

# 利用遗传算法解决非线性系统优化问题<sup>\*</sup>)

李献礼

(长江师范学院现代教育技术中心 重庆 408003)

**摘要** 处理非线性最优化问题的常规方法是采用启发式策略,但这些启发式算法多数只能得到局部最优解。本文结合最优化理论及遗传算法,提出了一种新的优化方法,并通过用例对此方法进行了验证。最终实现系统的全局最优解。

**关键词** 函数优化,遗传算法,适应度函数

## Optimize of NonLine System Based on Genetic Arithmetic

LI Xian-Li

(Center of Modern Educate Technique, Yangtze Normal University, Chongqing 408003)

**Abstract** Optimize of not Line Control System to turn the normal regulations method of problem is the adoption in A-Plre type strategy, but these can get partial superior solution. This text combines superior turn theories and heredity calculate way, put forward a kind of new excellent turn a method, and passed to use an example to carry on a verification to this method. I am end to carry out the superior solution in the overall situation of system.

**Keywords** Optimize of function, Genetic arithmetic, Orientation degree function

### 1 函数优化问题的描述

函数优化问题可以描述为:

$$\max_{x \in S} f(x) \quad (1)$$

$$\text{或: } \min_{x \in S} f(x) \quad (2)$$

其中  $S \subset R^n$  称为搜索空间,  $f: S \rightarrow R$  称为目标函数, (1)式描述的优化问题称为极大化问题, (2)式描述的称为极小化问题。

**定义 1** 对最优化问题(1)式, 设  $x' \in S$ , 若存在  $\delta > 0$ , 使得当  $x \in S \cap \{x: \|x - x'\| < \delta\}$  时有:

$$f(x') \geq f(x) \quad (3)$$

则称  $x$  是  $f$  在  $S$  上的一个局部最优点, 解  $f(x')$  称为一个局部最优值。若(3)式中的严格不等式成立, 则  $x'$  和  $f(x')$  分别称为严格局部最优点和严格局部最优值。

**定义 2** 对最优化问题(1)式, 设  $x' \in S$ , 若存在  $\delta > 0$ , 使得对任意  $x' \in S$  都有:

$$f(x') \geq f(x) \quad (4)$$

则称  $x'$  是  $f$  在  $S$  上的一个全局极大点或整体最优点,  $f(x')$  称为整体最优值。若(4)式中的严格不等式成立, 则  $x'$  和  $f(x')$  分别称为严格整体最优点和严格整体最优值。

### 2 遗传算法

遗传算法是模拟自然选择和遗传的随机搜索算法。该算法多用来作为问题求解和最优化的工具。同时它又是一种迭代算法。它在每一次迭代时都拥有一组解答。这组解答最初是随机生成的。在每次迭代时又有一组新的解答由模拟进化和继承的遗传操作生成。每个解答都由一个目标函数—适应度函数给予评价, 而这一过程不断重复, 直至达到某种形式的收敛。新的一组解答不但可以有选择地保留一些目标函数值

较高的旧的解答, 而且可以包括一些经由其它解答结合而得的新解答。

遗传算法中所求问题的一个解通过编码被映射到一个染色体符号串, 并由此构成由问题解空间到染色体符号串编码空间的一个映射, 进而把问题解空间中的搜索转化为染色体符号串编码空间的搜索。染色体由基因构成, 基因担负了个体的某种性状, 它通常是染色体符号串的一个在特定部位上、具有特定长度的子串。典型遗传算法步骤如下<sup>[1~3]</sup>:

1) 初始化, 随机生成一个符号串群体  $P(0)$ 。

2) 基于适应度函数( $f$ )对第  $t$  代群体  $P(t)$  中的符号串进行评价。

3) 应用一组遗传操作(繁殖、交叉、突变)生成一组新的符号串群体  $P(t+1)$ 。

4) 重复步骤(2)和(3)直到解答收敛。

在应用遗传算法之前, 当然首先要确认算法的收敛性, 即执行算法可以得到最(次)优解。我们以二进制编码为例, 染色体符号串每位上如果确定则可以为 1 或 0; 如果不确定则记作 \*。现在, 我们把这样的串称为“模板”。模板可以用两个值来刻画: 阶和定义长度。其中阶是模板  $S$  中 0、1 出现的次数, 以  $\sigma(S)$  表示。模板  $S$  中的定义长度是其中第一位和最后一位之间的距离, 记作  $c$ 。采用如下符号表示:  $n$  是二值符号串的位数;  $P(t)$  是在  $t$  时刻的符号串群体;  $m$  是  $P(t)$  中符号串的个数, 即群体大小;  $f(s)$  是符号串  $s \in P(t)$  的适应度的值;  $F(t) = \sum \text{if}(s_i) \quad s_i \in P(t)$  是  $P(t)$  中所有符号的适应度的和;  $F(S) = \sum \text{if}(s_i) \quad s_i$  与  $S$  匹配, 是  $P(t)$  中所有与模板  $S$  匹配的符号串的平均适应度值;  $\xi(S, t)$  是  $P(t)$  中与模板  $S$  匹配的符号串个数的期望值;  $P_c$  是交叉概率;  $P_m$  是突变概率。

