

基于改进 PSO 算法的短期发电计划研究

蒋秀洁¹,熊信良²,吴耀武²

(1. 三峡大学 电气信息学院,湖北 宜昌 443002;

2. 华中科技大学 电气工程学院,湖北 武汉 430074)

摘要:介绍了粒子群优化算法 PSO(Particle Swarm Optimization),并针对短期发电计划中的优化问题提出了一种改进 PSO 算法,将表示机组开停机状态的离散变量转换为 0~1 范围内的连续变量,与机组出力一起进行 PSO 优化搜索,然后再利用就近取整函数“round”将其转换成整数变量。详细描述了应用改进 PSO 算法求解机组优化启停问题的具体步骤。将该方法应用于 10 机系统,实验结果表明该改进 PSO 算法用于短期发电计划是可行的。

关键词:电力系统;粒子群算法;短期发电计划

中图分类号: TM 715

文献标识码: A

文章编号: 1006-6047(2005)03-0034-04

0 引言

电力系统短期发电计划是在满足系统安全和质量的条件下确定一定调度周期内各时段系统机组的运行状态,并相应地确定各机组的运行出力。

短期发电计划问题是一个规模很大的多阶段、多约束的非线性混合整数规划问题。为此,人们提出了各种方法^[1~8]。粒子群优化 PSO(Particle Swarm Optimization)算法^[9,10]是一种基于群智能方法的演化计算方法,可用于求解大量非线性、不可微和非凸的复杂优化问题。PSO 优化技术能在较短计算时间内产生高质量的解,比其他随机搜索方法具有更稳定的收敛特性,但其缺点是易陷入局部极小点,搜索精度不高。为避免陷入局部最优,本文将改进的 PSO 优化技术用于求解短期发电计划问题,并用 Matlab 进行了仿真计算,算例表明了该算法的有效性和可行性。

1 数学描述

设调度周期为日,将之分为 H 个时段,系统中机组或等效机组数为 G ,各时段系统总负荷为 D_t ,则机组优化组合的数学描述如下。

1.1 目标函数

短期发电计划的目标是在满足各类约束的条件下合理安排机组启停,并经济分配负荷,使整个系统在 H 个时段中各机组的总耗量最小,目标函数为

$$\min F(U_{it}, P_{it}) = \sum_{i=1}^H \sum_{t=1}^G [F_{it}(P_{it}) + (1 - U_{it-1})S_{it}]U_{it} \quad (1)$$

式中 U_{it} 为机组 i 在 t 时段运行状态变量,仅取 0,1 两个值, $U_{it}=1$ 表示开机, $U_{it}=0$ 表示停机; P_{it} 为机组 i 在 t 时段的发电有功功率, $U_{it}=0$ 时, $P_{it}=0$; $F_{it}(P_{it})$ 为机组 i 运行耗量, $F_{it}(P_{it})=a_i P_{it}^2 + b_i P_{it} + c_i$,

a_i, b_i, c_i 为运行耗量特性参数; S_{it} 为机组 i 在 t 时段的启动耗量,它与停机时间 τ 的长短有关, $S_{it}=S_{0i}+S_{it}(1-e^{-\tau/\tau_i})$, S_{0i} 表示汽机的启动耗量,一般为常数, S_{it} 为锅炉由完全冷却状态启动时的启动耗量, τ_i 为锅炉冷却时间常数。

1.2 约束条件

a. 功率平衡约束

$$\sum_{i=1}^G U_{it} P_{it} = D_t + P_{Lt} \quad t=1, 2, \dots, H \quad (2)$$

式中 P_{Lt} 为 t 时段系统的网损。

b. 机组技术出力约束

$$P_{imin} \leq P_{it} \leq P_{imax} \quad t=1, 2, \dots, H \quad (3)$$

式中 P_{imin}, P_{imax} 分别为机组 i 的出力最小、最大值。

c. 旋转备用约束

$$\sum_{i=1}^G U_{it} P_{imax} \geq R_t + D_t \quad t=1, 2, \dots, H \quad (4)$$

式中 R_t 为旋转备用功率。

d. 启停约束

启停次数约束为

$$\sum_{i=1}^H |U_{it} - U_{it-1}| \leq M_i \quad i=1, 2, \dots, G \quad (5)$$

最小停运时间约束为

$$(U_{it} - U_{it-1}) \sum_{j=t-\beta_{1i}H/24}^{t-1} (1 - U_{ij}) \geq \beta_{1i} \quad t=1, 2, \dots, H; i=1, 2, \dots, G \quad (6)$$

