

基于改进型纵横交叉算法的电力系统无功规划优化

将纵横交叉 (CSO) 算法应用到电力系统无功规划优化问题, 并对CSO算法进行改进。运用混沌映射产生分布均匀的初始解, 引进纵交叉微算子, 并将量子粒子群算法搜索机制与CSO结合, 增加算法的搜索能力。利用电压稳定裕度指标, 找寻系统电压稳定性较薄弱节点, 将其作为无功规划节点, 并将综合规划费用和电压稳定裕度指标写入目标函数, 追寻综合规划费用最低的同时提高电压稳定性。通过IEEE-30节点系统的仿真, 验证了该规划方法和算法的有效性。

殷 豪 陈志江 陈智慧 刘 凯 郭壮志/广东工业大学自动化学院



殷 豪/副教授

无功规划优化是多变量、多约束的混合非线性规划问题, 求解方法主要有数学规划法和各种智能算法。它是确定无功补偿设备的最优安装位置和容量, 保持电力系统电压稳定、降低网损的重要措施。

文献[1-3]提出利用模态分析法或电压稳定裕度指标分析系统的负荷节点, 并在电压稳定性较弱节点安装无功补偿装置。文献[4]将安装无功补偿装置费用最小引入无功规划的目标函数。文献[5]将电压稳定裕度指标引入到无功规划的目标函数。文献[6]采用灵敏度分析方法选择少数高灵敏度节点作为候选补偿点。文献[7]以最大化潮流方程雅可比矩阵最小的奇异值作为无功规划目标函数。文献[8]运用最优网损微增率准则和层次聚类法确定待补偿点, 运用改进蚁群算法以网损最小为目标函数搜索最佳补偿容量。文献[9]将阻抗矩阵对角线元

素与相应负荷阻抗的模之比 I_{vs} 列为电压稳定裕度指标, 用于补偿点的选取, 把电压稳定裕度指标和网损共同引入了目标函数, 运用改进粒子群算法寻找补偿容量最优值。文献[10]提出了多种负荷预测方式下综合效果最优的无功规划模型。文献[11]将改进灾变遗传算法运用于无功规划优化。文献[12]将基于非线性互补理论的内点法应用于求解含有电压稳定裕度约束条件的无功规划问题, 并增加无功储备这一目标。

本文在总结前人研究成果上, 运用电压稳定裕度指标选取无功补偿点, 并提出追寻电压稳定裕度指标 I_{vs} 最小和全年多种负荷方式下投资、网损的综合费用最小的无功规划模型。将纵横交叉 (Crisscross Optimization, CSO) 算法应用到求解无功规划优化问题, 并引进纵交叉微算子和将可使粒子在整个可行解空间中搜索全局最优解的量子粒子群算法与CSO算法结合, 提高粒子的多样性和算法搜索能力, 通过IEEE-30节点系统的仿真, 验证了该算法对于无功规划优化是有效可行的。

无功规划数学模型

1. 电压稳定裕度指标

文献[13]从某一负荷节点出发将系统等效为戴维南等效模型, 并将系统阻抗模值与该负荷等效阻抗模值之比定义为电压稳定裕度指标。当这一比值大于某一标准值时, 系统电压可能出现不稳定。以负荷节点 i 为例, 其计算方法如下。

关键词/Keywords

无功规划优化 ·
静态电压稳定 ·
裕度指标 ·
纵横交叉 (CSO) 算法 ·
量子粒子群算法 ·

广东省自然科学基金项目 (S2013040013776)。

1) 输入数据, 形成节点导纳矩阵, 求逆获得节点阻抗矩阵, 其对角线元素 $Z_{bus,i,i}$, 即为该节点 i 的戴维南等效阻抗。

2) 用潮流计算得到节点 i 的电压 V_i , 通过下式得到节点 i 的阻抗值

$$Z_{load,i} = \frac{V_i^2}{S_i} \quad (1)$$

式中, S_i 、 V_i 为节点 i 的复功率和电压幅值。

3) 计算节点 i 电压稳定裕度指标 I_{vs}

$$I_{vs,i} = \frac{|Z_{bus,i,i}|}{|Z_{load,i}|} \quad (2)$$

由此可看出此电压稳定裕度指标 I_{vs} 值简单易行, 没有过多的复杂计算步骤, 通过矩阵变换即可获得, 大大节约了计算时间, 为其成为目标函数之一提供可能性。