可证明下式成立:

$$\xi(S, t+1) \geq \xi(S, t) \frac{F(S)}{F(t)} [1 - pc - \frac{\delta(s)}{n-1} - pm^n(s)]$$

<sup>\*</sup>) 基金项目: 重庆市教委科学研究项目(项目合同号: KJ071306)。李献礼 副教授, 研究方向: 计算机网络应用, 数据挖掘。

即:短的、低阶的、高于平均的模板在后代中获得呈指数增长的扩散。由此可知,遗传算法是可以收敛并得到较优化的解的。

### 3 遗传算法用于非线性控制系统优化问题<sup>[4]</sup>

#### 3.1 算法的基本流程

1)遗传算法控制参数的初始化,并确定编码;2)译码;3)对非线性系统的用例测试;4)求目标函数;5)求适应值函数;6)选择;7)杂交;8)变异;9)群体更新;10)判断是否满足条件,若满足,转向下一步;否则转向2步;11)得到最优解;12)程序结束。

#### 3.2 遗传算子的选择

选择算子在父代群体中选出父体和母体。生物界中,父母亲素质比较高的其后代素质高的概率也大。模拟这种现象通常采用轮赌算法。流程如下:

```
sum=0; i=0;
wheelposition=rand()*fitness;
for(sum<wheelposition && i<pop-size)
{i++;
if(i>=pop-size)
{sum=0; i=0;
Wheelposition=rand()*fitness;}
j=rand()*pop-size;
sum+=fitness[j];}
return j;
```

#### 3.3 交叉算子

交叉算子起全局搜索的作用。交叉算子通常有单点交叉、双点交叉、多点交叉等等。在本算法中,采用单点交叉和双点交叉比较合适。为了模拟此现象,我们引入交叉因子pc。其基本流程如下:

```
if(random(pc))
cross1(mother,father);
else if(random(pc))
cross2(mother,father);
else
copy(mother);
copy(father);
```

//random()函数中,产生一个0到1的随机数,若小于pc,则返回1,否则返回0。

#### 3.4 变异算子

变异算子起局部搜索的作用。生物界中,父母的染色体交叉后产生后代个体的染色体雏形,这个雏形在成长过程中会发生基因的变异,正是这种变异使得下一代的群体中会出现各种特征的个体。另外,生物界中并非每个基因都会变异,模拟此现象,引入变异因子pm。流程如下:

```
while(all frequentpoint)
{if(flip(pm))
mutate(frequentpoint);}
```

## 4 关键问题及分析

#### 4.1 确定编码方案

编码方案对算法的性能、效率等产生很大的影响。在求解高维或复杂问题时常常使用实数编码。而且实数编码较易引入非线性控制系统领域内的知识。搜索空间S上的每个点x可表示为一个n维实向量,其中n表示遗传算法优化参数的个数。

#### 4.2 控制参数初始化的确定

选择一个整数M作为群体规模的参数,然后从S上随机地选取M个点 $x(i,0)$ , $i=1,\dots,M$ ,这些点组成初始群体 $P(0)=\{x(1,0),\dots,x(M,0)\}$ ( $M=20\sim 100$ )。杂交概率越高,群体中串的更新就越快;而杂交率过低,搜索可能会停滞不

前。杂交率的经验取值一般为 $P_c=0.60\sim 0.95$ 。变异是增加群体的多样性,变异概率经验取值一般为 $P_m=0.001\sim 0.01$ 。

#### 4.3 确定适应值函数

适应值用来区分群体中个体的好坏,适应值越大的个体性能越好,反之,适应值越小的个体性能越差。遗传算法正是基于适应值对个体进行选择,以保证适应值好的个体有机会在下一代中产生更多的子个体。

#### 4.4 确定选择策略

优胜劣汰的选择机制使得适应值大的解有较高的存活概率,这是遗传算法与一般搜索算法的主要区别之一。对每个个体 $x(i,k)$ ,其生存概率为:

$$p_i^k = \frac{F(x(i,k))}{\sum_{j=1}^M F(x(j,k))}$$

其中k表示代数,则每个个体的繁殖量为 $N_i = \text{round}(P_i^k \cdot N)$ ,其中 $\text{round}(x)$ 表示与x距离最小的整数。显然,个体 $x(i,k)$ 的生存概率 $p_i^k$ 越大,繁殖量 $N_i$ 也越大,进行交配的机会也就越多。

