

基于粒子群算法的软件可靠性分配问题研究

贾冀婷

(西安邮电学院 计算机学院, 陕西 西安 710121)

摘 要: 针对软件可靠性分配中不易求解全局最优解这一问题, 将可靠性指标分配到每个模块中, 并利用改进的粒子群优化算法来搜索模型的最优解. 实验结果表明, 改进的粒子群优化算法在求解软件可靠性分配问题时的效果优于遗传算法等其他智能优化算法.

关键词: 粒子群优化算法; 可靠性分配; 智能优化算法

中图分类号: TP311

文献标识码: A

文章编号: 1000-7180(2011)12-0068-04

Reliability Allocation of Software System Based on Particle Swarm Optimization

JIA Ji-ting

(Department of Computer, Xi'an University of Post and Telecommunications, Xi'an 710121, China)

Abstract: In accordance with difficulties of achieving the optimal result in the problem of the software reliability allocation, reliability targets are allocated to each model, and utilizing the improved particle swarm optimization algorithm to achieve the optimal result for the problem in this paper. Experiment has proved that this improved version of particle swarm optimization algorithm is more effective on solving the software reliability allocation problems than other intelligent optimization algorithms such as genetic algorithm.

Key words: particle swarm optimization; reliability allocation; intelligent optimization algorithms

1 引言

软件可靠性是衡量软件质量的一个重要指标, 而软件可靠性分配则是软件可靠性工程中的一项重要设计技术. 软件可靠性分配问题是不可微、不连续、多维、有约束条件、高度非线性的 NP 完全问题^[1], 很难精确求解全局最优解. 近年来, 许多学者利用遗传算法、模拟退火算法等多种智能优化算法来求解此问题, 取得了一定成果, 但求解效果还不尽如人意.

粒子群优化算法^[2-4]的基本思想是通过群体中个体之间的协作和信息共享来寻找最优解. 它包含了进化计算和群体智能的特点, 是进化计算和智能优化领域的一个研究热点, 并且已在神经网络训练、目标函

数优化等领域被广泛应用. 本文改进的粒子群优化算法在求解软件可靠性分配问题中取得了较好的效果.

2 多模块软件可靠性分配模型

对于一个有 n 个模块、 m 项操作过程的软件系统, 它的可靠性函数可表示为^[5]

$$R(\lambda_1, \lambda_2, \dots, \lambda_n; \tau) = \exp\left(-\sum_{j=1}^n \sum_{i=1}^m p_i q_{ij} \lambda_j \tau\right) \quad (1)$$

式中, λ_j 是模块 j 的故障率; τ 是系统的运行时间; p_i 是系统中执行第 i 项功能的概率; q_{ij} 是第 i 项功能中执行模块 j 的概率.

模块 j 在整个系统运行过程中被执行的概率可

收稿日期: 2011-03-20; 修回日期: 2011-05-19

基金项目: 国家自然科学基金项目“基于群体智能的组合软件测试优化关键问题研究”(61050003); 陕西省科技攻关计划项目“嵌入式软件智能组合测试平台”(2009K08-26); 陕西省教育厅自然科学基金项目“基于群体智能的组合测试方法研究”(09JK727)

定义为

$$\Phi_j = \sum_{i=1}^m p_i q_{ij} \quad (2)$$

由式(1)、(2)可以得到 n 个模块的软件系统的可靠性函数为

$$R(\lambda_1, \lambda_2, \dots, \lambda_n; \tau) = \prod_{j=1}^n \exp(-\Phi_j \lambda_j \tau) \quad (3)$$

该软件系统的开发费用可表示为

$$TC(\lambda_1, \lambda_2, \dots, \lambda_n) = \sum_{j=1}^n C(\lambda_j) \quad (4)$$

该软件系统的可靠性应满足如下表达式:

$$R(\lambda_1, \lambda_2, \dots, \lambda_n; \tau) \geq \rho \quad (5)$$

式中, ρ 为 n 个模块、 m 项操作过程的软件系统的可靠性目标。下面就采用粒子群优化算法来求解这一工程问题。

3 粒子群优化算法的基本原理

粒子群优化算法是通过种群中个体之间的协作和信息共享来实现最优解的搜索,与其他进化算法的区别在于它是对社会行为的模拟,其基本思想是源于群体组织的社会行为。粒子群优化算法的基本原理就是加速每个粒子朝它自己所经历的和种群所经历的最好位置逼近。

