

文章编号: 1001-0920(2008)08-0857-06

Tent 混沌粒子群算法及其在结构优化决策中的应用

张浩^a, 张铁男^b, 沈继红^c, 李阳^b

(哈尔滨工程大学 a. 自动化学院, b. 经济管理学院, c. 理学院, 哈尔滨 150001)

摘要: 首先对 Tent 混沌序列加以改进, 将其引入粒子群算法中; 然后提出一种基于改进的 Tent 映射的粒子群算法. 采取分阶段更新的优化策略, 使其在搜索初期更具遍历性, 在搜索后期, 通过人为更替最差粒子的速度和位置, 使算法具有更快的收敛速度与更好的全局搜索能力. 构建一种资源配置结构优化模型, 并将改进的 Tent 映射粒子群算法引入资源配置结构优化决策中, 在寻优速度、精度和成功率等方面均显示出良好的优化效果.

关键词: 粒子群算法; 混沌优化; 遍历性; Tent 映射; 结构优化

中图分类号: F224

文献标识码: A

Research on decision-makings of structure optimization based on improved Tent PSO

ZHANG Hao^a, ZHANG Tie-nan^b, SHEN Ji-hong^c, LI Yang^b

(a. College of Automation, b. School of Economics and Management, c. College of Science, Harbin Engineering University, Harbin 150001, China. Correspondent: ZHANG Hao, E-mail: zhaozhhao@126.com)

Abstract: The random function is introduced to Tent map, and the improved Tent map is introduced to particle swarm optimization(PSO). At the initial searching stage, the improved Tent PSO has better ergodicity. At the later stage, the algorithm has faster convergence speed and better global search capability through changing the particles' speed and position. The improved Tent PSO and the standard PSO are compared with each other in the speed, accuracy and successful ratio of searching. The improved Tent PSO is applied to the structure optimization of resources allocation, and the simulation result shows better optimization function.

Key words: Particle swarm optimization; Chaos optimization; Ergodicity; Tent map; Structure optimization

1 引言

粒子群优化算法(PSO)由 Kenney 等人于 1995 年提出, 是一种模拟鸟群和鱼群觅食过程中迁徙和聚集行为的进化算法^[1]. 这种算法的基本设计思想来源于人工生命和进化计算两个方面, 通过个体间的协作与竞争, 实现复杂空间中最优解的搜索. 在 PSO 系统中, 每个备选解被称为一个“粒子”, 多个粒子共存、合作寻优, 每个粒子根据自身“经验”和群体的最佳“经验”在问题空间中向更好的位置“飞行”, 搜索最优解^[2,3]. PSO 运算机理简单, 收敛速度快, 在很多领域已得到广泛应用, 但该算法易过早地陷入局部极值点. 混沌是一种普遍存在的非线性现象, 其行为复杂, 类似随机运动, 具有内随机性、初值敏感性和遍历性等特点^[4]. 将混沌优化思想引入粒

子群优化算法中求解函数优化问题已经越来越多地受到关注, 利用混沌搜索的随机性和遍历性, 能够更好地弥补粒子群算法的缺陷, 实现全局最优, 取得了一定的成果.

本文提出以引入随机函数的方法对 Tent 混沌序列加以改进, 并引入粒子群中, 构建了一种基于改进的 Tent 映射粒子群优化算法. 采用分阶段的策略, 在搜索过程中更新粒子, 使算法具有更高的全局寻优能力, 并对改进的 Tent 映射粒子群算法和标准粒子群算法在寻优速度、精度和成功率等方面进行比较分析. 最后, 本文将改进的 Tent 映射粒子群应用于资源配置结构优化中, 得到了较好的效果.

2 基本粒子群优化算法

PSO 算法的过程表示如下: 假定搜索空间为 n

收稿日期: 2007-12-02; 修回日期: 2008-03-09.

基金项目: 国家自然科学基金项目(70672086).

作者简介: 张浩(1978—), 男, 河北唐山人, 博士生, 从事复杂系统建模与仿真、企业系统演化等研究; 张铁男(1963—), 男, 哈尔滨人, 教授, 博士生导师, 从事企业管理等研究.

