为了使系统能识别案例记录中与待测对象性质无直接关系的证据，我们可以通过以下步骤筛选并淘汰这类证据：

步骤一：建立所有待测指标的相关证据集合的并集。凡不在该集合中的证据与任何待测指标均无关，可以舍弃；

步骤二：a)对于证据推理原理自学习算法生成的mass函数，设有阈值 $T(0\leqslant T\leqslant 1)$，若 $\max(m(./e_j))\leqslant T$ 且 $\max(m(./\neg e_j))\leqslant T$；即无论 e_j 是否出现，对推理结果都无重大影响，则认为 e_j 属无效证据，可以剔除。b)对于证据推理模式自学习算法生成的规则"$e_i\rightarrow h_j$"的信任区间$[B(i,j),L(i,j)]$，设有阈值 $T(0\leqslant T\leqslant 1)$，若 $\max(L(i,j))\leqslant T(h_j\in D)$，则一定有 $\max(B(i,j))\leqslant T$；即 e_j 任何假设都无重大预见性，则认为 e_j 属劣质证据，可以剔除。

步骤三：若一个证据可以由其它证据推得，则该证据属冗余证据，应删除。这时可将待删除证据作为假设，其它证据作为求解该假设的依据。设有阈值 $T(0\leqslant T\leqslant 1)$，若依自学习算法求得待删除证据的测试样本集求解精度 $S(T)>T$，则删除该证据。

为了保证证据推理的有效性，经筛选的证据集最好是完备的。所谓完备是指通过现有的证据集合可以确定待测假设集合。如果满足这一条件，则证据推理方法就可视为完全可信的假设集求解办法。如果证据集不完备，就应提醒系统有进一步搜集证据的必要，判断一个证据集是否完备，对于提高系统的自学习水平和提供尽可能好的指标估计是十分必要的。我们可以根据下面两个简单原则来发现一些证据集不完备的情形。

原则一(无冲突原则)：任何对于假设进行测度的完备集均应满足在相同的证据条件下不会出现冲突的假设采样。我们可以从对某个假设测度的样本集中选出所有相同证据条件的样本，若这些样本中对假设的案例采样有较大差异(可设定一可行的阈值来判断)则认为证据集不完备，须采集对该假设测度的其它证据。

原则二(推理成功率下限原则)：经过样本集训练得到的证据推理方法应满足一定的推理成功率下限。对于一个特殊的例子，设证据集为$(e_1,e_2,\cdots,e_m)$，$e_i(1\leqslant i\leqslant m)$的取值为"出现"或"不出现"，即为二值变量。则此证据集的样本集最大样本个数为 2^m，此时有：

(1)若假设集可构成的样本集最大样本个数超过 2^m，则证据集不完备。

(2)若已由 k 个不同训练样本进行自学习，则理想情况下，自学习样本可以经证据推理重演，即证据推理成功率下限为 $k/2^m$。如果对测试样本集的测试命中率$<k/2^m$，则可认为证据集不完备。

(3)还可以由系统得出对 k 个训练样本自学习后的推理命中率下限，做为判别完备性的依据。

参考文献

[1] 傅京孙、蔡自兴、徐光，人工智能及应用，清华大学出版社，1987

[2] 张文修，不确定性推理原理，西安交通大学出版社，1994

[3] 张文修、陈雁，合情推理与发现逻辑，贵州科技出版社，1994

[4] 孙波、袁慧萍、李怀祖，管理专家系统与情境研究，计算机科学，6(23)1996

(上接第 50 页)

0.25～0.75。P_m 太大将产生过多的预测模型串，P_m 太小又会导致不产生新的预测模型串，建议取值 0.01～0.20。

结束语 本文研究了通过遗传算法寻找最佳的预测模型的方法。笔者仅以环境质量预测为例，对 Brown、Horton、Prati、Nemerow 等大气评价模型及有关数据进行二进制编码后，使用本遗传算法寻找到不同污染情形下的预测模型，具有相当的适用价值。经过研究笔者认为遗传算法特别适合于结构复杂的非线性问题，并能对大多数的问题给出满意的解，而且比其它算法更容易实现。但如何将问题进行二进制编码值得进一步研究，这将直接影响问题求解的精度和遗传算法收敛的速度。