粒子群优化算法的主要步骤为:首先生成初始种群,即在搜索空间中随机初始化一群粒子,这一群粒子中的每个粒子都是优化问题的一个可行解,并且每个粒子都应确定一个适应值。种群中的每个粒子在运动过程中都是由一个速度来决定其距离和方向的。种群中的每个粒子都将追随当前的最优粒子经过逐代搜索最终得到最优解。在逐代搜索的过程中,每个粒子都会跟踪两个极值,一个是粒子本身迄今为止找到的最优解即个体极值,另一个是整个种群迄今为止找到的最优解即全局极值^[6]。算法流程图如图1所示。

粒子群优化算法的一般数学表示为:设在一个 n 维的空间中,有一个种群 X 包含了 m 个粒子,即 $X = \{X_1, \dots, X_i, \dots, X_m\}$,种群中第 i 个粒子位置为 $X_i = \{X_{i1}, X_{i2}, \dots, X_{in}\}^T$,其速度是 $V_i = \{V_{i1}, V_{i2}, \dots, V_{in}\}^T$ 。第 i 个粒子的个体极值为 $P_i = \{P_{i1}, P_{i2}, \dots, P_{in}\}^T$,种群的全局极值是 $P_g = \{P_{g1}, P_{g2}, \dots, P_{gn}\}^T$,依据粒子群优化算法的基本原理,种群中第 i 个粒子将根据式(6)、(7)更新速度和位置^[7]:

$$v_{id}^{(t+1)} = v_{id}^{(t)} + c_1 r_1 (p_{id}^{(t)} - x_{id}^{(t)}) + c_2 r_2 (p_{gd}^{(t)} - x_{id}^{(t)}) \quad (6)$$

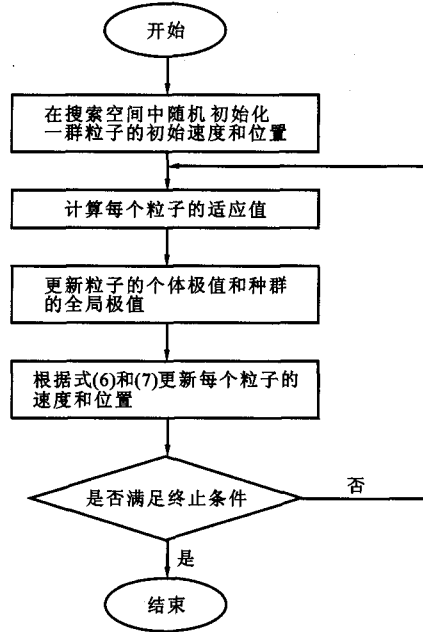


图1 粒子群优化算法流程图

$$x_{id}^{(t+1)} = x_{id}^{(t)} + v_{id}^{(t+1)} \quad (7)$$

式中, $d = 1, 2, \dots, n$, n 表示解空间中粒子的维数; $i = 1, 2, \dots, m$, m 表示种群规模的大小; t 是当前的进化代数,表示粒子的飞行步数; r_1, r_2 是分布在 $[0, 1]$ 之间的随机数; c_1 和 c_2 是加速常数或学习因子,分别调节该粒子向自身已寻找的最优位置和同伴已寻找的最优位置方向飞行的最大步长。

4 粒子群优化算法的改进

由于在处理高维的复杂问题时,基本粒子群优化算法容易陷入局部最优^[8],本文针对此问题对其进行了改进。在自然界中,一个机体最容易被它临近的机体所影响并且在所有邻近的机体中,它最容易被比它更好的粒子所影响。因此,本文针对基本的粒子群优化算法,在局部最优位置、全局最优位置对粒子当前位置影响的基础上,增加了“邻居”最优位置对粒子当前位置的影响。即将种群中除粒子本身之外的其它所有粒子都看作此粒子的邻居,使粒子不仅能从局部最优和全局最优粒子中学习经验,而且能从“邻居”即临近的粒子中适应值最好的粒子吸取经验,使粒子之间能更好地共享社会信息。