维,令 m 表示粒子总数, $X = (x_1, x_2, \dots, x_m)$, 则第 i 个微粒位置向量表示为 $X_i = (x_{i1}, x_{i2}, \dots, x_{in})$, 速度向量表示为 $V_i = (v_{i1}, v_{i2}, \dots, v_{in})$, 粒子经历过的最好位置(即个体极值)记为 $P_i = (p_{i1}, p_{i2}, \dots, p_{in})$, 也称为 p_{best} . 在群体中所有微粒经历过的最好位置(即种群全局极值)用符号 g 表示, 即 $P_g = (p_{g1}, p_{g2}, \dots, p_{gn})$, 也称为 g_{best} .

粒子在每一次迭代找到上述两个极值后,对自己的速度和位置进行如下更新:

$$v_{id}^{(t+1)} = \begin{cases} v_{id}^{(t)} + c_1 r_1 (p_{id}^{(t)} - x_{id}^{(t)}) + c_2 r_2 (p_{gd}^{(t)} - x_{id}^{(t)}), \\ X_{\min} < x_{id}^{(t)} < X_{\max}; \\ 0, \text{ otherwise;} \end{cases} \quad (1)$$

$$x_{id}^{(t+1)} = \begin{cases} x_{id}^{(t)} + v_{id}^{(t+1)}, & X_{\min} < x_{id}^{(t)} + v_{id}^{(t+1)} < X_{\max}; \\ X_{\max}, & x_{id}^{(t)} + v_{id}^{(t+1)} > X_{\max}; \\ X_{\min}, & x_{id}^{(t)} + v_{id}^{(t+1)} < X_{\min}. \end{cases} \quad (2)$$

式中: $d = 1, 2, \dots, n$, $i = 1, 2, \dots, m$, m 为种群规模; t 为当前进化代数; r_1 和 r_2 为分布于 $[0, 1]$ 之间的随机数; c_1 和 c_2 为学习因子或加速常数. 为使粒子速度不致过大,可设置速度上限 V_{\max} , 即当式(1)中 $v_{id} > V_{\max}$ 时, $v_{id} = V_{\max}$; $v_{id} < -V_{\max}$ 时, $v_{id} = -V_{\max}$ ^[5-9]. 式(1)中 $v_{id}^{(t)}$ 为粒子先前的速度; $c_1 r_1 (p_{id}^{(t)} - x_{id}^{(t)})$ 为“认知”部分,表示粒子自身的经验或记忆; $c_2 r_2 (p_{gd}^{(t)} - x_{id}^{(t)})$ 为“社会”部分,表示粒子间的信息共享与相互合作.

PSO 算法的寻优能力主要依靠粒子之间的相互作用和相互影响. 如果从算法中去除粒子之间的相互作用和相互影响,则 PSO 算法的寻优能力就变得非常有限^[10]. 由于标准 PSO 算法寻优依靠的是群体之间的竞争与合作,粒子本身缺乏变异机制,单个粒子一旦受某个局部极值约束后,本身很难跳出这种约束,此时需要借助其他粒子的成功发现. 在算法运行的初始阶段,收敛速度比较快,运动轨迹呈正弦波摆动,但运行一段时间后,速度开始减慢甚至停滞. 当所有粒子的速度几乎为 0 时,粒子群丧失了进一步进化的能力,可以认为算法执行已经收敛. 而在许多情况下,算法并没有收敛到全局极值,甚至连局部极值也未达到,这种现象被称为早熟收敛或停滞^[11,12]. 发生该现象时粒子群高度聚集,严重缺乏多样性,粒子群会长时间或永远跳不出聚集点^[13]. 因此,大量对粒子群优化算法的改进集中在提高粒子群的多样性上,使得粒子群在整个迭代过程中能保持进一步优化的能力.

