

预想事故安全约束最优潮流问题的并行多中心校正内点算法

李 捷¹, 王 鹏², 杨林峰³

(1 广西职业技术学院 计算机与电子信息工程系, 广 西 南 宁 530226; 2 渤海石油装备中成机械制造有限公司, 天津 300280;
3 广西大学 计算机与电子信息学院, 广 西 南 宁 530004)

摘 要: 对预想事故安全约束最优潮流(expected security cost optimal power flow, ESCOPF)问题模型进行讨论, 探讨了内点法求解该问题的修正方程的特殊结构, 并根据该修正方程结构, 结合带权多中心校正内点技术, 设计了一个并行多中心校正内点算法. 算法可以进一步提高模型求解速度, 具有良好的加速比和可扩展性.

关键词: 内点算法; 并行算法; 中心校正; 预想事故安全约束

中图分类号: TP301.6

文献标识码: A

文章编号: 1000-7180(2014)02-0137-05

Parallel Multiple Centrality Correctors Interior Point Method for ESC-OPF

LI Jie¹, WANG Peng², YANG Lin-feng³

(1 Department of Computer, Electronics and Information Engineering, Guangxi Polytechnic, Nanning 530226, China;
2 Bohai Petroleum Equipment Zhongcheng Machinery Manufacturing Company, Tianjin 300280, China;
3 School of Computer and Electronics and Information, Guangxi University, Nanning 530004, China)

Abstract: This paper presents the huge but useful formulation of expected security cost optimal power flow (ESCOPF). Corrector equation, which has special structure, of interior-point method (IPM) for ESC-OPF has been given. A novel parallel IPM based on Multiple Centrality Correctors (MCC) has been proposed to solve the formulation of this ESCOPF model. In our novel algorithm, iterative times can be reduced and CPU time can be shut down because of the good speedup.

Key words: interior point method; parallel algorithm; centrality correctors; expected security cost

1 引言

最优潮流(optimal power flow, OPF)问题首先由法国电力公司的 Carpentier 提出. 传统的 OPF 研究主要研究可行 OPF 问题^[1], 也就是 N-0 问题, 即 0 故障问题. 近年来, N-1 带安全约束的静态 OPF 问题成为了研究的热点, 即同一时间内只有一个事故发生的情况下的 OPF 问题. 在保证故障后安全的前提下, 计算最小代价的运行状态就是所谓的安全约束最优潮流问题(security-constrained optimal power flow, SCOPF)^[2]. 如果用系统调整的经济开

销来代替约束, 并且考虑各个事故发生的可能概率, 来计算 OPF 就是所谓的预想事故安全约束最优潮流问题(expected security cost optimal power flow, ESCOPF)^[3]. 文献[4]将预测—校正内点法用于 OPF 问题求解, 取得了很好的数值效果.

2 预想事故安全约束最优潮流模型

在 ESCOPF 中, 文献[3]将安全问题考虑成为一个经济费用而不是一个约束. 文献[3]的作者 P. Damrongkulkamjorn 和 T. W. Gedra. 随后对该模型展开了深入的研究^[5], 并取得了较好的效果. 参考

收稿日期: 2013-03-19; 修回日期: 2013-05-22

基金项目: 国家自然科学基金项目(71201049); 广西自然科学基金项目(2013GXNSFBA019246); 广西大学科研基金资助项目(XJZ110593); 广西高等学校科研资助项目(201106LX028)

文献[3,5]可建立如下 ESCOPF 模型:

$$\begin{aligned} \min_{\{Y^0, Y^1, \dots, Y^K\}} & \pi^0 C^0(Y^0) + \sum_{k=1}^K \pi^k C^k(Y^0, Y^k) \\ \text{s. t.} & \begin{cases} h^k(Y^k) = 0, k = 0, 1, \dots, K \\ g^k(Y^k) \leq 0, k = 0, 1, \dots, K \\ d^k(Y^0, Y^k) \leq 0, k = 1, \dots, K \end{cases} \end{aligned} \quad (1)$$