参考文献

[1] 张晓缋等，一种新的优化搜索算法—遗传算法，控制理论及应用，22(3)1996

[2] 韩桢祥等，模拟进化优化方法及应用，计算机科学，22(2)1995

[3] 王丽微等，遗传算法的收敛性研究，计算机学报，19(10)1996

[4] 张玲等，统计遗传算法，软件学报，8(5)1997

[5] 陈恩红等，基于遗传算法的概念学习中的约束满足预处理方法，计算机研究与发展，34(7)1997

计算机科学 1998 Vol. 25№. 2

预测模型获取的遗传算法研究

On Genetic Algorithms for Obtaining Forecasting Model

余建桥

（西南农业大学计算机系　重庆 400716）

摘　要　This paper advances a Genetic Algorithms in order to obtain Forecasting Model. Using the mighty searching power of Genetic Algorithms, we will get Forecasting Model which is fittest to the objective facts.

关键词　Genetic algorithms, Forecasting model

1. 引言

预测活动是国民经济各个领域及现代生活中常见的行为。如经济增长指标预测、国民产值预测、生态环境预测、环境质量预测、农作物产量预测、虫害预测、人口增长预测、商业行情展望等等。目前使用较多的预测方法是：先选定某种数学模型，再结合实测数据，求出模型中参数，得到具体的预测模型——数学公式，用以预测未来某事件或活动的发展趋势。这种方法的问题是：选定不同的模型，即使是使用同一组数据，其预测结果也可能会不同。那么，哪一种预测结果以及相应的模型更可信、更贴近客观事实呢？

鉴于上述问题，本文采用近年来迅速发展的一种全局优化方法——遗传算法的原理和思想，设计出了一个具体遗传算法，运用遗传算法自适应寻优及智能搜索技术，获取与客观事实最相容的预测模型。

2. 遗传算法

遗传算法(Genetic Algorithms)的基本思想是基于达尔文进化论和Mendel的遗传学说。进化论认为每一物种的每个个体的基本特征被后代所继承，但后代又不完全同于父代，这些新的变化，若适应环境，则被保留下来，若不适应环境，则被淘汰。物种在这种不断的发展过程中越来越适应环境，这就是适者生存的原理。遗传学认为遗传是作为一种指令遗传码封装在每个细胞内，并以基因的形式包含在染色体中，不同的基因所产生的个体对环境有不同的适应性，基因杂交和基因突变产生的后代对环境适应性可能更强。通过优胜劣汰的自然选择，适应值高的基因结构就保存下来。

遗传算法实质上就是一种把自然界有机体的优胜劣汰的自然选择、适者生存的进化机制，以及在同一群体中个体与个体间的随机信息交换机制相结合的搜索算法。这种算法可用计算机程序实现，用以人工模拟自然界的自然选择和进化机制，并以强大的

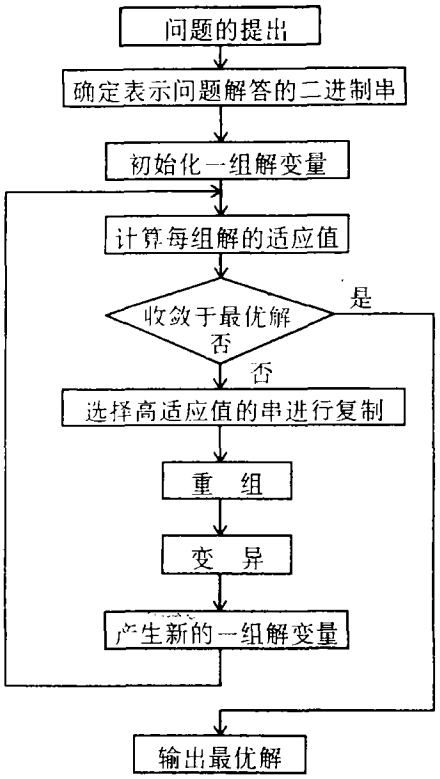

图1　遗传算法的实现过程

搜索能力解决复杂问题。遗传算法将问题的求解表示成"染色体"(用计算机编程时,一般是用二进制码串表示),将问题的所有可能解构成一群"染色体",并把它们置于问题的"环境"中,根据适者生存的原则,从中选择出适应环境的"染色体"进行复制,即再生(reproduction,也称selection),通过交换(crossover)、变异(mutation)两种基因操作产生出新的一代更适应环境的"染色体"群,这样一代代地不断进化,最后收敛到一个最适应环境的个体上,求得问题的最优解。