#### 4.5 设计遗传算子

遗传算子包括繁殖算子、杂交算子和变异算子。在非线性和控制系统参数优化中,由于采用了实数编码方案,设计遗传算子时就应以实数编码方案为前提。在实数编码时采用离散杂交方式。在实数编码时,变异算子采用均匀性变异算子。

#### 4.6 确定算法的终止准则

由于遗传算法无法用传统的方法来判定算法的收敛与否以终止算法,通常是预先规定一个最大代数或算法,在连续多少代以后解的适应值没有明显改进时,即终止。

## 5 测试用例<sup>[5]</sup>

已知一非线性控制系统,如图1所示。

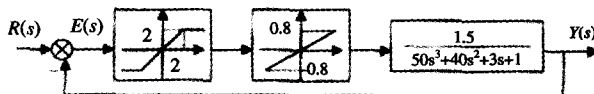


图1 非线性系统结构框图

该系统的单位阶跃响应如图2所示。

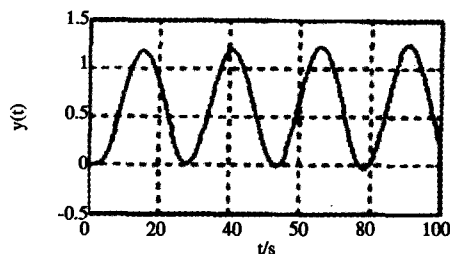


图2 非线性控制系统单位阶跃响应

从系统的单位阶跃响应可以看出,这是一不稳定的振荡非线性系统。对这一不稳定的非线性控制系统,通常采用PID调节器进行校正,设PID调节器的传递函数为:

$$C_{PID}(s) = K_p \left( \frac{1+T_I s}{T_I s} + \frac{T_D}{T_I} \right)$$

其中, $K_p$ , $T_I$ , $T_D$ 分别为PID调节器比例、积分、微分的寻优参数。用MATLAB测试时,将PID调节器设计为比例、积

(下转第232页)

息的传输不仅仅与本机有关系,因此,设计的多 Agent 系统中信息传输子系统如图 3 所示。

如图 3 所示, Agent Space 是整个多 Agent 系统中 Agent 的生存空间,它可以是跨平台的; Agent Container 是本地机 Agent 容器,本地机中可以有多个 Agent 容器,也可以只有一个 Agent 容器;在多 Agent 平台中,每一个 Agent 都有一个全局的名字,其名字的组成为“Agent 本地名/Agent 容器名@主机名:端口号”; Information Router 是信息路由,由它对 Agent 所发送信息的目标地址进行解析,并发送到相应的信息接收 Agent,同时信息路由还具有管理本 Agent Container 内 Agent 的功能。

### 3.2 Agent 信息传输机制

在一个 Agent Container 内由 Information Router 管理一个公用数据结构,它是一个 Agent 信息缓冲池,每个要发送信息的 Agent 只需成功申请一个信息缓冲区,并填上相应的内容即可。同时每个 Agent 都带有一个接收信息用的私有信息队列,该私有信息队列只允许其本身和 Information Router 访问。在一个 Agent Container 内 Information Router 和 Agent 之间的信息交互如图 4 所示。

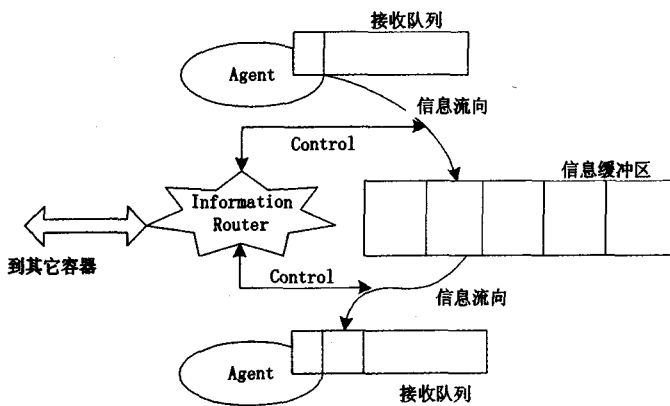


图 4 Information Router 和 Agent 之间的信息交互

从图中可以看出,如果一个 Agent 要发送信息,首先在 Information Router 的控制下向信息缓冲池申请一个缓冲区,成功之后填上相应的要发送的信息,Information Router 发现缓冲区中有待发信息,则将该待发信息的目标地址提取出来进行分析,如果目标地址是在同一个容器中,则直接将该信息转发到目标地址;如果是在同一主机但非同容器,则将其转发到相应容器的信息路由;如果不在同一主机,则找到目标地