式中, Y^0 为事故前稳态的控制变量和状态变量; Y^k 为事故发生后系统调整至新的 k 稳定点时的控制变量和状态变量; 假设系统中共有 K 个可能发生故障; π^0 为系统无事故发生的可能性; π^k 为系统发生故障 k 的可能性. $C^0(Y^0)$ 代表系统在事故前的运行费用; $C^k(Y^0, Y^k)$ 为系统在初始状态 Y^0 , 发生故障 k 后, 调整进入新的稳态 Y^k 后的运行费用. h^0 为事故前的等式约束, 一般为潮流约束; g^0 为事故前的不等式约束, 一般为界约束; h^k 为事故 k 发生后的等式约束; g^k 为事故后的不等式约束; d^k 为事故 k 发生后的耦合约束, 该约束中一般包含事故前和事故后的状态变量和控制变量, 例如支路潮流约束或者发电机爬坡约束.

3 ESCOPF 的并行多中心校正内点算法

对于同样规模的网络来说, ESCOPF 问题的规模是经典 OPF 的 K 倍, 采用经典内点法求解 ESCOPF 则比解 OPF 的计算量大得多, 为了能够实时得到 ESCOPF 结果以对电力系统进行调整, 在改进算法本身的效率的同时, 充分利用算法特点, 结合同步并行分布式计算技术, 来进一步减少计算时间, 也是推进 ESCOPF 进入实用的一个重要手段. 我们对多中心校正 (Multiple Centrality Corrections, MCC) 技术进行改进, 并基于改进的 MCC 给出了一种求解 ESCOPF 的并行带权多中心内点算法.

对式(1), 引入松弛变量将不等式变成等式约束, 再引入内点障碍函数去掉不等式约束, 写出其 Lagrangian 函数如下:

$$\begin{aligned} L = & \pi^0 C^0(Y^0) - \mu \sum_{k=0}^K \left(\sum_{i=1}^{n_k^k} \ln s_i^k \right) + \sum_{k=1}^K \pi^k C^k(Y^0, Y^k) \\ & - \mu \sum_{k=1}^K \left(\sum_{i=1}^{n_d^k} \ln s_{di}^k \right) + \sum_{k=0}^K \lambda^{kT} h^k(Y^k) + \\ & \sum_{k=0}^K u^{kT} (g^k(Y^k) + s^k) + \sum_{k=1}^K v^{kT} (d^k(Y^0, Y^k) + s_d^k) \end{aligned} \quad (2)$$

计算 Lagrangian 函数对偶变量 Y^0, Y^k , Lagrangian 乘子 λ^k, u^k 以及 v^k , 松弛变量 s^k, s_d^k 的偏导数如下:

$$\begin{aligned} \nabla_{Y^0} L = & \pi^0 \frac{\partial C^0(Y^0)}{\partial Y^0} + \sum_{k=1}^K \pi^k \frac{\partial C^k(Y^k, Y^0)}{\partial Y^0} + \\ & \lambda^{0T} \frac{\partial h^0(Y^0)}{\partial Y^0} + u^{0T} \frac{\partial g^0(Y^0)}{\partial Y^0} + \sum_{k=1}^K v^{kT} \frac{\partial d^k(Y^0, Y^k)}{\partial Y^0} \end{aligned} \quad (3)$$

式中, $\lambda^{0T} \frac{\partial h^0(Y^0)}{\partial Y^0} = J_{h^0}^T \lambda^0$, 其他类似.