本文对基本粒子群优化算法公式(6)进行了改进,则该公式变为

$$v_{id}^{(t+1)} = v_{id}^{(t)} + c_1 r_1 (p_{id}^{(t)} - x_{id}^{(t)}) + c_2 r_2 (p_{gd}^{(t)} - x_{id}^{(t)}) + c_3 r_3 (p_{nd}^{(t)} - x_{id}^{(t)}) \quad (8)$$

由式(8)、(7)组成的迭代算法为改进的粒子群优化算法。其中: c_3 为学习因子或加速常数; r_3 为分

布于[0,1]之间的随机数; p_k 为第*i*个粒子的“邻居”中具有最好适应值的粒子 *k* 的位置;其余参数与基本粒子群优化算法公式中的参数一致.

本文改进的粒子群优化算法中增加了“邻居”最优位置对粒子当前位置的影响,在基本粒子群优化算法局部收敛能力的基础上,进一步提高了算法的局部收敛能力.改进的算法引导粒子朝着粒子领域中适应值最好的粒子飞去,比基本的粒子群优化算法优化效果好.改进后的粒子群优化算法在避免陷入局部最优的同时还保持了较好的收敛速度.在算法中,避免了个体最好解与全局最好解或邻域最好解之间的杂交,算法的搜索效率有了很大的提高.

5 实验及结果分析

本文采用一个“XX省邮政物流信息系统”项目中的“礼仪中心子系统”的算例,使用本文改进的粒子群优化算法来求解软件可靠性分配问题,并与基本粒子群优化算法、遗传算法、非平衡模拟退火算法求解此问题的性能进行了比较.

算例:最小化礼仪中心子系统的开发成本
目标函数:

$$\min\{f=-238\ln(1-\exp(-\lambda_1))-305\ln(1-\exp(-\lambda_2))\}$$

约束条件:

$$\begin{cases} \exp(-(0.39\lambda_1+0.61\lambda_2))\geq 0.973 \\ \lambda_1\geq 0 \\ \lambda_2\geq 0 \\ \exp(-0.39\lambda_1)\geq 0.925 \\ \exp(-0.61\lambda_2)\geq 0.925 \\ 387+262\exp(-0.39\lambda_1)\leq 0.47\times 1\,382.5 \\ 206+685\exp(-0.61\lambda_2)\leq 0.47\times 1\,896.5 \\ 593+262\exp(-0.39\lambda_1)+ \\ 685\exp(-0.61\lambda_2)\leq 2\,780 \end{cases}$$

式中, λ_1 、 λ_2 为礼仪中心子系统的两个模块的故障率参数,算例应满足约束条件即在系统的可靠性大于97.3%、单个模块的可靠性大于92.5%的情况下,完成目标函数即最小化该软件系统的开发成本.本文采用改进的粒子群优化算法来求解这一工程问题.模拟运行时的参数如下:粒子群数 $m=20$, $c_1=c_2=2$, $c_3=0.1$,迭代次数为1 000次.

本文分别使用改进的粒子群优化算法、基本粒子群优化算法、遗传算法、非平衡模拟退火算法来求解上述算例的可靠性分配问题,每个算法分三组进行,每组随机执行60次,结果如表1所示.从表1可以看出,本文改进的粒子群优化算法在求解上述算例的可靠性分配问题的优化结果较好.