2. 目标函数

本文为计及系统静态电压稳定性、投资和网损综合费用最小, 把这多目标函数通过权重合并成了一个单目标函数, 追寻目标函数的最小值, 即

$$\begin{cases} f_1 = c_1 t_{\max} P_{\text{agoloss}} \times 0.1 \\ f_2 = c_1 t_{\max} P_{\text{nowloss}} \times 0.1 + c_2 \sum_{i \in M} Q_i \\ f_3 = \sum_{i \in M} I_{vs,i} \\ \min F = d_1 \frac{f_2}{f_1} + d_2 \frac{f_4}{f_3} \end{cases} \quad (3)$$

式中, c_1 为电价, 元/kW·h; c_2 为无功补偿设备每年折合投资单价, 万元/Mvar; P_{agoloss} 为进行无功规划前的有功网损, MW; P_{nowloss} 为无功规划后的有功网损, MW; Q_i 为节点 i 处电容器总补偿容量, Mvar; M 为补偿点的集合; f_1 为由于网损引起的运行费用, 万元; f_2 为由于网损和无功补偿装置装设引起的费用, 万元; $t_{\max}$ 为最大负荷损耗小时数; $I_{vs,i}$ 为节点 i 的电压稳定裕度指标值, 此值本质为阻抗模的比值; f_3 为无功规划前各补偿点电压稳定裕度总和值; f_4 为无功规划后各补偿点电压稳定裕度总和值, f_4 计算方式与 f_3 一样; d_1 和 d_2 为权重系数, 且 $d_1 + d_2 = 1$, 通过调整权重系数可以追寻经济性和电压稳定这两个不同的侧重点; F 为目标函数值。

3. 约束条件

1) 潮流方程约束为

$$\begin{cases} P_{Gi} = P_{Li} + V_i \sum_{j \in N_i} V_j (G_{ij} \cos \theta_{ij} + B_{ij} \sin \theta_{ij}) \\ Q_{Gi} = Q_{Li} - Q_{Ci} + V_i \sum_{j \in N_i} V_j (G_{ij} \sin \theta_{ij} - B_{ij} \cos \theta_{ij}) \end{cases} \quad (4)$$

式中, P_{Gi} 、 P_{Li} 分别为 i 的有功注入和有功负荷; Q_{Gi} 、 Q_{Ci} 、 Q_{Li} 为节点 i 的无功注入、无功补偿容量及无功负荷; G_{ij} 、 θ_{ij} 、 B_{ij} 分别为节点 i 、 j 之间的电导、电压相角差和电纳; V_i 、 V_j 为节点 i 、 j 的电压。

2) 变量约束方程。状态变量的不等式约束为

$$\begin{cases} V_{i\min} \leq V_i \leq V_{i\max} \\ Q_{Gj\min} \leq Q_{Gj} \leq Q_{Gj\max} \end{cases} \quad (5)$$

式中, $V_{i\min}$ 、 $V_{i\max}$ 分别为节点电压上下限; $Q_{Gj\min}$ 、 $Q_{Gj\max}$ 分别为发电机无功输出功率上下限。

控制变量的不等式约束

$$\begin{cases} V_{Gj\min} \leq V_{Gj} \leq V_{Gj\max} \\ Q_{Ci} \leq Q_{Ci\max} \\ T_{i\min} \leq T_i \leq T_{i\max} \end{cases} \quad (6)$$

式中, $V_{Gj\min}$ 、 $V_{Gj\max}$ 分别为PV节点电压上下限; $Q_{Ci\max}$ 为第 i 个待选补偿节点所允许安装电容器组的容量上限; $T_{i\min}$ 、 $T_{i\max}$ 为分别为第 i 个变压器电压比上下限。