$$\begin{aligned} \nabla_{Y^k} L = & \pi^k \frac{\partial C^k(Y^k, Y^0)}{\partial Y^k} + \lambda^{kT} \frac{\partial h^k(Y^k)}{\partial Y^k} + \\ & u^{kT} \frac{\partial g^k(Y^k)}{\partial Y^k} + v^{kT} \frac{\partial d^k(Y^0, Y^k)}{\partial Y^k} \end{aligned} \quad (4)$$

$$\nabla_{\lambda^0} L = \pi^0 h^0(Y^0) \quad (5)$$

$$\nabla_{\lambda^k} L = \pi^k h^k(Y^k), k = 1, \dots, K \quad (6)$$

$$\nabla_{u^0} L = s^0 + g^0(Y^0) = 0 \quad (7)$$

$$\nabla_{u^k} L = s^k + g^k(Y^k) = 0, k = 1, \dots, K \quad (8)$$

$$\nabla_{v^k} L = s_d^k + d^k(Y^0, Y^k) = 0, k = 1, \dots, K \quad (9)$$

$$\nabla_{s^0} L = -\mu \frac{1}{s^0} + u^{0T} S^0 e - \mu e = 0 \quad (10)$$

$$\begin{aligned} \nabla_{s^k} L = & -\mu \frac{1}{s^k} + u^{kT}, k = 1, \dots, K \Leftrightarrow U^k S^k e - \mu e = \\ & 0, k = 1, \dots, K \end{aligned} \quad (11)$$

$$\begin{aligned} \nabla_{s_d^k} L = & -\mu \frac{1}{s_d^k} + v^{kT}, k = 1, \dots, K \Leftrightarrow V^k S_d^k e - \\ & \mu e = 0, k = 1, \dots, K \end{aligned} \quad (12)$$

其中: $\mu > 0$; 由 $\ln s_i^k, \ln s_{di}^k$ 得 $s^0, s^k, s_d^k > 0, k = 1, \dots, K$, 则 $u^0, u^k, v^k > 0, k = 1, \dots, K$; $S^0 = \text{diag}(s^0)$, $U^0 = \text{diag}(u^0)$, e 为所有元素都为 1 的向量, $S^k = \text{diag}(s^k)$, $S_d^k = \text{diag}(s_d^k)$, $V^k = \text{diag}(v^k)$.

式(3)~(12)为 ESCOPF 问题的近似最优性条件, 记为 $\otimes$. 将 $\otimes$ 按照 (3)(5)(7)(10)(4)(6)(8)(12)(9)(11) 的顺序重新排列, 这样则按 Y^k 将所有式子分组. 采用 MCC 内点法求解上述 ESCOPF 问题, 需要计算如下修正方程:

$$\begin{bmatrix} W_0 & W_{01} & W_{02} & \cdots & W_{0K} \\ W_{10} & W_1 & 0 & \cdots & 0 \\ W_{20} & 0 & W_2 & \cdots & 0 \\ \vdots & 0 & 0 & \ddots & 0 \\ W_{K0} & 0 & \cdots & 0 & W_K \end{bmatrix} \begin{bmatrix} \Delta z_0 \\ \Delta z_1 \\ \vdots \\ \Delta z_K \end{bmatrix} = - \begin{bmatrix} \omega_0(Z) \\ \omega_1(Z) \\ \vdots \\ \omega_K(Z) \end{bmatrix} \quad (13)$$

式中, $Z = [z_0; z_1; \dots; z_K]$, $z_0 = [Y^0; \lambda^0; u^0; s^0]$, $z_k = [Y^k; \lambda^k; u^k; s^k; v^k; s_d^k], k = 1, \dots, K$, $W_k, k = 0, \dots, K$ 与经典 OPF 问题的修正方程具有类似结构, $W_{k0} = W_{0k}^T = \begin{bmatrix} H_{k0}, 0, 0, 0, 0, 0, 0, 0, 0, 0 \\ 0, 0, 0, 0, 0, D_{k0}^k, 0, 0, 0, 0, 0 \end{bmatrix}$,

$$H_{k0} = \pi^k \frac{\partial^2 C^k(Y^k, Y^0)}{\partial Y^k \partial Y^0} + \sum_i^{n_d^k} v_i^k \frac{\partial^2 d_i^k(Y, {}^0Y^k)}{\partial Y^0 \partial Y^k}, D_{Y^0}^k \text{ 为}$$