最小运行时间约束为

$$(U_{it-1} - U_{it}) \sum_{j=t-\beta_{2i}H/24}^{t-1} (1 - U_{ij}) \geq \beta_{2i} \quad t=1, 2, \dots, H; i=1, 2, \dots, G \quad (7)$$

式中 $M_i, \beta_{1i}, \beta_{2i}$ 分别为机组在调度周期内最大允许启停次数、最小连续停运小时数和最小连续运行小时数。

根据式(1),当要解决机组组合问题时必须首先决定机组的开停机计划。

2 PSO 算法介绍

2.1 标准 PSO 算法^[9,10]

基于群体中的个体(如粒子或主体)行为发展了 PSO 算法。算法采用速度-位置模型,即 PSO 算法在允许范围内初始化为一群随机粒子(潜在解),每个粒子都有一个速度决定其飞行方向和距离,在每一次迭代中通过跟踪 2 个极值更新自己;粒子本身迄今为止所找到的个体极值 $\mathbf{P}_{bestj}$ 和整个种群迄今为止所找到的全局极值 $\mathbf{G}_{best}$ 。所有粒子的优劣由被优化函数所决定的适应值衡量。

设在一个 N 维的搜索空间,在 PSO 优化算法中第 j 个粒子的位置和速度可表示为 $\mathbf{X}_j = (x_{j1}, x_{j2}, \dots, x_{jN})$ 和 $\mathbf{V}_j = (v_{j1}, v_{j2}, \dots, v_{jN})$, 其中 $j = 1, 2, \dots, m, m$ 为群体规模;相应的,第 j 个粒子迄今为止搜索到的最优位置为 $\mathbf{P}_{bestj} = (x_{j1best}, x_{j2best}, \dots, x_{jNbest})$, 整个粒子群迄今为止搜索到的最优位置为 $\mathbf{G}_{best} = (x_{1best}, x_{2best}, \dots, x_{Nbest})$ 。利用这些信息,采用式(8)对 PSO 算法中第 j 个粒子的速度和位置更新:

$$\mathbf{V}_i^{k+1} = \omega \mathbf{V}_i^k + c_1 \text{rand}_1(\mathbf{P}_{besti}^k - \mathbf{X}_i^k) + c_2 \text{rand}_2(\mathbf{G}_{best} - \mathbf{X}_i^k) \quad (8)$$

$$\mathbf{X}_i^{k+1} = \mathbf{X}_i^k + \mathbf{V}_i^{k+1} \quad (9)$$

式中 $\mathbf{V}_j^k$ 是第 k 代粒子 j 的速度, $|\mathbf{V}_j^k| \leq \mathbf{V}_j^{\max}$; ω 是非负数,称作惯性因子或权重系数; c_1 和 c_2 是非负常数,称为学习因子,根据以往的经验一般取 $c_1 = c_2 = 2$; rand_1 和 rand_2 是均匀分布在 $(0, 1)$ 区间的随机数。

迭代终止条件一般选为最大迭代次数或粒子迄今为止搜索到的最优位置满足适应阈值。

在以上参数中, $\mathbf{V}_j^{\max}$ 要合理选择,若 $\mathbf{V}_j^{\max}$ 选得太大,粒子可能错过最优解,若 $\mathbf{V}_j^{\max}$ 太小,粒子可能陷入局部搜索。在以前的 PSO 优化算法实践中, $\mathbf{V}_j^{\max}$ 一般限定为每个粒子在每一维度上位置的允许变化范围 $(\mathbf{X}_j^{\max} - \mathbf{X}_j^{\min})$ 的 10%~20%。