改进型CSO优化算法

CSO算法是一种基于种群的随机搜索算法, 种群则由粒子组成, 粒子包含全部变量信息。为方便说明改进型CSO算法, 本节使用如下概念。

改进型CSO算法的种群用矩阵 B 表示, 矩阵中每一行用 $B(i)$ 表示, 其代表问题的一个解, 矩阵中每一个元素为 $B(i,j)$; 种群规模 M 即矩阵的行数, 解空间维数 D 即矩阵列数。使用横向交叉算子、纵向交叉粗算子、纵向交叉微算子和量子粒子群算法搜索算子得到的解分别用 MS_{hc} 、 MS_{rvc} 、 MS_{tvc} 和 MS_q 表示; 横向交叉算子、纵向交叉粗算子、纵向交叉微算子和量子粒子群算法搜索算子经过竞争算子后得到的解分别表示为 DS_{hc} 、 DS_{rvc} 、 DS_{tvc} 和 DS_q 。

改进型CSO算法流程可简单总结步骤如下:

- 1) 初始化种群。
- 2) 行横向交叉算子后进入竞争算子。
- 3) 执行纵向交叉粗算子后进入竞争算子。
- 4) 执行纵向交叉微算子后进入竞争算子。
- 5) 执行量子粒子群算法搜索算子后进入竞争算子。
- 6) 终止条件。如果达到迭代次数或者满足其他已知边界条件, 算法结束, 输出结果; 否则转入步骤2)。

下面分别介绍初始化种群、横向交叉算子、纵向交叉粗算子、纵向交叉微算子、量子粒子群算法搜索算子和竞争算子的具体执行步骤。

1. 初始化种群

本文发现原始CSO算法初始化时, 其粒子分布不均匀, 对此算法应用于无功规划优化收敛的速度和效果会产生影响。本文引入混沌映射, 使其产生在解空间中分布较均匀的初始解。混沌映射的数学表达式为

$$x_{n+1} = x_n u(1 - x_n) \quad u \in [0, 4], x \in [0, 1] \quad (7)$$

具体初始种群构造流程如下: 首先生成一行随机数, 列数取值为种群变量大小即维数, 随机数值如式(7)中 x 的取值范围; 然后通过对应维的值构造一列混沌映射值, 行数取值为种群大小; 最后反归一化即乘以每一维的上下限差值。

2. 横向交叉算子

假设父代个体 $B(i)$ 和 $B(j)$ 参与横向交叉算子, 那么其子代使用如下公式得出, 即

$$MS_{hc}(i, d) = r_1 B(i, d) + (1 - r_1) B(j, d) + c_1 [B(i, d) - B(j, d)] \quad (8)$$

$$MS_{hc}(j, d) = r_2 B(j, d) + (1 - r_2) B(i, d) + c_2 [B(j, d) - B(i, d)] \quad (9)$$

式中, r_1 、 r_2 为0~1之间的随机数; c_1 、 c_2 为-1~1之间的随机数; $MS_{hc}(i, d)$ 和 $MS_{hc}(j, d)$ 分别为 $B(i, d)$ 和 $B(j, d)$ 通过横向交叉算子产生的子代。

横向交叉算子的具体流程: 首先需要对矩阵 DS_q (在第一次迭代中输入的是初始化的种群) 中的个体进行配对, 随机抽取矩阵 DS_q 的两个粒子, 例如被抽中的个体如 $B(no_1)$ 和 $B(no_2)$ 通过式(7)和式(8)产生其后代 $MS_{hc}(no_1)$ 和 $MS_{hc}(no_2)$, 抽取共

进行 $M/2$ 次, 每次抽取的行数不同, 矩阵每个行数都必须参加计算且只有一次被选中。进行横向交叉的概率设 p_1 为1。因为粒子总是两两配对, 所以种群数量 M 设置为偶数。横向交叉得到种群 MS_{hc} 后, 种群 MS_{hc} 会通过竞争算子与父代种群 DS_q 进行比较, 只有具有更好的适应度的粒子会被保留下来, 保存到下一代的种群 DS_{hc} 中。