3 Tent 映射性能分析及改进

Tent 映射又称帐篷映射,是分段线性的一维映射,具有均匀的概率密度、功率谱密度和理想的相关特性,其数学表达式为

$$x_{n+1} = -1 - |x_n|, \quad (1, 2], \quad (3)$$

Tent 映射的 Lyapunov 指数为

$$= \lim_n \frac{1}{n} \ln \left| \frac{dx_{n+1}}{dx_n} \right| = \ln 2. \quad (4)$$

当 $|\lambda| < 1$ 时, $\lambda > 0$, 系统处于稳定状态;当 $|\lambda| > 1$ 时, $\lambda > 0$, 系统处于混沌状态;当 $|\lambda| = 2$ 时, $\lambda_{\max} = \ln 2$, 为中心 Tent 映射,其数学表达式为

$$x_{k+1} = \begin{cases} 2x_k, & 0 \leq x_k \leq 0.5; \\ 2(1 - x_k), & 0.5 < x_k \leq 1. \end{cases} \quad (5)$$

Tent 映射结构简单,具有很好的遍历均匀性,更适合大数量级数据序列的运算处理,迭代速度要快于 Logistic 映射. Tent 映射有 0 和 $2/3$ 两个不动点,每一点的斜率 $|f'(x)| = 2 > 1$,因此这两个不动点均为不稳定的. 当在式(5)中取 $x_k = 0, 1/2, 1$ 时, x_{k+1} 分别为 0, 1, 0. 这表明原来 x_k 的取值是从 $0 \sim 1/2$ 和从 $1/2 \sim 1$ 的两个区间,通过一次迭代后使得 x_{k+1} 的取值变为从 $0 \sim 1$ 的区间,相当于拉长了映像. 这种迭代过程不断地进行下去,其伸长特性最终导致相邻点的指数分裂,产生对初值的敏感依赖性,同时其折叠过程保持轨道有界.

Tent 映射比 Logistic 映射具有更好的遍历性,基于 Tent 映射的混沌优化方法具有更高的寻优效率. 但 Tent 映射迭代序列中存在小周期,例如 4 周期: $(0.2, 0.4, 0.6, 0.8)$; 还存在不稳周期点,例如 $0.25, 0.5, 0.75$ 都将迭代到不动点 0 ^[14-17]. 为了不趋向不动点,本文采用引入随机方程的方法改进 Tent 映射: 如果 $x_k = 0, 0.25, 0.5, 0.75$, 或 $x_k = x_{k-m}, m = \{0, 1, 2, 3, 4\}$ (即 x 落入不动点或 5 周期以内的小循环), 则引入式(6)加以改进. 在随机方程的扰动下, Tent 映射能够在达到小周期点或不动点时重新进入混沌状态.

$$x_{k+1} = \begin{cases} 2(x_k + 0.1 \times \text{rand}(0, 1)), & 0 \leq x_k \leq 0.5; \\ 2(1 - (x_k + 0.1 \times \text{rand}(0, 1))), & 0.5 < x_k \leq 1. \end{cases} \quad (6)$$

改进的 Tent 映射在达到小周期点或不动点时,由于随机函数 $f(x) = 0.1 \times \text{rand}(0, 1)$ 的作用,对 x_k 序列进行扰动跳出小周期点或不动点,使得 Tent 映射重新进入混沌状态,而非进入周期循环状态,因而改进的 Tent 映射更具有遍历性.

if $x_k = \{0, 0.25, 0.5, 0.75\}$
 or $x_k = x_{k-m}, m = \{0, 1, 2, 3, 4\}$,
 then $x_k = x_k + 0.1 \times \text{rand}(0, 1)$.

改进的 Tent 映射跳出了小周期点或不动点. 采用改进的 Tent 序列, 初值取为 0.4567, 迭代 30000 次, 在 [0, 1] 各点的概率分布如图 1 所示. 可看出最少取值次数为 467, 最大为 847 次, 平均在 700 次左右. 改进的 Tent 序列相比 Logistic 混沌序列具有更好的遍历性, 而且它克服了自身的小周期、不稳周期点等不足.