址的主机地址并转发信息到目标机器的相应容器的信息路由。

### 3.3 Agent 信息对象设计

Agent 信息对象是 Agent 之间交流的语言,也是 Information Router 能将信息传递到目标 Agent 的依据,因此它的设计应遵循一定的规范。回顾前面曾谈到 Agent 之间的通信应能使 Agent 对所传递的信息有共同的理解,能对传递的信息进行理性化的处理,这包括语气、语用等方面。目前对于 Agent 通信语言有多种,如 KQML, FIPA 等,根据某一种规范的 Agent 通信语言,可以设计 Agent 信息对象,如表 1 所示。

表 1 Agent 信息对象

SpeechActType; <text>
Content; <statement>
Sender; <Agent global name>
Receiver; <Agent global name>
Ontology; <Ontology>

其中: SpeechActType 表示语气方面的信息,包括告知 (INFORM)、询问 (QUERY)、拒绝 (DENY) 等, Content 是具体的信息,它也根据领域特性有具体的格式, Sender 和 Receiver 指定了通信的发起者和接收者, Ontology 则指明了 Agent 应以何种方式来理解信息的具体内容,它指明了对 Content 的解析方法。

结论 理性的 Agent 是智能的 Agent, 在多 Agent 系统中, Agent 之间的通信不单是信息的传递问题, 还有 Agent 对所传递信息的共同理解。本文在分析 Agent 通信模型的基础上, 详细讨论了在多 Agent 系统平台的开发过程中如何设计及实现信息传输子系统, 提出了一种共享数据结构与私有数据结构相结合, 设置信息路由的方法来实现多 Agent 系统中 Agent 之间的通信。该方法在实践中取得了较好效果。

### 参考文献

- 1 Wooldridge M 著. 石纯一, 张伟, 徐晋晖译. 多 Agent 系统引论 [M]. 电子工业出版社, 2003, 10
- 2 Kraus S. Negotiation and cooperation in multi-agent environments. Artificial Intelligence, 1997, 94(1): 79~97
- 3 Shehory O, Kraus S, Yadgar O. Emergent cooperative goal-satisfaction in large scale automated agent systems. Artificial Intelligence, 1999, 110(1): 1~55
- 4 Kraus S, Sycara K, Evenchik A. Reaching agreements through argumentation: A logical model and implementation. Artificial Intelligence, 1998, 104(1): 1~69
- 5 Henze N, Nejd W. Knowledge Modeling for Open Adaptive Hypermedia
- 6 Brenner W, Zarnekow R, Wittig H. Intelligent Software Agents: Foundations and Applications. Springer-Verlag Berlin Heidelberg, 1998. 79~111

(上接第 218 页)

分、微分变量的 S 函数, 这样可以方便遗传算法的求解。设 PID 调节器给定最优化初始值为  $K_p = 0.63$ ,  $T_I = 0.0504s$ ,  $T_D = 1.9688s$ 。对于这组初始值, 可得系统的单位阶跃响应。设群体规模  $N = 10$ , 杂交概率  $P_c = 0.25$ , 变异概率  $P_m = 0.001$ , 最大代数  $k = 10$ , 因此可得 PID 调节器最优化参数值为  $K_p = 3.6232$ ,  $T_I = 0.1175s$ ,  $T_D = 16.9950s$ 。

从上述测试结果可以看出, 采用遗传算法进行非线性控制系统参数优化时, 系统的超调量较低、上升时间较短、调节时间较短。

小结 采用遗传算法对非线性控制系统参数进行优化时, 系统具有超调量较低、上升时间较短、调节时间较短的特

点, 且系统的稳定性和快速性都较好。

### 参考文献

- 1 Murphey R A, Pardalos P M, et al. Frequency Assignment Problems, Handbook of combinatorial optimization, Kluwer Academic Publishers [S], 1999
- 2 Vittorio M, Antonella C. An ANTS Heuristic for the Frequency Assignment Problem[S]. <http://www.csr.unibo.it>
- 3 Bater J, Jeavons P, Cohen D. Are there optimal reuse distance constraints for FAPs with random Tx placement? [S]: [CSD-TR-98-01]. CS Royal Holloway Uni. Of London, 1998
- 4 Aardal K I, et al. Algorithms for Frequency Assignment Problems [S]. CWI Quarterly, Vol9(1&2), 1996
- 5 郑志蕴. 利用神经网络进行无损检测的探讨[M]. 微计算机信息, 2002, 18(6): 27~29