2.2 自适应 PSO 算法^[11]

经研究发现:惯性权重 ω 对优化性能有很大的影响, ω 较大算法具有较强的全局搜索能力, ω 较小则算法有利于局部搜索。因此,提出自适应 PSO 算法调整 ω 的策略,如果让 ω 随算法迭代的进行而线性地减少,将显著改善算法的收敛性能。设 $\omega_{\max}$ 为最大惯性权重,一般取为 0.9; $\omega_{\min}$ 为最小惯性权重,一般取 0.4; k 为当前迭代次数; K 为算法的总迭代次数。 ω 由下式迭代:

$$\omega = \omega_{\max} - \frac{\omega_{\max} - \omega_{\min}}{K} k \quad (10)$$

将 ω 设置为从 0.9 到 0.4 的线性下降,使得 PSO 算法在开始时搜索较大的区域,较快地确定最优解的大致位置,随着 ω 逐渐减小,粒子的速度减慢,开始精细的局部搜索。这种方法加快了收敛速度,提高了 PSO 算法的性能。

2.3 杂交算子的引入

为了提高粒子群的多样性,文献[12]将进化算法中的交叉操作引入 PSO,提出了混合粒子群算法模型。粒子群中的粒子被赋予一个交叉概率。在每次迭代中,依据交叉概率选取一定数量的粒子放入池中,然后两两随机组合进行交叉操作产生同样数目的孩子粒子,孩子粒子的位置和速度由下式计算:

$$\text{child}_1(\mathbf{X}) = \mathbf{p} * \text{parent}_1(\mathbf{X}) + (1 - \mathbf{p}) * \text{parent}_2(\mathbf{X}) \quad (11)$$

$$\text{child}_2(\mathbf{X}) = \mathbf{p} * \text{parent}_2(\mathbf{X}) + (1 - \mathbf{p}) * \text{parent}_1(\mathbf{X}) \quad (12)$$

式中 $\mathbf{X}$ 是 N 维位置向量; $\mathbf{p}$ 是 N 维均匀分布的随机数向量,其每个分量都在 $[0, 1]$ 内取值。

$$\text{child}_1(\mathbf{V}) = \frac{\text{parent}_1(\mathbf{V}) + \text{parent}_2(\mathbf{V})}{|\text{parent}_1(\mathbf{V}) + \text{parent}_2(\mathbf{V})|} |\text{parent}_1(\mathbf{V})| \quad (13)$$

$$\text{child}_2(\mathbf{V}) = \frac{\text{parent}_1(\mathbf{V}) + \text{parent}_2(\mathbf{V})}{|\text{parent}_1(\mathbf{V}) + \text{parent}_2(\mathbf{V})|} |\text{parent}_2(\mathbf{V})| \quad (14)$$

式中 $\mathbf{V}$ 为 N 维位置向量。

交叉操作继承了双亲粒子的优点,加强了跳出局部最优的能力。实验结果显示,混合粒子群算法的收敛速度比较快,搜索精度也相对比较高。

3 改进 PSO 算法的应用

传统的短期发电计划问题由两个优化子问题组成,即机组方案的确立问题和经济分配问题。本文应用 PSO 算法解决实际电力系统中短期发电计划问题,得到了一个有效且高质量的解。

3.1 粒子的表示

作为一个有效率的进化方法,如何将问题的参数表示为基因链是很关键的。由于机组发电计划的控制变量不仅有表示机组状态的 U_{it} (机组 i 在 t 时段的状态开‘1’、停‘0’),还要表示每一时段机组出力的 P_{it} ,即位置和速度变量中分别有两套参数,因此在采用 PSO 算法时,位置变量由两个变量组成: $\mathbf{U} = [U_{11}, U_{12}, \dots, U_{1H}, \dots, U_{G1}, \dots, U_{GH}]$; $\mathbf{P} = [P_{11}, P_{12}, \dots, P_{1H}, \dots, P_{G1}, \dots, P_{GH}]$ 。在本文中应用 Matlab 进行求解时,将 $\mathbf{U}$ 和 $\mathbf{P}$ 用两个二维矩阵变量表示,其维数均为 $H \times G$ 。在矩阵变量 $\mathbf{U}$ 中,每台机组的状态(0 或 1)为一个基因,每台机组在一个调度周期内的状态组成一条子链,假设系统中有 G 台机组,则 G 条子链组成一个个体(粒子)。因此,一个粒子表示一个调度周期内所有机组状态,矩阵行向量表示一台机组在一个调度周期内 H 个时段的状态,列向量表示每个时段 G 台机组状态。如粒子群规模为 m ,则粒子群维数为 $G \times H \times m$ 。同样的,在矩阵变量 $\mathbf{P}$ 中,行向量表示每台机组在一个调度周期每个时段的出力,列向量表示每一时段 G 台机组的出力。每个粒子的位置矩阵表示如下:

$$\mathbf{U} = \begin{bmatrix} U(1,1) & \dots & U(1,H) \\ \vdots & & \vdots \\ U(G,1) & \dots & U(G,H) \end{bmatrix}$$

$$\mathbf{P} = \begin{bmatrix} P(1,1) & \dots & P(1,H) \\ \vdots & & \vdots \\ P(G,1) & \dots & P(G,H) \end{bmatrix}$$

相应地,其速度变量也用两个二维变量 VU 和 VP 表示,维数也为 $H \times G$,即

$$VU = \begin{bmatrix} vu(1,1) & \cdots & vu(1,H) \\ \vdots & & \vdots \\ vu(G,1) & \cdots & vu(G,H) \end{bmatrix}$$

$$VP = \begin{bmatrix} vp(1,1) & \cdots & vp(1,H) \\ \vdots & & \vdots \\ vp(G,1) & \cdots & vp(G,H) \end{bmatrix}$$

在每次迭代中求得的 P_{it}^k ,首先利用式(4)以及式(15)对其进行检验:

$$\sum_{i=1}^G U_{it} P_{imin} \leq D_t + P_{L,t} \quad t=1,2,\dots,H \quad (15)$$

若 P_{it}^k 不满足上述条件,则解一定为不可行解。

3.2 对离散变量 U_{it} 的处理

在目标函数式(1)中,控制变量不仅有表示机组 i 在 t 时段的开停机状态的离散变量 U_{it} ,而且还有表示机组 i 在 t 时段出力的连续变量 P_{it} ,因此,在应用改进 PSO 算法解决发电计划问题时必须对离散变量进行处理。应用 Matlab 进行求解时,由于 $U_{it}^k=0$ 或 1,因此,将 U_{it}^k 松弛为 $[0,1]$ 范围内的连续变量,然后再对 U_{it}^k 向最近整数取整,即将式(1)~(7)以及式(15)中的 U_{it}^k 用 $\text{round}(U_{it}^k)$ 代替。

3.3 适应度函数

在 PSO 算法中根据适应度函数值评价每个个体的优劣。因此,首先根据目标函数确定一个适应度函数,同时将系统功率平衡约束作为罚函数考虑其中。定义适应度函数如下:

$$f = F(U_{it}, P_{it}) + \sum_{t=1}^H [A f_t \times g_t(U_{it}, P_{it})] \quad (16)$$

式中 $F(U_{it}, P_{it})$ 是系统机组的发电总费用,由式(1)决定; $g(U_{it}, P_{it})$ 是功率平衡约束条件,由式(17)决定。

$$g(U_{it}, P_{it}) = \left[\sum_{i=1}^G \text{round}(U_{it}) P_{it} - D(t) \right]^2$$

$$t=1,2,\dots,H \quad (17)$$

为了限制粒子群中每个粒子的适应度函数值在可行范围内,在评价每个个体的适应度值前必须使机组出力满足式(3)(4)和(15),如果一个粒子满足所有约束,则它是一个可行解,同时其 f 值也比较小。