3. 纵向交叉粗算子

纵向交叉粗算子是所有个体都会参与自身个体中不同维之间进行的交叉操作。假设 d_1 维和 d_2 维是个体 i 进行纵向交叉操作的维度, 其子代 $MS_{rvc}(i)$ 使用式(10)产生

$$MS_{rvc} = rB(i, d_1) + (1 - r)B(i, d_2) \quad (10)$$

$$i \in N(1, M), d_1, d_2 \in N(1, D)$$

式中, $r \in U(0, 1)$; $MS_{rvc}(i, d_1)$ 为 $B(i, d_1)$ 和 $B(i, d_2)$, 即 $DS_{hc}(i, d_1)$ 和 $DS_{hc}(i, d_2)$ 产生的“子代”。

纵向交叉粗算子的具体流程: 首先需要对父代种群即经过横向交叉算子和竞争算子后保留下来的解 DS_{hc} 进行归一化操作, 即每一维的值通过除以其上下限的差值化为0~1之间的值。然后随机抽取归一化后的 DS_{hc} 的两列数据, 例如 d_1 、 d_2 列, 然后通过式(10)计算产生 MS_{rvc} 的 d_1 列数据, 共抽取 $D/2$ 次, 每次抽取的列数不相同, 且矩阵的每个列数都必须参加计算且只有一次被选中。其实际只改变了 DS_{hc} 的 $D/2$ 列数据。然后对矩阵进行反归一化处理。纵向交叉粗算子操作的概率 p_2 通常被设置在0.2~0.8之间。纵向交叉粗算子操作完成后, 竞争算子将其所得的矩阵 MS_{rvc} 和其父代种群进行竞争操作, 同样, 只有拥有更好的适应度的粒子会在竞争中被保留, 进入矩阵 DS_{rvc} , 进入下一次迭代。

4. 纵向交叉微算子

由于纵交叉粗算子改变了 DS_{hc} 的 $D/2$ 列数据, 所以当只有一维数据陷入局部最优时, 纵向交叉也会改变其他列的数据, 这使得种群其他本已全局最优的维数数值改变, 不利于最优点的搜索。本文仿照纵向交叉粗算子, 引入纵向交叉微算子, 纵向交叉微算子只改变一列数据。流程如下: 把种群先进行归一化处理, 然后随机抽取种

群的两列数据,如 d_1 、 d_2 列,通过式(10)计算产生子代种群 d_1 列数据。并对矩阵进行反归一化处理得 MS_{tvc} 。纵向交叉微调操作完成后,竞争算子将其所得的矩阵和其父代种群进行竞争操作得 DS_{tvc} 。同样,只有拥有更好适应度的粒子会在竞争中被保留。纵向交叉微调算子操作的概率 p_3 为1。

5. 量子粒子群算法搜索算子

量子粒子群算法搜索算子由量子粒子群算法改变而来。其计算公式如下

$$\begin{cases} m_{best}(t) = \frac{1}{M} \sum_{i=1}^M p_i(t) \\ = \left\{ \frac{1}{M} \sum_{i=1}^M p_{i1}(t), \frac{1}{M} \sum_{i=1}^M p_{i2}(t), \dots, \frac{1}{M} \sum_{i=1}^M p_{id}(t) \right\} \\ p_{id}(t+1) = \phi p_{id}(t) + (1-\phi) p_{gd}(t) \\ X_{id}(t+1) = p_{id}(t) \pm \beta |m_{best}(t) - \theta p_{best}(t)| \ln \frac{1}{u} \end{cases} \quad (11)$$