耦合约束对 Y^0 的雅克比矩阵。

3.1 修正方程的并行求解

我们将采用分解-回代的方式求解修正方程。将式(13)做如下调整:

$$\begin{bmatrix} W_K & 0 & \cdots & 0 & W_{K0} \\ 0 & W_1 & 0 & \cdots & W_{10} \\ \vdots & 0 & W_2 & \cdots & W_{20} \\ 0 & 0 & 0 & \ddots & \vdots \\ W_{0K} & W_{01} & W_{02} & \cdots & W_0 \end{bmatrix} \begin{bmatrix} \Delta z_K \\ \Delta z_1 \\ \Delta z_2 \\ \vdots \\ \Delta z_{K-1} \\ \Delta z_0 \end{bmatrix} = - \begin{bmatrix} \omega_K(Z) \\ \omega_1(Z) \\ \omega_2(Z) \\ \vdots \\ \omega_{K-1}(Z) \\ \omega_0(Z) \end{bmatrix} \quad (14)$$

对于新的修正方程(14),进行 Cholesky 分解:

$$\begin{bmatrix} W_K & 0 & \cdots & 0 & W_{K0} \\ 0 & W_1 & 0 & \cdots & W_{10} \\ \vdots & 0 & W_2 & \cdots & W_{20} \\ 0 & 0 & 0 & \ddots & \vdots \\ W_{0K} & W_{01} & W_{02} & \cdots & W_0 \end{bmatrix} = \begin{bmatrix} L_K & & & & \\ & L_1 & & & \\ & & L_2 & & \\ & & & \ddots & \\ B_{0K} & B_{01} & B_{02} & \cdots & L_0 \end{bmatrix} \cdot \begin{bmatrix} L_K^T & & & & \\ & L_1^T & & & \\ & & L_2^T & & \\ & & & \ddots & \\ & & & & L_0^T \end{bmatrix} \begin{bmatrix} B_{0K}^T \\ B_{01}^T \\ B_{02}^T \\ \vdots \\ L_0^T \end{bmatrix} \quad (15)$$

其中:

$$L_i L_i^T = W_i, i = K, 1, 2, \dots, K-1 \quad (16)$$

$$B_{0i} = W_{0i} L_i^{-T}, i = K, 1, 2, \dots, K-1 \quad (17)$$

$$L_0 L_0^T = W_0 - \sum_{i=1}^K B_{0i} B_{0i}^T \quad (18)$$

分解完成后,再进行回代完成修正方程的求解:

$$\begin{cases} \tau_p = 0.9995 \min \left\{ \min_{i,k} \left\{ \frac{-s_i^0}{\Delta_a s_i^0} \left| \Delta_a s_i^0 < 0, \frac{-s_i^k}{\Delta_a s_i^k} \left| \Delta_a s_i^k < 0, \frac{-s_{di}^k}{\Delta_a s_{di}^k} \left| \Delta_a s_{di}^k < 0 \right\}, 1 \right\} \right. \\ \left. \tau_d = 0.9995 \min \left\{ \min_{i,k} \left\{ \frac{-u_i^0}{\Delta_a u_i^0} \left| \Delta_a u_i^0 < 0, \frac{-u_i^k}{\Delta_a u_i^k} \left| \Delta_a u_i^k < 0, \frac{-v_i^k}{\Delta_a v_i^k} \left| \Delta_a v_i^k < 0 \right\}, 1 \right\} \right. \right\} \end{cases} \quad (24)$$