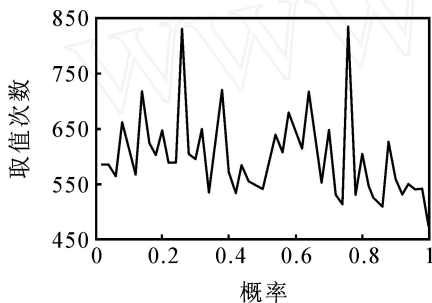


图 1 改进的 Tent 迭代 30 000 次的分布图

4 基于改进的 Tent 映射的混沌粒子群算法

4.1 优化策略

本文在粒子群算法中引入改进的 Tent 混沌序列, 丰富了粒子的行为, 并将粒子均匀映射到定义空间中, 有利于粒子跳出局部极值点. 提出了改进的混沌粒子群算法, 优化策略是以分阶段的思想对基于改进的 Tent 映射的混沌粒子群算法加以改进. 在粒子群搜索初期, 以改进的 Tent 混沌序列进行粒子的初始化, 加强粒子对其邻域快速搜优, 改善了算法的随机性, 有助于提高寻优的效率和精度. 并以改进的 Tent 混沌序列代替随机数对粒子的速度、位置进行更新. 到粒子群搜索的中后期, 为了加快收敛速度, 提高搜索效率, 采用收缩搜索范围的方法. 每迭代一定步数后将离最优解距离最远的粒子剔除, 以所有粒子位置的平均值重新创建新粒子的位置, 进行新一次的搜索. 由于搜索范围的减小, 搜索速度会在一定范围内有所提升, 直到最终找到最优解.

4.2 基于改进的 Tent 映射粒子群算法流程

在粒子群搜索的早期, 按如下步骤进行:

Step1: 混沌初始化粒子群, 以改进的 Tent 序列产生一个 n 维、每个分量数值介于 [0, 1] 之间的向量 $x_1 = (x_{11}, x_{12}, \dots, x_{1n})$, 根据式 (6) 得到 N 个向量 x_1, x_2, \dots, x_N . 计算目标函数, 从 N 个初始群体中选择性能较好的 m 个解作为初始解, 随机产生 m 个初始速度, 令粒子最大速度为 v_{\max} .

Step2: 根据当前位置和速度产生各个粒子的新位置并计算适应值, 若粒子的适应值优于原来的个体极值 p_{best} , 则重置 p_{best} ; 如果适应值优于全局极值 g_{best} , 则重置 g_{best} . 以改进的 Tent 序列代替随机数 r_1 和 r_2 , 应用式 (1) 更新粒子速度 v_{id} , 利用式 (2) 更新粒子位置 p_{id} .

Step3: 如果达到初期混沌搜索次数, 执行 Step4; 否则执行 Step2.

在粒子群搜索的后期, 按如下步骤进行:

Step4: 每隔一定周期 (20 步) 对最差粒子 l (离最优解距离最远的粒子) 进行改变, 以所有粒子位置的平均值 \bar{x} 重新创建新粒子的位置. 令 $x_l = \bar{x} = \frac{1}{m} \sum_{i=1}^m x_i$, 进行新一次的搜索.

Step5: 应用式 (1) 和 (2) 更新每个粒子的速度 v_{id} 和位置 p_{id} .

Step6: 如果达到总的迭代次数, 执行 Step7; 否则返回执行 Step4.

Step7: 根据各个粒子的个体极值 p_{best} , 找出全局极值 g_{best} 和全局极值位置, 判断搜索是否成功.

5 数值仿真

5.1 测试函数简介

本文运用 4 个标准的测试函数对混沌粒子群的寻优能力进行分析. 在表 1 中, De Jong 函数和 Rosenbrock 函数是用来评价混沌粒子群的全局收敛性的经典函数. Griewank 函数具有无穷多个极值点, 用来评价算法的探索、开发能力. Schaffer F6 是标准的二维函数.