式中, M 为种群中粒子数目; D 为粒子维数; ϕ 为 $[0,1]$ 上服从均匀分布的量子粒子学习因子; $m_{best}(t)$ 为第 t 次迭代的种群中全部粒子(即 DS_{tvc})的平均最佳位置; $p_i(t)$ 、 $p_g(t)$ 和 $p_{id}(t)$ 分别为第 t 次迭代时第 i 个粒子的当前最佳位置(p_{best})、全局最优位置(g_{best})和两个最优位置之间的随机点; $X_{id}(t+1)$ 即为第 $t+1$ 代 MS_q ; β 为收缩扩张系数,一般按照下式取值

$$\beta = m - (m - n)T / \text{MaxTimes} \quad (12)$$

式中,随着迭代线性地从 m 递减到 n ,通常取 $m=0.9$, $n=0.5$; MaxTimes 为最大迭代次数。需要注意的是原始量子粒子群算法^[14-15]迭代的是粒子 $x(t)$,本文量子粒子群算法搜索算子迭代的是 p_{best} ,即 DS_{tvc} 。这是由竞争算子这种竞争机制决定的。 θ 为 $[0,1]$ 上服从均匀分布的扰动因子。扰动因子发生的概率如下式所示

$$T / \text{Maxtimes} \geq \text{rand} \quad (13)$$

量子粒子群算法搜索算子以不同于原始CSO算法迭代方式产生粒子,扰动因子 θ 随着迭代次数增加扰动发生概率增加,这两者增加了种群的多样性,并且增强粒子跳出局部最优的能力,使改进型CSO有更快的搜索速度和更强的搜索准确度。

6. 竞争算子

竞争算子是为子代种群和父代种群提供一个相互竞争的机会,以横向交叉算子为例,子代粒子只有具有比父代粒子更好的适应度值时,才能取代父代粒子进入迭代。竞争算子使种群总是保持在当前最佳位置,而每个算子的父代都是当前对应粒子的最优值,保证种群朝着更好的方向发展。

无功规划步骤

- 1) 输入数据,计算系统潮流,输出各个负荷节点的电压稳定裕度指标 I_{vs} ,并排序。
- 2) 将负荷节点 I_{vs} 指标值大于某一标准值的点作为无功补偿规划点。
- 3) 用混沌映射生成初始种群。
- 4) 用改进型CSO算法计算,寻找待规划点的无功容量最优值。
- 5) 输出规划结果。各个负荷节点的电压稳定裕度指标 I_{vs} 、无功规划容量。

算例分析

(1) 规划效果分析

将本文提出的规划方法运用于Matlab软件matpower文件包的IEEE-30节点系统,6台发电机,20个负荷节点,电压设置为0.97~1.07 V。每个节点最大规划电容量为50 Mvar。本文参数 $t_{\max}^{[10]}$ 设置为8 760 h,电价 c_1 为0.5元/kW·h; c_2 为0.3万元/Mvar;权重系数 d_1 为0.4, d_2 为0.6。 p_1 为1, p_2 为0.4, p_3 为1。种群规模为50,迭代次数为200。该系统原网损为2.444 MW。

为寻找IEEE-30系统电压薄弱节点,本文计算了各个负荷节点的 I_{vs} 指标值,并列举了12个负荷点 I_{vs} 的值,如表1所示。

表1 负荷节点 I_{vs} 指标值

节点	8	7	21	12	17	24
I_{vs}	2.809 4	1.648 1	1.258 7	0.831	0.669 2	0.665 9
节点	19	30	15	4	14	10
I_{vs}	0.635 9	0.631 9	0.530 9	0.495 9	0.394	0.380 6

负荷节点的 I_{vs} 指标值越大电压稳定性就越差,为留有一定裕度本文选择表1中大于0.5的前9个节

点作为无功规划点，并使用改进型CSO算法对该节点系统进行无功规划计算，规划结果如表2所示。

表2 IEEE-30节点系统规划结果

	节点	下限	上限	规划前	规划后
发电机	2	0.97	1.07	1	1.027 1
	13	0.97	1.07	1	1.014 4
	22	0.97	1.07	1	1.024 6
	(pu)	23	0.97	1	1.036 4
	27	0.97	1.07	1	1.036 8
负荷节点	8	0	50	0	28.889 9
	7	0	50	0	10.265 9
	21	0	50	0	15.436 6
	12	0	50	0	28.647 1
	17	0	50	0	9.262 3
	24	0	50	0	8.175 1
	19	0	50	0	5.223 4
	30	0	50	0	3.357 9
容量/Mvar	15	0	50	0	3.083 6