计算预测对偶间隙:

$$g_a = \sum_{i=0}^K (s^i + \tau_p \Delta_a s^i)^T (u^i + \tau_p \Delta_a u^i) + \sum_{i=1}^K (s_d^i + \tau_d \Delta_a s_d^i)^T (v^i + \tau_d \Delta_a v^i) \quad (25)$$

$$\begin{bmatrix} L_K^T & & & & B_{0K}^T \\ & L_1^T & & & B_{01}^T \\ & & L_2^T & & B_{02}^T \\ & & & \ddots & \vdots \\ & & & & L_0^T \end{bmatrix} \begin{bmatrix} \Delta z_K \\ \Delta z_1 \\ \Delta z_2 \\ \vdots \\ \Delta z_{K-1} \\ \Delta z_0 \end{bmatrix} = \begin{bmatrix} \ell_K \\ \ell_1 \\ \ell_2 \\ \vdots \\ \ell_{K-1} \\ \ell_0 \end{bmatrix} \quad (19)$$

$$L_i \ell_i = \omega_i(Z), i = K, 1, 2, \dots, K-1 \quad (20)$$

$$L_0 \ell_0 = \omega_0(Z) - \sum_{i=1}^K B_{0i} \ell_i \quad (21)$$

$$L_0^T \Delta z_0 = \ell_0, L_i^T \Delta z_i = \ell_i - \ell_0, i = K, 1, 2, \dots, K-1 \quad (22)$$

上述分解-回代过程中,式(16)、(17)都分别为 K 个独立的子问题,可以并行进行计算;式(20)为 K 个独立的下三角线性方程组,可以并行进行求解,式(22)为 $K+1$ 个独立的上三角线性方程组,也可以并行进行求解。

我们采用分解-回代方式求解修正方程有几个优点^[7]:①可以充分利用矩阵的特殊结构,并行进行分解和回代;②由于矩阵的分解的计算量往往比回代大得多,可以充分利用分解后的信息,增加回代次数来减少内点迭代的次数,以减少计算时间。

3.2 算法描述

(1) 产生预测校正方向 Δ_p

为实现并行带权多中心内点算法,首先需要先计算 Mehrotra 预测校正方向。为了叙述方便将式(14)记做:

$$MZ' = -W(Z) \quad (23)$$

假设我们的当前迭代点是: Z^k , PCIPM 法的方向 $\Delta_p = \Delta_a + \Delta_c$, 其中 Δ_a 为仿射方向, Δ_c 为校正方向。

① 仿射方向 Δ_a

对式(23)中 M 进行式(15)~(18)所示的并行分解,然后令 $v=0$,第一次并行回代得到仿射方向 Δ_a 。然后计算仿射步长:

② 校正方向 Δ_c

计算扰动因子: $\mu = \left(\frac{g_a}{g} \right)^3 \left(\frac{g}{\sum_{i=0}^K n_g^i + \sum_{i=1}^K n_d^i} \right)$ 。在

式(23)中令 $\omega_0 = [0; 0; 0; \mu e - \Delta_a^0 \Delta_a u_a^0]$, 令 $\omega_i =$

$[0; 0; 0; \mu e - \Delta s_{da}^i \Delta u_a^i e; 0; \mu e - \Delta s_{da}^i \Delta v_a^i e], i = 1, \dots, K$, 第二次并行回代得 Δ_i . 其中 $\Delta s_{da}^i \Delta u_a^i e$ 以及 $\Delta s_{da}^i \Delta v_a^i e$ 项的引入可以抵消在预测点的二阶误差, 实际上就是对仿射方向的二阶修正.

在路径跟随内点法中, 如果迭代序列点越接近中心轨迹点则, 迭代收敛的速度越快, 然后 Mehrotra 在文献[8]中通过大量的数值实验表明, 更高阶的近似可以使得迭代次数下降, 但是由于超过二阶的 Taylor 展开式计算本身的复杂性, 使得整个迭代过程的计算时间反而有所增加. 本文的并行带权多中心内点法, 充分利用式(23)中 M 矩阵已经分解的信息, 以比较小的计算代价来进一步提高迭代点的中心性, 以减少迭代次数, 并最终达到减少计算时间的目的.

