

可靠性优化的蚁群算法

高 尚^{1,2} 杨静宇² 吴小俊¹ 刘同明¹¹(华东船舶工业学院电子与信息系 镇江 212003) ²(南京理工大学计算机系 南京 210094)

摘 要 建立了可靠性冗余优化模型,分析了各种优化方法的优缺点。采用模拟退火算法、遗传算法和蚁群算法分别解决了此问题,并通过实例,结果表明蚁群算法比较有效。

关键词 蚁群算法 模拟退火算法 遗传算法 可靠性优化

ANT COLONY ALGORITHM FOR OPTIMIZATION OF SYSTEM RELIABILITY

Gao Shang^{1,2} Yang Jingyu² Wu Xiaojun¹ Liu Tongming¹¹(Department of Electronics and Information, East China Shipbuilding Institute, Zhenjiang 212003)²(Department of Computer, Nanjing University Science and Technology, Nanjing 210094)

Abstract A redundancy optimization model is given in this paper. Many optimization methods to solve optimum model and their advantages and shortages are analyzed. Simulated annealing algorithm, genetic algorithm and ant colony algorithm are put forward to solve optimum model. The effectiveness of ant colony algorithm is illustrated through result.

Keywords Ant colony algorithm Simulated annealing algorithm Genetic algorithm Reliability optimization

1 引 言

在工业、军事和日常生活的许多方面,系统可靠性的性能对于各种条件下的任务来说,都极其重要,因此可靠性问题是系统设计、研究和运行过程中必须考虑的关键因素之一。由于设计时受到资源限制,包括费用、重量、体积、功耗等方面的限制,系统可靠性最优问题引起了广泛的重视和研究,可靠性优化是可靠性工程中的一项重要工作。有许多改进系统可靠性的方法,但实践表明为比较好的要算最优冗余这一方法。

2 最优冗余优化模型及解法

假设系统由 n 个独立子系统组成,在每个子系统中使用同一种部件(如图 1),冗余优化模型为^[1,2]:

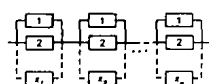


图 1 串-并联系统

$$\begin{aligned} \min C_s &= \sum_{i=1}^n c_i x_i \\ s. t. \quad &\prod_{i=1}^n R_i(x_i) \geq R_0 \\ &x_i = 1, 2, \dots, \quad (i = 1, 2, \dots, n) \end{aligned} \quad (1)$$

这里: C_s —系统费用; R_s —系统可靠度; c_i —第 i 种部件的单价; x_i —第 i 子系统冗余第 i 种部件的个数, $x_i \geq 1$; p_i —第 i 种部件的可靠度, $R_i(x_i)$ —第 i 子系统的可靠度, $R_i(x_i) = 1 - (1 - p_i)^{x_i}$; R_0 —系统要达到预定的可靠度。

上述可靠性优化模型属于 NP-难题。系统冗余可靠性优化方法较多,文献[2]中作了综述,推荐了十几种方法,但在用于大规模非线性规划问题时,仅有少数算法被证明是有效的,且没

有哪一种算法被证明比其它的算法更优越。例如启发式算法虽简单直观,但启发式信息不易找,而且有点盲目性;整数规划得到的是整数解,但为了应用整数规划,必须把非线性目标函数和约束条件转化成线性形式,这是一项困难的任务;动态规划存在着维数“灾难”问题,并且对三个以上约束问题,求解相当困难;极大值原理求解三个以上约束问题也是困难的;几何规划只限于求解可用泊松函数形式表述的问题;序列无约束极小化方法、修正的单纯形序列搜索、广义的拉格朗日函数法用于大型非线性规划问题中被证明是有效的方法,虽然解是非整数,但解决非冗余问题很有效。对于同时确定最优部件数和可靠度的问题,是一类混合整数非线性问题,解决这类问题很困难。文献[2]推荐了将 Hooke 和 Jeeves 等人提出的模式搜索法与 Aggarwal 等人提出的启发式算法组合起来的方法。这种方法首先假设部件的可靠度。然后用启发式算法确定最优冗余数,最后用 Hooke 和 Jeeves 的模式搜索法来进行序列搜索。20 世纪 80 年代以来,一些新颖的优化算法,如人工神经网络、遗传算法、模拟退火、蚁群算法以及混合优化策略等,通过模拟或揭示某些自然现象或过程而得到发展,为解决复杂问题提供了新的思路 and 手段。本文分别采用模拟退火算法、遗传算法和蚁群算法分别来解,最后作一比较。