3.4 改进 PSO 算法计算步骤

将上述各问题进行处理后就可以利用杂交 PSO 算法求解。用改进 PSO 算法求解机组优化启停问题可按下述步骤进行。

a. 输入原始数据,包括粒子种群数 m ,迭代总次数 K ,交叉概率 P_c ,学习因子 c_1 和 c_2 ,权重因子 ω ,输入各机组耗量特性参数、系统网络参数及各时段系统负荷;定义解空间以及边界并初始化算法各参数,令迭代次数 $k=1$ 。

b. 在解空间范围内(即满足式(3)(4)和(11))随机产生 m 个粒子,形成初始粒子群的位置 U_j^1 和 P_j^1 以及速度 VU_j^1 和 VP_j^1 。

c. 根据式(16)(17)计算各粒子的适应度值,并设个体极值初始值 $PU_{bestj} = U_j^1, PP_{bestj} = P_j^1$,比较初始粒子群的适应度函数值,将适应度值最好的设为 GU_{best} 和 GP_{best} ,其中 PU_{bestj}, PP_{bestj} 分别为第 j 个个体迄今为止机组状态和机组出力适应度值最优的基因链; GU_{best}, GP_{best} 分别为粒子群迄今为止机组状态和机组出力适应度值最优的基因链。

d. 按式(8)(10)调整每个粒子的速度 VU_j^{k+1}, VP_j^{k+1} 和 ω ,并限制 $VU_{min} < VU_j^{k+1} < VU_{max}, VP_{min} < VP_j^{k+1} < VP_{max}$,描述如下:

若 $VU_j^{k+1} > VU_{max}$, 则 $VU_j^{k+1} = VU_{max}$

若 $VU_j^{k+1} < VU_{min}$, 则 $VU_j^{k+1} = VU_{min}$

若 $VP_j^{k+1} > VP_{max}$, 则 $VP_j^{k+1} = VP_{max}$

若 $VP_j^{k+1} < VP_{min}$, 则 $VP_j^{k+1} = VP_{min}$

其中 VU_{max} 元素值都为 0.15; $VP_{max} = 0.15(P_{max} - P_{min})$,即速度最大值设为相应的位置变量变化范围的 15%。

e. 根据式(9)调整粒子的位置 U_j^{k+1} 和 P_j^{k+1} , U_j^{k+1} 和 P_j^{k+1} 必须满足各种约束。

f. 对已产生的种群以交叉概率 P_c 选择粒子形成子种群,从子种群中随机的两两选取个体 $X_r = [U_r, P_r]$ 和 $X_s = [U_s, P_s]$,按式(11)~(14)进行交叉操作,产生两个新个体 X'_r 和 X'_s 。

g. 根据式(16)(17)计算各粒子的适应度值,将每个粒子的当前位置 U_j^{k+1} 和 P_j^{k+1} 的适应度值和粒子自身以前最好位置 PU_{bestj}, PP_{bestj} 的适应度值相比较,如果当前位置适应度值优于 PU_{bestj}, PP_{bestj} 的适应度值,则更新 PU_{bestj}, PP_{bestj} ;再将各个粒子的 PU_{bestj}, PP_{bestj} 的适应度值与全局最优位置 GU_{best}, GP_{best} 的适应度值进行比较,并取最优者更新 GU_{best} 和 GP_{best} 。

h. 如果迭代次数达到最大迭代数 K ,则转入 **i**;否则转入 **d**。

i. 输出 GU_{best} 和 GP_{best} ,并且计算输出的最优值 $F(GU_{best}, GP_{best})$ 。

4 算例

为了验证提出的改进 PSO 算法的可行性和有效性,本文对具有 10 台发电机组的算例系统进行了计算,机组特性参数参考文献[4],负荷曲线如图 1 所示。取参数 $c_1=c_2=2, \omega_{max}=0.9, \omega_{min}=0.4$,交叉概率 $P_c=0.5$,粒子群规模 $m=100$,最大迭代数 $K=200$,计算周期分为 24 个时段。