表 1 测试函数

函数	表达式	可行域	维数	停止标准	全局最优的 x^*	全局最小值
Rosenbrock	$f(x) = \sum_{i=1}^n (100 \times (x_{i+1} - x_i^2)^2 + (x_i - 1)^2)$	[- 100, 100]	30	0.01	(1, 1, ..., 1)	0
De Jong	$f(x) = \sum_{i=1}^n x_i^2$	[- 30, 30]	30	100	(0, 0, ..., 0)	0
Griewank	$f(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	[- 600, 600]	30	0.1	(0, 0, ..., 0)	0
Schaffer F6	$f(x) = 0.5 - \frac{(\sin \sqrt{x^2 + y^2})^2 - 0.5}{(1.0 + 0.001(x^2 + y^2))^2}$	[- 100, 100]	2	0.00001	(0, 0, ..., 0)	0

5.2 混沌粒子群参数设置

本文对4个测试函数分别应用标准PSO和改进的Tent PSO进行比较分析,设置循环次数为500次,其中前100次设置为搜索初期,第101~500次为搜索的中期及后期.在粒子群中, $c_1 = 2.8$, $c_2 = 1.3$,粒子数为30.在搜索的中后期,每隔20步,对粒子进行一次更新.改进的Tent混沌序列中初值为0.4567.

5.3 仿真效果及分析

5.3.1 仿真效果

图2~图5是改进的Tent PSO和标准PSO两

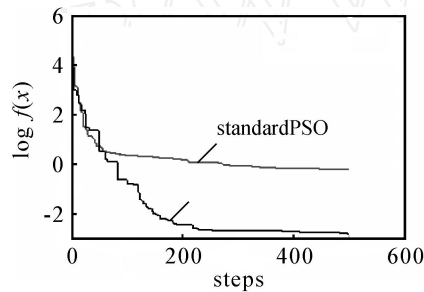


图2 Rosenbrock 函数

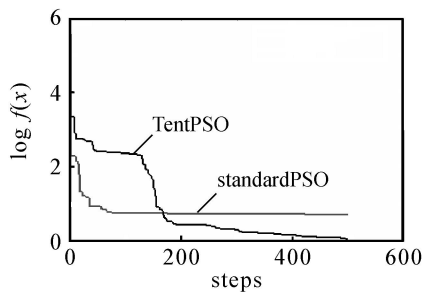


图3 De Jong 函数

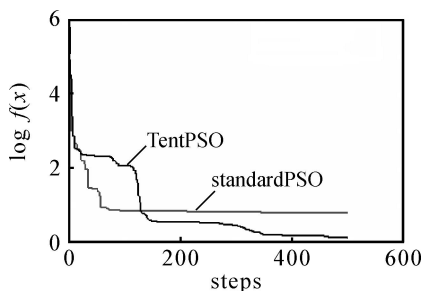


图4 Griewank 函数

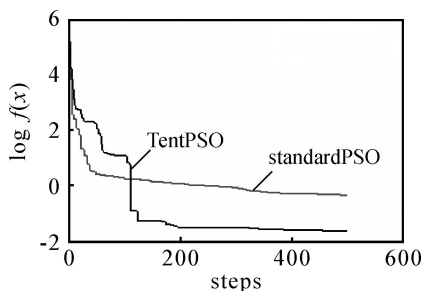


图5 Schaffer F6 函数

种算法应用于4个测试函数的寻优效果图.其中纵轴数据均取 $\log f(x)$,以便观察.

经过测试100次后,标准PSO和改进的Tent PSO两种算法的搜索成功率见表2.

	Rosenbrock	De Jong	Griewank	Schaffer F6
Standard PSO	74	100	59	32
Improved Tent PSO	92	99	96	99

5.3.2 仿真结果比较分析

Rosenbrock 和 De Jong 函数是简单的单峰函数,标准PSO在处理高维的单峰函数时,结果较为理想;Griewank 和 Schaffer F6 函数是多峰函数,应用标准PSO时优化性能降低,易陷入局部最优.标准PSO在实现过程中易早熟收敛、局部搜索能力弱,在进化的后期,解的精度很难得到提高.标准PSO早期收敛速度较快,但到寻优的后期,其结果改进则不太理想,这主要由于算法收敛到局部极小,缺乏有效机制使算法逃离极小点.通过引入改进的Tent映射,增强了算法的全局搜索能力与搜索精度,摆脱了局部极值点的吸引,同时又不降低收敛速度.通过对粒子位置更新加以引导,减少算法随机性以提高搜索效率,可使种群在更新中向前进化,从而避免了算法早熟收敛,保证了收敛到全局最优,而没有更新的种群继续向前进化,又保证了搜索精度的连续提高.