(2) 产生多中心校正方向 $\Delta = \Delta_p + \Delta_m$

① 沿着 Δ_p 将仿射步长拉大: $\tilde{\tau}_p = \min(\tau_p + \delta_a, 1)$, $\tilde{\tau}_d = \min(\tau_d + \delta_a, 1)$, 得到新的预测点: $\tilde{Z}_p^k = Z_p^k + \tilde{\tau}_p \Delta_{pp}$, $\tilde{Z}_d^k = Z_d^k + \tilde{\tau}_d \Delta_{pd}$;

② 迭代点的中心性的衡量标准为原问题的扰动 KKT 条件 $\otimes$ 中互补乘积接近扰动因子 μ 的程度. 若所有互补乘积都近似等于 μ , 则当前迭代点的中心性好, 反之则差. 为进一步提高迭代点的中心性, 需要将所有互补乘积校正至中心轨迹的领域内. 令 $\tilde{m}_0 = [\tilde{S}^{0k} \tilde{U}^{0k}]$, 引入: $m_{0i} = \pi(\tilde{m}_0 | H)$, 其中 $H = [\beta_{\min} \mu, \beta_{\max} \mu]$, 一般可以考虑 $\beta_{\min} = 0.1, \beta_{\max} = 10$, 即

$$m_{0i} = \begin{cases} \tilde{m}_0 & \tilde{m}_0 \in H \\ \beta_{\min} \mu & \tilde{m}_0 < \beta_{\min} \mu \\ \beta_{\max} \mu & \tilde{m}_0 > \beta_{\max} \mu \end{cases}$$

同理, 令 $\tilde{m}_i = [\tilde{S}^{ik} \tilde{U}^{ik}]$, 引入: $m_{it} = \pi(\tilde{m}_i | H)$.

③ 令 $\omega_0 = -[0; 0; 0; m_{0t} - \tilde{m}_0]$, 令 $\omega_i = -[0; 0; 0; m_{it} - \tilde{m}_i |_{\text{line1-n}_{gi}}]$, $i = 1, \dots, K$, 再一次并行回代解式(24), 得: Δ_m , 再令: $\Delta' = \Delta_p + \omega \Delta_m$, $\omega \in (0, 1]$, 进行线搜索, 确定 ω 使得 Δ' 方向步长最大, 为 $\hat{\tau}_p, \hat{\tau}_d$.

④ 若 $\hat{\tau}_p \geq \tau_p + \gamma \delta_a$ 且 $\hat{\tau}_d \geq \tau_d + \gamma \delta_a$, 则取 $\Delta_p = \Delta'$, 转 ①. 否则, 多中心校正结束, $\Delta = \Delta_p$.

(3) 沿多中心校正方向 Δ , 以及最长步长 $\hat{\tau}_p, \hat{\tau}_d$ 得到新的迭代点, 并一直迭代至最优或者不收敛终止.

4 仿真与分析

利用本算法, 使用 Matlab 2010b 编写程序, 对 14 节点至 300 节点共计 4 个系统进行数值试验. 实验平台为多核集群系统, 共使用 8 个节点, 每个节点 2 个计算内核. 节点基本信息: Intel I5 2520 双核处理器, 2.5 GHz, 3 MB 缓存; 4 GB RAM; Windows 7 64 bit 操作系统. 节点间采用千兆以太网交换机互联, 集群系统采用 Matlab Parallel Computing Toolbox (PCT) 4.3^[7,9] 搭建, 集群上配置 1 个 Matlab 的 JobManger, 1 个 Client, 16 个 Worker. 测试系统的基本参数见表 1 所示, 所有测试系统的预想事故都假设为线路断路故障, 对于断路后形成的不可行问题, 则假设该线路的故障概率为 0, 其他线路的故障概率假设为 0.1/总线路数.

表 2 给出了算法串行运行的计算结果. 如表 2 所示, 算法迭代次数受系统规模影响不大, 对于所有系统都能在合理的时间内收敛.