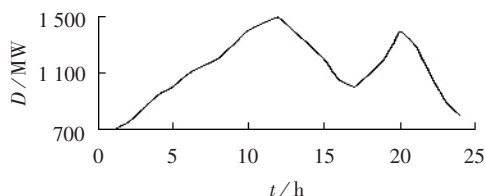


图 1 系统负荷曲线
Fig.1 System load curve

应用本文提出的进化方法求得最优机组组合方案及经济负荷分配见表 1,运行费用为 \$557454, 各台机组的启动费用如表 2 所示,总的启动费用为 \$3 409.281,总发电费用 F 为 \$560 863.3。

表 1 最优机组组合方案
Tab.1 Best unit commitment schedule
and economic dispatch

时段	机组编号										D/MW	成本/ \$
	1	2	3	4	5	6	7	8	9	10		
1	455	220	0	0	25	0	0	0	0	0	700	14 280
2	455	270	0	0	25	0	0	0	0	0	750	15 064
3	455	370	0	0	25	0	0	0	0	0	850	16 809
4	455	450	20	0	25	0	0	0	0	0	950	19 243
5	455	455	65	0	25	0	0	0	0	0	1 000	20 086
6	455	455	130	35	25	0	0	0	0	0	1 100	22 455
7	455	455	130	85	25	0	0	0	0	0	1 150	23 301
8	455	455	130	130	30	0	0	0	0	0	1 200	24 170
9	455	455	130	130	130	0	0	0	0	0	1 300	26 204
10	455	455	130	130	162	68	0	0	0	0	1 400	28 788
11	455	455	130	130	162	80	38	0	0	0	1 450	30 603
12	455	455	130	130	162	80	68	0	0	0	1 500	31 716
13	455	455	130	130	162	68	0	0	0	0	1 400	28 788
14	455	455	130	130	130	0	0	0	0	0	1 300	26 204
15	455	455	130	130	30	0	0	0	0	0	1 200	24 170
16	455	455	95	20	25	0	0	0	0	0	1 050	21 607
17	455	455	45	20	25	0	0	0	0	0	1 000	20 763
18	455	455	130	35	25	0	0	0	0	0	1 100	22 455
19	455	455	130	130	30	0	0	0	0	0	1 200	24 170
20	455	455	130	130	162	68	0	0	0	0	1 400	28 788
21	455	455	130	130	130	0	0	0	0	0	1 300	26 204
22	455	455	60	105	25	0	0	0	0	0	1 100	22 453
23	455	420	0	0	25	0	0	0	0	0	900	17 685
24	455	320	0	0	25	0	0	0	0	0	800	15 936

表 2 各机组启动费用
Tab.2 Start-up cost of each unit

机组编号	启动费/ \$	机组编号	启动费/ \$
1	0	6	671.3
2	0	7	520.0
3	1 054.9	8	60.0
4	1 103.1	9	0
5	0	10	0

该方法与文献[14]利用动态规划法以及遗传算法所得到的发电费用 \$565 825 相比,其成本减少了 \$4961.7,即减少了 0.88 %,因此优化质量更好,而且该算法操作更简单、使用更方便。适应度函数(f)收敛图如图 2 所示,从图中可看出该算法的收敛性能稳定。

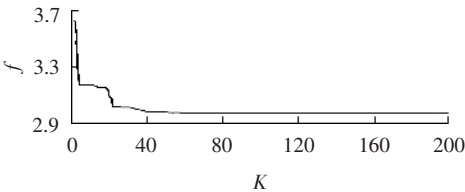


图 2 适应度函数值收敛曲线
Fig.2 Convergence tendency of
the evaluation value

5 结论

本文在研究基本 PSO 算法的基础上,根据电力系统短期发电计划的实际情况提出了一种改进的 PSO 方法,并应用 Matlab 进行仿真计算。该方法对表示机组状态的离散变量进行处理,首先将其转换为 [0,1]之间连续变量,与机组出力一起进行 PSO 优化搜索,然后再应用就近取整函数 round 将其转换成整数变量,实践证明这样的处理并不影响正常搜索。最后,将本文提出的方法应用于 10 机系统,算例结果表明,与动态规划法及遗传算法相比,PSO 有很好的特性,包括高质量的优化解、稳定的收敛性以及有效的计算效率。

参考文献:

[1] 王承民,郭志忠,于尔铿. 确定机组组合的一种改进的动态规划方法[J]. 电网技术,2001,25(5):20-24.
WANG Cheng-min, GUO Zhi-zhong, YU Er-keng. An improved dynamic programming method for determining unit commitment[J]. **Power System Technology**, 2001, 25(5): 20-24.