标准PSO收敛速度较快,特别是在算法的早期,但也存在着精度较低、易发散等缺点.若加速因子、最大速度等参数太大,粒子群可能错过最优解,算法不收敛;而在收敛的情况下,由于所有的粒子都向最优解的方向飞去,粒子趋向同一化,使得后期收敛速度变慢,同时算法收敛到一定精度时,无法继续优化,所能达到的精度也比改进的Tent PSO算法低.

改进的Tent PSO则克服了标准PSO由于无法事先知道多峰函数峰值点个数而很难确定合适群体大小的困难.对于Rosenbrock函数(见图2),改进的Tent PSO的寻优成功率明显优于标准PSO,且精度要略优于标准PSO,而且在搜索初期,搜索范围更广.对于De Jong函数(见图3),改进的Tent PSO寻优成功率略逊于标准PSO,收敛速度较慢,但改进Tent PSO的搜索精度要高.对于Griewank函数(见图4),改进的Tent PSO明显优于标准PSO,在寻优精度上更胜一筹,但在初始搜索阶段,它的收敛较慢.对于Schaffer F6函数(见图5),改进的Tent PSO的寻优成功率要优于标准PSO,并且Tent PSO有较

好的搜索精度,收敛速度也较快.

通过仿真可以看出,对于 Griewank 和 Schaffer F6 函数,基于改进的 TentPSO 无论在收敛速度还是次数上都优于标准 PSO,体现了混沌序列的遍历性,以及良好的全局搜索能力. 总体来看,对于大部分的测试函数,改进的 TentPSO 具有较好的寻优效果,解决了粒子群算法的局部收敛问题,而且通过优化策略确保了全局搜索性能和局部搜索性能的动态平衡.

6 改进的 TentPSO 在资源配置结构优化决策中的应用与仿真

设有价值为 W 的资源配置到项目集合 P 中, P 包含 n 个项目, $P_i (i = 1, 2, \dots, n)$ 为第 i 个项目. 对 n 个项目配置资源,将资源配置到项目的平均收益率是 b_i ,项目 P_i 的风险损失率是 r_i . 将资源 W_i 配置到项目 P_i 中需要付出成本,费用率为 i ,并且当配置的资源价值不超过给定值 μ_i 时,成本按资源配置 μ_i 计算. 配置到项目 P_i 的资源 W_i 占资源总价值 W 的比例为 x_i ,资源配置成本为

$$c_i(x_i) = \begin{cases} 0, & x_i = 0; \\ i\mu_i, & 0 < x_i < (1 + i) \frac{\mu_i}{W}; \\ \frac{\mu_i W}{1 + i} x_i, & (1 + i) \frac{\mu_i}{W} < x_i < 1. \end{cases} \quad (7)$$

资源配置的目标在于使净收益尽可能大,总体风险尽可能小,因此为双目标函数. 可以用乘法简化建立以下多目标决策模型(设收益为 $I(X)$, 风险为 $R(X)$):

$$\begin{cases} \max I(X), \\ \min R(X), \\ I(X) > 0, R(X) > 0, X \in D. \end{cases} \quad (8)$$

使资源配置后项目总效益达到最大,即

$$\begin{aligned} \max Q(X) &= \frac{I(X)}{R(X)}, \\ \min F(X) &= \min \frac{1}{Q(X)}. \end{aligned}$$

P_i 项目的收益为 $I_i(x_i) = b_i M x_i - b_i c_i$; P_i 项目的风险为 $R_i(x_i) = r_i M x_i - r_i c_i$; P_i 项目所需资源价值为 $i = M x_i$; 任务达成率为 i ; 资源配置组合为 $x = (x_1, x_2, \dots, x_n)$; 项目集合 P 总的效益 $I(x) = \sum_{i=1}^n I_i(x_i)$; 项目集合 P 的整体风险为 $R(x) = \sum_{i=1}^n R_i(x_i)$; 资源约束 $(x) = \sum_{i=1}^n i(x_i)$; 资源配置的相关项目集合 $P_{i_1} \quad P_{i_2} = \phi, i_1 \quad i_2$; 资源在分配过程中是没有缺失的, $\sum_{i=1}^n M_i = M, M_{i_1} \quad M_{i_2} = \phi,$

$i_1 \quad i_2$.