通过无功规划后发电机无功出力在允许范围内，所有节点电压合格，网损从2.444 MW降到1.838 5 MW，降幅达24.77%，全年运行综合费用从1 070.5万元降到838.959万元，降幅达21.62%，系统运行更加经济。而且规划节点的 I_{vs} 指标都有大幅度的减小，提高了系统电压稳定性，具体结果如表3所示。

表3 规划节点的 I_{vs} 指标对比

节点	8	7	21	12	17
$I_{vs}(\text{ago})$	2.809 4	1.648 1	1.258 7	0.831	0.669 2
$I_{vs}(\text{now})$	0.293	0.166 8	0.136 9	0.088 3	0.069 1
节点	24	19	30	15	
$I_{vs}(\text{ago})$	0.665 9	0.635 9	0.631 9	0.530 9	
$I_{vs}(\text{now})$	0.067 1	0.058 2	0.030 7	0.054 4	

(2) 算法对比分析

采用CSO、PSO和改进型CSO算法对IEEE-30节点系统进行无功规划优化仿真，无功规划模型及CSO和改进型CSO参数如上所述，PSO参数设置种群大小为50，迭代次数为200，惯性系数 $w \in (0.4, 0.9)$ ，学习因子 $c_1 \in (2, 1)$ ， $c_2(1, 2)$ ，并采用自适应调整。为避免偶然误差，各算法均独立运行20次，取三种算法的最优结果得出的收敛特性曲线如下图所示。

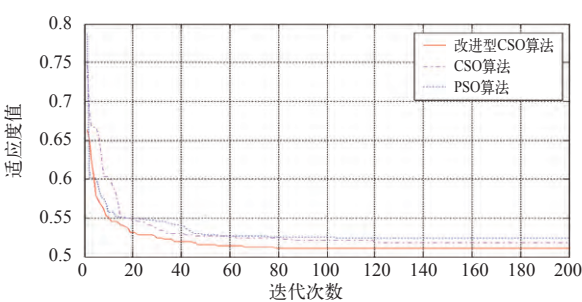


图 收敛特性曲线

从上图4可以看出三种算法都能有效地寻找目标函数最优值，改进型CSO、CSO和PSO算法寻得最优值分别为0.51、0.517 6和0.524 3，系统运行费用分别为831.418 7、853.886 2和865.202万元，费用下降率分别为21.62%、20.2%和19.2%。在迭代过程的开始阶段改进型CSO算法的初始适应度值小于其他两个算法，这正是初始化运用混沌映射和量子粒子群搜索算子快速搜索能力的体现。在迭代了80代左右PSO算法已陷入局部最优，停止不前，而改进型CSO算法具有更强的搜索能力和变异能力使其跳出局部最优找到更优解。因此本文所提改进型CSO算法具有较好的收敛速度和收敛准确度，并且更适合无功规划优化问题。

结束语

通过对IEEE-30节点系统的计算分析表明，本文提出的方法在增加系统电压稳定的同时减少了综合规划费用，改进型CSO算法搜索能力比原始CSO算法和PSO算法更强，提高了搜索速度和准确度。

参考文献

[1] Venkatesh B, Sandasivam G, Khan M A. Optimal reactive power planning against voltage collapse using the successive multi-objective fuzzy LP technique [J]. IEE Proceedings-Generation, Transmission and Distribution, 1999, 146 (4): 343-348.

[2] 王勤, 方鸽飞. 考虑电压稳定性的电力系统多目标无功优化 [J]. 电力系统自动化, 1999, 23 (3): 31-34.