3 模拟退火算法

模拟退火算法^[3-5]用于优化问题的出发点是基于物理中固

收稿日期:2003-11-03。高尚,讲师,主研领域:系统工程理论与优化研究。

体物质的退火过程与一般优化问题的相似性。算法的基本思想是从一给定解开始的,从邻域中随机产生另一个解,接受准则允许目标函数在有限范围内变坏,它由一控制参数 t 决定,其作用类似于物理过程中的温度 T ,对于控制参数 t 的每一取值,算法持续进行“产生新解-判断-接受或舍弃”的迭代过程,对应着固体在某一恒定温度下趋于热平衡的过程。经过大量的解变换后,可以求得给定控制参数 t 值时优化问题的相对最优解。然后减小控制参数 t 的值,重复执行上述迭代过程。当控制参数逐渐减小并趋于零时,系统亦越来越趋于平衡状态,最后系统状态对应于优化问题的整体最优解。

(1) 给定起、止“温度” $T = 100000$, $T_0 = 1$ 和退火速度 $\alpha = 0.9$, 模拟参数初始化 X_0, p, C ;

(2) 若 $T > T_0$, 转(3), 否则算法停止, 输出 X_0 ;

(3) 计算目标函数值 $CS_0 = CX_0^T$;

(4) 随机产生变量 x_j , 若 $\text{rand}(0, 1) \geq 0.5$, 则 $x_j \leftarrow x_j + 1$, 否则 $x_j \leftarrow x_j - 1$, 此时变量记为 X_1 ;

(5) 判断是否满足约束条件, 若满足转(6), 否则转(4);

(6) 计算目标函数 $CS_1 = CX_1^T$, $\Delta E = CS_1 - CS_0$, 若 $\Delta E \leq 0$, 接受新值, $X_0 \leftarrow X_1$, $T \leftarrow \alpha T$, 转(2); 否则若 $\exp(-\Delta E/T) > \text{rand}(0, 1)$, 也接受新值, $X_0 \leftarrow X_1$, $T \leftarrow \alpha T$, 转(2); 否则转(4)。

4 遗传算法

遗传算法^[3,6]是基于自然遗传和自然优选机理的寻优方法。所谓自然遗传和自然优选来自于达尔文的进化论学说, 该学说认为在生物进化过程中, 任一动植物经过若干代的遗传和变异, 使之能够适应新的环境, 是优胜劣汰的结果, 这种自然遗传思想也适用于求解优化问题。GA 采用选择、交叉和变异运算来实现“物竞天择, 适者生存”这一自然法则的模拟。

为了用遗传算法, 把目标函数取最小值改为求最大值, 数学优化模型为:

$$\min M = \sum_{i=1}^n c_i x_i \quad (2)$$

$$s.t. \quad \prod_{i=1}^n R_i(x_i) \geq R_0$$

$$x_i = 1, 2, \dots, \quad (i = 1, 2, \dots, n)$$

式中 M 为一个大的正数, 如 $M = 250$ 。

这里不采用二进制编码, 用染色体 $X = (x_1, x_2, \dots, x_n)$ 作为解的代码。对于染色体个数 N 、交叉概率 P_c 、变异概率 P_m 和遗传代数 u 的选取值得研究, 染色体个数越大, 其代表性越大, 最终进化到最优解的可能性越大, 但会造成计算量的增加, 一般取适度如 $N = 30$ ^[5]。交叉概率或变异概率过小使解有一定的局限性, 而较大使得进化的随机性增大, 而不容易得到稳定的解, 因此也应取适度。遗传代数越大, 进化最优解的可能性大, 但计算量大, 也要取适度。其遗传算法为:

(1) 给出 P, C , 输入染色体个数 $N = 30$, 交叉概率 $P_c = 0.2$, 变异概率 $P_m = 0.5$, 遗传代数 $u = 100$, $t = 1$;

(2) 产生 $N = 30$ 个并满足约束条件的染色体;

(3) 若 $t > u$, 则停止输出结果, 否则转(4);

(4) 计算目标函数 $f_i (i = 1, 2, \dots, N)$, 并按 f_i 递减顺序对染色体排序, 计算累积概率 $ps_0 = 0$,

$$ps_i = \frac{\sum_{j=1}^i f_j}{\sum_{j=1}^N f_j} \quad (i = 1, 2, \dots, N);$$

(5) 选择操作: 产生 $[0, 1]$ 的随机数 rnd , 若 $ps_{j-1} < rnd \leq ps_j$, 选择染色体 j , 这样选择 $N = 30$ 个;

(6) 交叉操作: 产生 $N = 30$ 个 $[0, 1]$ 的随机数 $rnd_i (i = 1, 2, \dots, N)$, 对每个染色体进行判断, 若 $rnd_i \leq p_c$, 对染色体 i 进行交叉, 随机产生染色体 j 和随机数 α , 交叉后的染色体为 $\alpha X_i + (1 - \alpha) X_j$; 否则不进行交叉;

(7) 变异操作: 产生 $N = 30$ 个 $[0, 1]$ 的随机数 $rnd_i (i = 1, 2, \dots, N)$, 对每个染色体进行判断, 若 $rnd_i \leq p_m$, 对染色体 i 进行变异, 随机产生一个方向 D , 变异后的染色体为 $X_i + mD$ (m 为系数, 如为 5), 若变异后的染色体不可行, 可减小 m 值, 直到可行; 否则不进行变异;

(8) 保留所有代中的最好的染色体, $t \leftarrow t + 1$, 转(3)。

5 蚁群算法

本世纪 50 年代中期创立了仿生学, 人们从生物进化的机理中受到启发, 提出了许多用以解决复杂优化问题的新方法, 如遗传算法、进化规划、进化策略等, 蚁群算法是最近几年才提出的一种新型的模拟进化算法, 由意大利学者 M. Dorigo 等人首先提出^[3,4], 用蚁群在搜索食物源的过程中所体现出来的寻优能力来解决一些离散系统优化中困难问题。已经用该方法求解了旅行商问题、指派问题、调度问题等, 取得了一系列较好的实验结果^[7-9]。

人们经过大量研究发现, 蚂蚁个体之间是通过一种称之为外激素的物质进行信息传递, 从而能相互协作, 完成复杂的任务。蚂蚁在运动过程中, 能够在它所经过的路径上留下该种物质, 而且蚂蚁在运动过程中能够感知这种物质的存在及其强度, 并以此指导自己的运动方向, 蚂蚁倾向于朝着该物质强度高的方向移动。因此, 由大量蚂蚁组成的蚁群的集体行为便表现出一种信息正反馈现象: 某一路径上走过的蚂蚁越多, 则后来者选择该路径的概率就越大, 蚂蚁个体之间就是通过这种信息的交流达到搜索食物的目的。

首先把原约束方程作为罚函数项加入到原目标中, 变成无约束的优化问题, 即:

$$\min \sum_{i=1}^n c_i x_i + M \left\{ \min \left\{ 0, \left[\prod_{i=1}^n (1 - R_i)^{x_i} - R_0 \right] \right\} \right\}^2 \quad (3)$$

其中, M 为一充分大的正数。

由于受到费用、体积、重量的限制, 冗余数量 $x_i (i = 1, 2, \dots, m)$ 不可能很大, 设最大为 x_{\max} , 可靠性框图可转化成如下的网络图(图 2)。每一级有 x_{\max} 个节点, 表示 $1 \sim x_{\max}$ 个冗余数目共有 $x_{\max} \times n$ 个节点。从第 1 级到第 n 级之间的连接在一起, 组成空间一个解, 如图 2 表示解 $(3, 2, 1, \dots, 1)$ 。