在以上模型的基础上,设某种资源总价值为 5 000 万元,分配至 6 个项目中,设粒子数目为 50,最大迭代次数为 500,学习因子 $c_1 = 2.5, c_2 = 1.5$,项目收益权重为 0.7,风险权重为 0.3,资源配置结构优化数据如表 3 所示,分别应用改进的 TentPSO 和标准 PSO 对资源配置模型进行结构优化.

表 3 数据值

$i(x_i)$	b_i	r_i	i	μ_i	i
1(x_1)	25	5.0	5	120	100
2(x_2)	36	3.5	4	150	98
3(x_3)	31	2.0	4.5	180	100
4(x_4)	19	1.5	5.5	100	95
5(x_5)	44	6.0	6	180	96
6(x_6)	51	4.5	4	200	95

表 4 资源配置结构优化仿真结果

	x_1	x_2	x_3
标准 PSO	0.01247	0.1640	0.27367
改进的 TentPSO	0.031	0.1701	0.30163
	x_4	x_5	x_6
标准 PSO	0.19682	0.17301	0.18002
改进的 TentPSO	0.26187	0.06889	0.18651

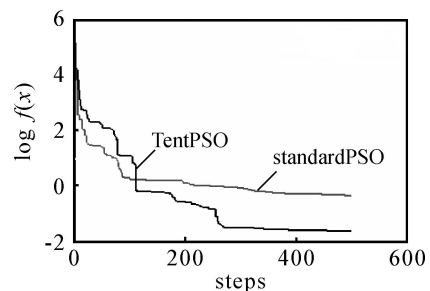


图 6 改进的 TentPSO 和标准 PSO 对资源配置结构优化的叠代曲线

从图 6 与表 4 可以看出,应用改进的 TentPSO 进行资源配置的结构优化,明显优于标准粒子群优化算法. 优化策略加强了算法的局部寻优效率,改进的 Tent 映射混沌搜索增加了粒子群的多样性,增强了算法的全局寻优能力,有利于实现资源配置结构最优决策. 改进的 TentPSO 算法与资源配置结构优化控制模型不仅为经济运行中有限资源最优分配方案提供了支持工具,还适用于动态复杂系统结构决策分析与优化控制,便于高效、准确地实现系统结构最优.

7 结 论

本文将 Tent 混沌映射同粒子群算法相结合,提出了一种基于改进的 Tent 序列的混沌粒子群算

法.通过仿真证实该方法对一般的函数具有较好的寻优能力,克服了传统优化算法易陷入局部极值的缺点,能够以较快的速度得到全局最优,其优化性能优于标准粒子群算法.在此基础上,本文将改进的 Tent PSO 算法应用于资源配置结构优化控制决策中,取得了较好的寻优效果,因而 Tent PSO 算法具有较强的探索开发能力.

参考文献(References)

