the multi-area optimal power flow problem. Ann Oper Res 2003;120(1–4):99–116.
[10] Conejo AJ, Nogales FJ, Prieto FJ. A decomposition procedure based on approximate Newton directions. Math Progr 2002;93(3):495–515.

[11] Rakpenthai C, Premrudeepreechacharn S, Uatrongjit S. Power system with multi-type FACTS devices states estimation based on predictor–corrector interior point algorithm. Int J Electr Power Energy Syst 2009;31(4):160–6.
[12] Wei Y, Yu J, Yu DC, Bhattarai K. A new optimal reactive power flow model in rectangular form and its solution by predictor corrector primal dual interior point method. IEEE Trans Power Syst 2006;21(1):61–7.
[13] Silva Jr Ivo C, Carneiro Jr Sandoval, de Oliveira Edimar J, Pereira JLR, Garcia Paulo AN, et al. A Lagrangian multiplier based sensitive index to determine the unit commitment of thermal units. Int J Electr Power Energy Syst 2008;30(9):504–10.
[14] Yu J, Wei Y, Li W. An unfixed piecewise optimal reactive power flow model and its algorithm for AC–DC systems. IEEE Trans Power Syst 2008;23(1):170–6.
[15] Bai X, Wei H, Fujisawa K, Wang Yong. Semidefinite programming for optimal power flow problems. Int J Electr Power Energy Syst 2008;30(6–7):383–92.