遗传算法的主要优点在于:问题的最优解与初始条件无关,而且搜索最优解的能力极强。它的实现过程如图1所示。

3. 预测模型获取的遗传算法

本遗传算法的思路是:建立一个事实库和一个预测模型库,从已知的事实库中,找到与事实相容的预测模型,用这些模型对未来可能发生的事件或活动进行预测。

事实库(用[F]表示)由一系列已知事实组成。每件事实表示一次实际使用过的预测模型结果。事实由实测数据及相应产生的并得以专家验证的预测模型结果构成。为了便于遗传操作的实现,将事实的实测数据(用〈Data1〉表示)和预测模型结果(用〈Result〉表示)都用二进制字符串编码,且所有事实的串长度一致。对事实库中的事实可形式化描述为:

Fact::=〈Data1〉:〈Result〉

其中〈Datal〉::=$\{0,1\}^L$,〈Result〉::=$\{0,1\}^M$

预测模型库(用[M]表示)由一系列预测模型组成,预测模型的产生形式是:

IF〈Data2〉THEN〈Model〉

其中〈Data2〉::=$\{0,1,\#\}^L$,#是0中1的通配符,〈Model〉::=$\{0,1\}^M$。

预测模型获取的遗传算法如下:

1)整理事实库[F]。即对[F]中事实进行两两匹配,删除仅是〈Data1〉部分匹配,而〈Result〉部分不匹配的事实。并对[F]中剩下的有用事实编号,F[1],F[2],F[3],…,F[n1],其中n1为[F]中的事实数。

2)初始化预测模型库[M]。给出一组具体的预测模型,并对每一具体的预测模型编号,M[1],M[2],M[3],…,M[n2],其中n2为[M]中预测模型数;确定适当大小的交换概率P_c和变异概率P_m;定义预测模型的适应值fitness[I](I=1,2,3,…,n2),并初始化:

fitness[I]←0(I=1,2,3,…,n2)。

3)从[M]中逐个取出M[I](I=1,2,3,…,n2)与[F]中的每一件事实F[J](J=1,2,3,…,n1)进行匹配。

若M[I]与F[J]完全匹配,则适应值fitness[I]←fitness[I]+1,否则fitness[I]←fitness[I]+0。

4)淘汰[M]中fitness[I]为0的预测模型M[I],则[M]库中只剩下n3(n3≤n2)个预测模型。

5)选择、保留高适应值的预测模型。

若M[I]≌M[J](I≠J,I,J=1,2,3,…,n3),且fitness[I]=fitness[J],则保留M[J],淘汰M[I];

若M[I]≌M[J](I≠J,I,J=1,2,3,…,n3),且fitness[I]>fitness[J],则保留M[I],淘汰M[J]。

6)重新组合产生新的预测模型。按一定的概率P_c从[M]中随机选择两个模型M[I]与M[J],交换彼此位串中对应的若干位的值,产生新的、不同于交换前的预测模型M[I]与M[J],如:

重组前:M[I]=0 1 1 0 1 1 0 1

M[J]=1 0 1 0 1 0 0 1

重组后:M[I]=0 0 1 0 1 1 0 1

M[J]=1 1 1 0 1 0 0 1

7)变异产生新的预测模型。按一定的概率P_m从[M]中随机地选择某一预测模型M[I],对于〈Data2〉部分将0变成1或#、1变成0或#、#变成0或1,对于〈Model〉部分将0变成1、1变成0。如:

变异前:M[I]= 0 1 1 # 1 # 0 1

变异后:M[I]= 1 0 0 1 0 0 1 0

8)在完成了淘汰、选择、重组和变异后,预测模型库[M]已换代,产生出新一代预测模型。然后进行收敛性判别,用适应值增长率与事先确定的误差精度进行比较。当适应值增长率仍较大时,返回3)重复执行,以便重新产生更适应环境的下一代预测模型;当适应值增长率较小时,说明最佳预测模型已经产生,转向9)。

9)选取具体预测模型。首先由用户输入经整理、并编制成二进制串的数据,将其与[M]库中所有的预测模型的〈Data2〉部分进行匹配。若没有一个模型的〈Data2〉部分能与之匹配,说明该数据所对应的事实至少是不常出现,本算法不提供任何结论;若匹配成功,则输出相应的预测模型,且被认为是最佳的。

用户在使用该系统时,应根据经济、生态、环境评价、虫害、人口增长、商业等不同的预测模型的特点,确定合理的二进制编码以及交换概率P_c和变异概率P_m。P_c太大会很快破坏适应值高的预测模型串结构,P_c太小会使搜索工作停止不前,建议取值

(下转第42页)