表1 本文算法与其他算法的实验结果比较

实验序号	算法	最佳优化值	平均优化值	最差优化值	优化值方差	陷入局部最小次数
1	本文算法	1 958.717	1 958.749	1 959.406	0.785 632 6	3
	基本粒子群算法	1 961.537	1 968.286	2 000.125	363.290 30	19
	遗传算法	1 972.537	1 981.126	2 002.225	176.579 57	25
	非平衡模拟退火算法	1 985.537	1 992.871	2 006.722	83.679 253	49
2	本文算法	1 958.715	1 958.733	1 959.393	0.268 752 1	5
	基本粒子群算法	1 960.812	1 969.272	2 007.029	439.054 22	22
	遗传算法	1 969.337	1 981.259	2 003.329	189.739 66	27
	非平衡模拟退火算法	1 978.537	1 991.106	2 008.292	162.112 79	49
3	本文算法	1 958.718	1 958.739	1 959.359	0.142 538 1	4
	基本粒子群算法	1 962.362	1 972.226	2 008.723	453.423 17	17
	遗传算法	1 979.537	1 988.753	2 005.359	117.231 30	28
	非平衡模拟退火算法	1 982.537	1 995.621	2 012.127	183.679 70	52

实验结果表明,本文改进的粒子群优化算法在求解软件可靠性分配问题时的稳定性较好,每次的结果相差都非常小,并且收敛速度较快,随着进化代

数的增加,总开发成本呈逐渐减少趋势.因此,本文改进的粒子群优化算法在求解软件可靠性分配问题中显示了较好的优化性能.

6 结束语

本文将改进的粒子群优化算法应用到软件系统的可靠性分配问题中,实验结果表明,改进的粒子群优化算法在稳定性和收敛性上均优于基本粒子群优化算法、遗传算法、非平衡模拟退火算法等智能优化算法,为软件可靠性分配问题提供了一个新的高效的解决途径。

参考文献:

- [1] 韩冰青, 高建华. 基于模拟退火遗传算法的软件可靠性分配及研究[J]. 计算机工程, 2003, 29(4): 67—69.
- [2] Kennedy J, Eberhart R. Particle swarm optimization [C]// Proc IEEE Int Conf on Neural Networks. Australia; Perth, 1995: 1942—1948.
- [3] 裴胜玉, 周永权, 罗淇方. 求解约束优化问题的混合粒子群算法[J]. 微电子学与计算机, 2010, 27(4): 5—8.
- [4] 黄少荣. 基于随机加速系数的粒子群优化算法[J]. 微电

子学与计算机, 2010, 27(6): 114—117.

- [5] Lyu M R. Handbook of software reliability engineering [M]. New York: McGraw Hill, 1996.
- [6] Clerc M. The swarm and the queen: towards a deterministic and adaptive particle swarm optimization [C]// Proceedings of 1999 Congress Evolutionary Computation. Piscataway, NJ: IEEE Press, 1999: 951—957.
- [7] Kennedy J, Eberhart R C. A new optimizer using particle swarm theory [C]// Proceedings of the sixth International Symposium on Micro Machine and Human Science. Nagoya, Japan, 1995: 39—43.
- [8] Shi Y, Eberhart R C. A modified particle swarm optimizer [C]// Proceedings of the IEEE Congress on Evolutionary Computation. USA; Alaska, 1998: 69—73.

作者简介:

贾冀婷 女, (1978—), 硕士, 工程师. 研究方向为软件测试、智能优化算法.

(上接第 67 页)

- [3] Lehmann T M, Conner C, Spitzer K. Addendum: B-spline interpolation in medical image processing [J]. IEEE Transactions on Medical Imaging, 2001, 20(7): 660—665.
- [4] 江巨浪. 基于改进 Catmull_Rom 样条的图像缩放算法 [J]. 计算机技术与发展, 2007, 17(4): 211—217.
- [5] 张阿珍, 刘政林, 雒雪城, 等. 基于双三次插值算法的图像缩放引擎的设计 [J]. 微电子学与计算机, 2007, 24(1): 49—51.
- [6] 冯华, 徐维朴. 一种视频图像分辨率提升的 VLSI 设计与

实现 [J]. 微电子学与计算机, 2004, 21(1): 118—120.

- [7] 刘政林, 肖建平, 雒雪城, 等. 基于边缘的实时图像缩放算法研究 [J]. 中国图象图像学报, 2008, 13(2): 225—229.

作者简介:

魏美荣 女, (1984—), 硕士研究生. 研究方向为集成电路设计.

蒋林 男, (1970—), 博士, 教授, 硕士生导师. 研究方向为专用集成电路设计.