- [1] 孟红记, 郑鹏, 梅国晖, 等. 基于混沌序列的粒子群优化算法[J]. 控制与决策, 2006, 21(3): 263-266.
(Meng H J, Zheng P, Mei G H, et al. Particle swarm optimization algorithm based on chaotic series [J]. Control and Decision, 2006, 21(3): 263-266.)
- [2] 刘华莹, 林玉娥, 张君施. 基于混沌搜索解决早熟收敛的混合粒子群算法[J]. 计算机工程与应用, 2006, 42(13): 77-79.
(Liu H Y, Lin Y E, Zhang J S. A hybrid particle swarm optimization based on chaos strategy to handle local convergence [J]. Computer Engineering and Applications, 2006, 42(13): 77-79.)
- [3] Suresh S, Sujit P B, Rao A K. Particle swarm optimization approach for multi-objective composite box-beam design[J]. Composite Structures, 2007, 81(4): 598-605.
- [4] Awad El-Gohary, Al-Ruzaiza A S. Chaos and adaptive control in two prey, one predator system with nonlinear feedback[J]. Chaos, Solitons and Fractals, 2007, 34(2): 443-453.
- [5] Bo Liu, Ling Wang, Yi-Hui Jin, et al. Improved particle swarm optimization combined with chaos[J]. Chaos, Solitons and Fractals, 2005, 25(5): 1261-1271.
- [6] Shi X H, Liang Y C, Lee H P, et al. Particle swarm optimization-based algorithms for TSP and generalized TSP[J]. Information Processing Letters, 2007, 103(5): 169-176.
- [7] 王希云, 刘瑞芳. 混沌粒子群算法及其在架结构优化设计中的应用[J]. 太原科技大学学报, 2006, 27(6): 478-480.
(Wang X Y, Liu R F. Chaos-based particle swarm optimization algorithm and its application in truss-optimization design [J]. J of Taiyuan University of Science and Technology, 2006, 27(6): 478-480.)
- [8] 陈冬阳, 孙昌志, 刘子俊, 等. 混沌粒子群混合算法对微型永磁电机的优化设计[J]. 沈阳工业大学学报, 2006, 28(6): 614-618.
(Chen D Y, Sun C Z, Liu Z J, et al. Optimal design of
- minitype permanent magnet motor based on hybrid algorithm of chaos and PSO [J]. J of Shenyang University of Technology, 2006, 28(6): 614-618.)
- [9] Kathiravan R, Ganguli R. Strength design of composite beam using gradient and particle swarm optimization[J]. Composite Structures, 2007, 81(4): 471-479.
- [10] Shur Kai S Fan, Erwie Zahara. A hybrid simplex search and particle swarm optimization for unconstrained optimization[J]. European J of Operational Research, 2007, 181(2): 527-548.
- [11] De Falco I, Della Cioppa A, Tarantino E. Facing classification problems with particle swarm optimization [J]. Applied Soft Computing, 2007, 7(3): 652-658.
- [12] 刘洪波, 王秀坤, 谭国真. 粒子群优化算法的收敛性分析及其混沌改进算法[J]. 控制与决策, 2006, 21(6): 636-645.
(Liu H B, Wang X K, Tan G Z. Convergence analysis of particle swarm optimization and its improved algorithm based on chaos [J]. Control and Decision, 2006, 21(6): 636-645.)
- [13] 李莉, 李洪奇. 基于混合粒子群算法的高维复杂函数求解[J]. 计算机应用, 2007, 27(7): 1754-1756.
(Li L, Li H Q. Solving for complex functions with high dimensions based on hybrid particle swarm optimization [J]. Computer Applications, 2007, 27(7): 1754-1756.)
- [14] Dixiong Yang, Gang Li, Gengdong Cheng. On the efficiency of chaos optimization algorithms for global optimization[J]. Chaos, Solitons and Fractals, 2007, 34(4): 1366-1375.
- [15] 单梁, 强浩, 李军, 等. 基于 Tent 映射的混沌优化算法[J]. 控制与决策, 2005, 20(2): 179-182.
(Shan L, Qiang H, Li J, et al. Chaotic optimization algorithm based on Tent map [J]. Control and Decision, 2005, 20(2): 179-182.)
- [16] 程志刚, 张立庆, 李小林, 等. 基于 Tent 映射的混沌混合粒子群优化算法[J]. 系统工程与电子技术, 2007, 29(1): 103-106.
(Cheng Z G, Zhang L Q, Li X L, et al. Chaotic hybrid particle swarm optimization algorithm based on Tent map [J]. Systems Engineering and Electronics, 2007, 29(1): 103-106.)
- [17] Sonja Stimac. A classification of inverse limit spaces of tent maps with finite critical orbit[J]. Topology and Its Applications, 2007, 154(11): 2265-2281.