对每一级设置 1 个蚂蚁(也可多于 1 个), 每个蚂蚁只在本级 $1 \sim x_{\max}$ 节点之间转移, 蚂蚁在本级中第 i 个节点转移到本级中第 j 个节点的概率 p_{ij} 为:

$$p_{ij} = \frac{\tau_{ij}}{\sum_{j=1}^{x_{\max}} \tau_{ij}} \quad (4)$$

τ_{ij} 理解为第 j 级第 i 个节点的吸引强度。

更新方程为: $\tau_{ij}^{new} = \rho \tau_{ij}^{old} + Q$ 。

式中, ρ 表示强度的持久性系数, 一般取 0.5 ~ 0.9 左右, Q 为一正常数。

η_{ij} 表示目标函数差, $\eta_{ij} > 0$, 目标函数值降低, 蚂蚁从第 i 个节点转移到第 j 个节点, $\eta_{ij} \leq 0$, 蚂蚁维持原状。

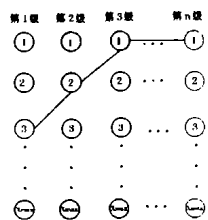


图2 可靠性网络图

解可靠性优化的蚁群算法如下:

- (1) $nc \leftarrow 0$ (nc 为循环次数), 给 τ_{ij} 矩阵赋相同的数值, 给出 Q 、 ρ 的值;
- (2) 将 n 个蚂蚁置于每一级的第一个节点 (相当于给出初解 $(1, 1, \dots, 1)$);
- (3) 对每个蚂蚁按转移概率 p_{ij} 选择下一个节点, 如目标函数差 $\eta_{ij} > 0$ 就转移到该节点;

- (4) 按更新方程修改吸引强度, $nc \leftarrow nc + 1$;
- (5) 若 $nc >$ 规定的循环次数, 记录当前蚂蚁的位置 (当前的解), 停止运行; 否则转 (3)。

6 算例分析

以一个实例来比较三个算法, 已知 5 种部件的可靠度和费用分别为 $p_1 = 0.96, p_2 = 0.93, p_3 = 0.85, p_4 = 0.80, p_5 = 0.75, c_1 = 3$ 元, $c_2 = 12$ 元, $c_3 = 8$ 元, $c_4 = 5$ 元, $c_5 = 10$ 元, 要求 $R_0 = 0.9$ 。

模拟退火算法参数初始化 $X_0 = (4, 4, 4, 4, 4)$, 对大多数优化问题而言, 模拟退火算法要优于局部搜索算法, 所得近似最优解的质量也比局部搜索算法好, 图 3 是模拟退火算法的过程。

遗传算法擅长全局搜索, 以其简单通用、鲁棒性强、适于并行处理等特点。尤其适用于处理传统搜索方法难以解决的复杂问题和非线性问题, 图 4 是遗传算法的计算过程。

蚁群算法的初始值: $x_{max} = 4, \rho = 0.9, [\tau_{ij}]_{4 \times 5} = [10]_{4 \times 5}, M = 10^6, Q = 10$, 图 5 是蚁群算法的过程。图 3、图 4 和图 5 的纵坐标为费用, 横坐标为解的变化次数。

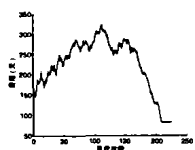


图3 模拟退火算法的计算过程

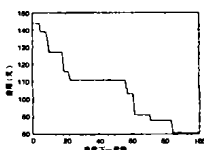


图4 遗传算法的计算过程

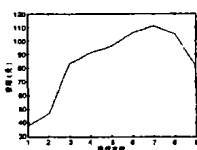


图5 蚁群算法的计算过程

各种算法各随机测试 50 次, 结果如表 1 所示。实际最优解 $(2, 2, 2, 3, 2)$, 此时总费用为 81 元。从表 1 可知, 用蚁群算法比模拟退火算法和遗传算法的效率更高, 推荐采用蚁群算法。