[2] HABIBOLLAHZADEH H, BUBENKO J A. Application of decomposition techniques to short-term operation planning of hydrothermal power system[J]. **IEEE Trans. on Power Systems**, 1986, 1(1):41-47.

[3] ZHANG F, GALIANA F D. Unit commitment by simulated annealing[J]. **IEEE Trans. on Power Systems**, 1990, 5(1):311-318.

[4] KAZARLIS S A, BAKIRTZIS A G, PETRIDIS V. A genetic algorithm solution to the unit commitment problem[J]. **IEEE Trans. on Power Systems**, 1996, 11(1):83-92.

[5] RAJAN C C A, MOHAN M R, MANIVANNAN K. Neural based Tabu search method for solving unit commitment problem [A]. **Power System Management and Control [C]**. [s. l.]:[s. n.], 2002. 180-185.

[6] SASAKI H, WATANABE M, KUBOKAMA J, et al. A solution method of the unit commitment by artificial neural networks [J]. **IEEE Trans. on Power Systems**, 1992, 7(3):974-981.

[7] LI S, SHAHIDEHPOUR S M, WANG C. Promoting the application of experts systems in short-term unit commitment[J]. **IEEE Trans. on Applied Superconductivity**, 1993, 3(1):286-292.

[8] OUYANG Z, SHAHIDEHPOUR S M. A hybrid artificial neural network- dynamic programming approach to unit commitment[J]. **IEEE Trans. on Power Systems**, 1992, 7(1):236-242.

[9] KENNEDY J, EBERHART R C. Particle swarm optimization[A]. **Proceedings of IEEE International Conference on Neural Networks[C]**. Perth, Australia: [s.n.], 1995. 1942-1948.

[10] EBERHART R C, KENNEDY J. A new optimizer using particle swarm theory [A]. **Proceedings of Sixth International Symposium Micro Machine and Human Science[C]**. Nagoya, Japan: [s.n.], 1995. 39-43.

(上接第 37 页 continued from page 37)

[11] SHI Yu-hui, EBERHART. A modified particle swarm optimizer [A]. **IEEE World Congress on Computational Intelligence** [C]. [s.l.]: IEEE, 1998. 69-73.

[12] 周 驰, 高海兵, 高 亮, 等. 粒子群优化算法 [J]. 计算机应用研究, 2003, (12): 7-12.

ZHOU Chi, GAO Hai-bing, GAO Liang, *et al.* Particle Swarm Optimization (PSO) algorithm [J]. **Application Research of Computer**, 2003, (12): 7-12.

(责任编辑: 李育燕)

作者简介:

蒋秀洁 (1972-), 女, 四川广安人, 讲师, 硕士, 研究方向为电力系统规划方法、配电网自动化及其应用研究 (E-mail: jxiujie@ctgu.edu.cn);

熊信良 (1945-), 男, 湖北浠水人, 教授, 研究方向为电力系统运行分析与规划及电力系统谐波分析;

吴耀武 (1963-), 男, 湖北黄冈人, 副教授, 研究方向为电力系统运行方向与规划。

Improved PSO algorithm and its application
in short-term generation scheduling

JIANG Xiu-jie¹, XIONG Xin-gen², WU Yao-wu²

(1. Sanxia University, Yichang 443002, China; 2. Huazhong University of Science and Technology, Wuhan 430074, China)

Abstract: PSO (Particle Swarm Optimization) algorithm is introduced, and an improved PSO algorithm is proposed for the optimization of short-term generation scheduling. The discrete variables representing the unit status are transformed to continuous variables from zero to one, which together with unit output, are optimized by PSO and finally transformed to integral variables using function 'round'. Its implementation steps are detailed. A ten-machine system is tested with the proposed method, and the results show its feasibility.

Key words: power system; particle swarm optimization; short-term generation scheduling