[3] 姚小寅, 孙元章, 王志芳. 电力系统无功源最佳配置地点的研究 [J]. 电力系统自动化, 1999, 23

- (3): 12-15.
- [4] Ajarapu V, Lau P L, Battula S. An optimal reactive power planning strategy against voltage collapse [J]. IEEE Transactions on Power Systems, 1994, 9 (2): 906-917.
- [5] Chebbo A M, Irving M R, Sterling M J H. Reactive power dispatch incorporating voltage stability [J]. IEE Proceedings-Generation, Transmision and Distribution, 1992, 139 (3): 253-260.
- [6] 余健明, 杜刚, 姚李孝. 结合灵敏度分析的遗传算法应用于配电网无功补偿优化规划 [J]. 电网技术, 2002, 26 (7): 46-49.
- [7] 熊曼丽. 提高电压稳定的多目标优化模型及其神经网络算法 [J]. 电力系统自动化, 1996, 20 (1): 27-31.
- [8] 王邵, 周鑫. 应用层次聚类法和蚁群算法的配电网无功优化 [J]. 电网技术, 2011, 35 (8): 161-167.
- [9] 李宏仲, 金义雄, 程浩忠, 等. 基于改进粒子群优化算法并计及静态电压稳定性的电力系统无功规划 [J]. 电网技术, 2005, 29 (22): 50-55.
- [10] 周双喜, 郑智, 鲁宗相, 等. 基于多种群遗传算法的无功规划 [J]. 电力系统及其自动化学报, 2007, 19 (6): 66-71.
- [11] 林广明, 欧阳森, 曾江, 等. 改进灾变遗传算法在无功优化规划中的应用 [J]. 电网技术, 2010 (4): 128-133.
- [12] 曲绍杰, 王绍然, 刘明波, 等. 基于互补内点法的多目标静态电压稳定约束无功规划 [J]. 电力系统保护与控制, 2010 (23): 49-54, 86.
- [13] Manjure D P, Makra E B. Steady state stability assessment using the bus impedance matrix [C]. 2001 Large Engineering Systems Conference on Power Engineering. LESCOPE'01, 2001: 153-157.
- [14] Sun J, Feng B, Xu W B. Particle swarm optimization with particles having quantum behavior [C]. Proceedings of 2004 Congress on Evolutionary Computation, 2004: 325-331.
- [15] 陈道君, 龚庆武, 金朝意, 等. 基于自适应扰动量子粒子群算法参数优化的支持向量回归机短期风电功率预测 [J]. 电网技术, 2013 (4): 974-980.

(收稿日期: 2015-01-20) **EA**

(上接第31页)

参考文献

- (10): 30-33.
- [6] 彭辉, 黄亦农, 王茂海. 配电网中三相不平衡负荷补偿 [J]. 电力自动化设备, 2002, 22 (1): 32-34.
- [7] Lee S-Y. Reactive power compensation and load balancing for unbalanced three-phase four-wire system by a combined system of an SVC and a series active filter [J]. IEEE Proc. -Electr. Power Appl., 2000, 147 (6): 563-571.
- [8] 吴杰. 中性点有效接地系统不平衡负载的平衡化补偿理论研究 [J]. 电工技术杂志, 2003 (9): 73-75.
- [9] 单铁铭, 杨仁刚. 不平衡电流无功补偿方法研究 [J]. 电力自动化设备, 2004, 24 (12): 26-29.
- [1] 张五一, 张言滨, 刘华伟. 配电网三相负荷不对称的线损分析 [J]. 电力科学与工程, 2007, 32 (1): 16-18.
- [2] 林海雪. 电力系统的三相不平衡 [M]. 北京: 中国电力出版社, 1998.
- [3] 张永军, 孟文博, 赵积春, 等. 三相不平衡与无功功率综合补偿系统研究 [J]. 电工技术杂志, 2004 (12): 63-65.
- [4] 张明江, 纪延超, 顾强. 三相不平衡与无功功率的综合补偿方法 [J]. 电网技术, 2008, 32 (1): 20-23.
- [5] 李心广, 赖声礼, 秦华标, 等. 电网的无功及三相不平衡综合补偿研究 [J]. 电网技术, 2001, 25

(收稿日期: 2015-03-16) **EA**