表1 各种算法比较

各种算法	结果比较				
	平均时间 (S)	平均值 (元)	最差解 (元)	最好解 (元)	最好解次数
退火算法	27.6	87.6	109	81	8
遗传算法	15.2	87.3	102	81	12
蚁群算法	0.2	86.2	98	81	22

7 结束语

可靠性优化问题是一个非线性整数混合规划问题, 特别当系统级数 n 很大时, 模拟退火算法和遗传算法均可以运用, 经

过比较, 蚁群算法解决可靠性优化问题比较有效。对蚁群算法稍加修改, 可解决类似的非线性整数混合规划问题。

参考文献

- [1] 曹晋华、程侃, 可靠性数学引论[M], 北京: 科学出版社, 1986: 70 ~ 74.
- [2] Tillman F A, Hwang C L, Kuo W, 刘炳章译, 系统可靠性最优化[M], 北京: 国防工业出版社, 1988: 5 ~ 15.
- [3] 刑文循、谢金星, 现代优化计算方法[M], 北京: 清华大学出版社, 1999: 3 ~ 80.
- [4] 王凌, 智能优化算法及其应用[M], 清华大学出版社, 2001(10): 17 ~ 59.
- [5] E. Aarts, J. Korst, Simulated Annealing and Boltzmann Machine[M], John Wiley & Sons, New York, 1989: 10 ~ 15.
- [6] G. J. Koehler, New directions in genetic algorithm theory, Annals of Operations Research 75 (1997): 49 ~ 68.
- [7] Colomi A, Dorigo M, Maniezzo V. Distributed optimization by ant colonies [J]. Proc. of the First European Conf. On Artificial Life. Paris, France: Elsevier Publishing, 1991: 134 ~ 142.
- [8] Colomi A, Dorigo M, Maniezzo V. An investigation of some properties of an ant algorithm[A]. Proc. Of the Parallel Problem Solving from Nature Conference (PPSN'92) [C]. Brussels, Belgium: Elsevier Publishing, 1992: 509 ~ 520.
- [9] 张纪会、徐心和, “具有变异特征的蚁群算法[J]”, 《计算机研究与发展》, 1999, 36(10): 1240 ~ 1245.

(上接第 39 页)

```

for(int i=0; i<ObjectNum; i++)
    ObjectList[i] -> Send(); // 发送
} else if(ret == WAIT_TIMEOUT) // 超时
{
    ..... // 超时处理
}

```

5 结论

插件技术通过添加插件的方式来扩展软件系统的功能, 以适时满足用户的需要。靶场遥测数据传输软件存在需要多样化和不确定性等特点, 传统的软件实现技术难以适应靶场发展的需求, 应用插件技术正好可以将需求的多样化和不确定性细节屏蔽在插件的内容具体实现上, 对应用程序提供透明的行为, 使得该软件的开发实现了模块化、标准化、易升级、易扩展。

参考文献

- [1] 马光、高中文, “插件技术在电能计量费报表中的应用[J]”, 《哈尔滨理工大学学报》, 2001, 6(6): 80 ~ 83.
- [2] 刘蕴才主编, 遥测遥控系统 (上册) [M], 北京: 国防工业出版社, 2000.
- [3] 陈以恩主编, 遥测数据处理 [M], 北京: 国防工业出版社, 2002.
- [4] 卢小兵, “一种基于对象的软件插件技术[J]”, 《华东地质学院学报》, 2002, 25(2): 178 ~ 180.
- [5] 刘毅、陈慧蓉、荣钢, “Plug-in 结构应用程序设计[J]”, 《计算机应用》, 2002, 22(4): 9 ~ 11.
- [6] 潘爱民, COM 原理与应用 [M], 北京: 清华大学出版社, 1999.
- [7] 张海藩, 软件工程导论 [M], 北京: 清华大学出版社, 1998.
- [8] 李多多等著, Windows 2000 编程 [M], 北京: 人民邮电出版社, 2000.
- [9] Microsoft Corp, MSDN Library Visual Studio 6.0, 1998.