图 1 统计了采用本文算法计算以上 4 个系统的并行加速比情况, 算法加速比随处理器核数目增加能够得到有效提升, 同时问题规模越大, 算法加速比则更加显著, 主要因为问题规模越大, 通信开销所占比例则越小, 并行计算所占比例越大, 则并行效率越高, 加速比越高.

表 1 测试系统参数

测试系统	节点	发电机	线路
IEEE14	14	3	20
IEEE30	30	6	41
IEEE118	118	16	179
IEEE300	300	21	409

表 2 算法串行计算结果

测试系统	迭代次数	计算时间/s
IEEE14	33	2.03
IEEE30	19	4.81
IEEE118	41	34.13
IEEE300	39	83.22
IEEE14	41	158.31

5 结束语

本文结合内点算法、MCC 技术以及并行算法理论, 给出了一个求解 ESCOPF 问题的并行内点算

法,该算法在求解 ESCOPF 问题时候可以避免对原问题进行分解而产生的误差,得到更加准确的解.同时算法具有较好的并行加速比,可扩展性,适合求解大规模 ESCOPF 问题.

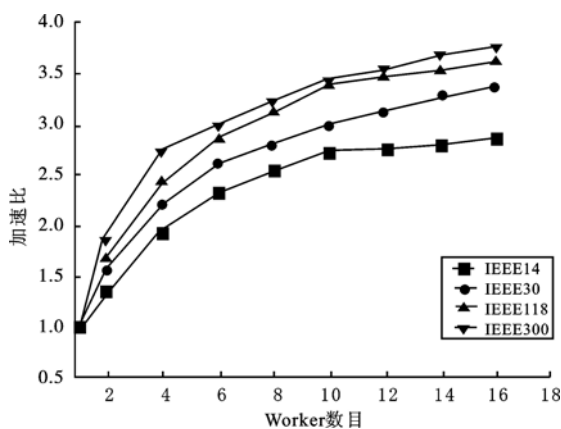


图1 并行计算加速比

参考文献:

- [1] Wei H, Sasaki H, Yokoyama R. An application of interior point quadratic programming algorithm to power system optimization problems[J]. IEEE Transactions on Power Systems, 1996, 11(1): 260-267.
- [2] Monticelli M P A, Granville S. Security constrained optimal power flow with post-contingency corrective rescheduling[J]. IEEE Transactions on Power Systems, 1987, 2(1): 175-182.
- [3] Damrongkulkamjorn P. Optimal power flow with expected security costs[D]. USA: Oklahoma State University, 1999.
- [4] 覃振成, 乐秀瑶, 蓝澜, 等. 基于改进非线性预报-校正内点法的最优潮流[J]. 电网技术, 2005, 29(9): 25-30.
- [5] Condren J, Gedra T W, Damrongkulkamjorn P. Optimal power flow with expected security costs[J]. IEEE Transactions on Power Systems, 2006, 21(2): 541-547.
- [6] Mehrotra S. On the implementation of a primal-dual interior method[J]. SIAM Journal on Optimization, 1992, 2(4): 575-601.
- [7] Gondzio J. Multiple centrality corrections in a primal-dual method for linear programming[J]. Computational Optimization and Applications, 1996(6): 137-156.
- [8] 郑汉垣, 刘智翔, 封卫兵, 等. MPI+OpenMP 混合编程模型在大规模三对角线性方程组求解中的应用[J]. 微电子学与计算机, 2011, 28(8): 157-160.
- [9] Matlab Inc. Matlab parallel computing toolbox 4 [EB/OL]. (2012-11-10) [2013-04-20]. http://www.mathworks.com/access/helpdesk/help/pdf_doc/distcomp/distcomp.pdf.

作者简介:

李捷 女, (1979—), 硕士, 讲师. 研究方向为并行与分布式计算及其应用.

杨林峰 男, (1979—), 博士, 副教授. 研究方向为并行最优化算法及其应用.