

科学计算及其软件教学丛书 / 石钟慈主编

计算智能

黄竞伟 朱福喜 康立山 © 编著



科学出版社

www.sciencep.com

TP183

科学计算及其软件教学丛书

计算智能

黄竞伟 朱福喜 康立山 编著

科学出版社

北京

内 容 简 介

本书为“科学计算及其软件教学丛书”之一。全书共4个部分,分别介绍了计算智能的4个典型代表:演化计算、群体智能算法、人工神经网络和Fuzzy计算。第1部分介绍了遗传算法、遗传程序设计、演化策略和演化规划4种主要的演化计算技术;第2部分介绍了粒子群优化和蚁群优化两种具有代表性的群体智能算法;第3部分介绍了人工神经网络的基本概念和学习算法;第4部分介绍了Fuzzy计算的基本理论及应用。

本书可作为相关专业的高年级本科生或研究生教材,也可供从事计算智能研究的相关教师和研究人員参考。

图书在版编目(CIP)数据

计算智能/黄竞伟,朱福喜,康立山编著. —北京:科学出版社,2010.6
(科学计算及其软件教学丛书)
ISBN 978-7-03-027699-5

I. ①计… II. ①黄…②朱…③康… III. ①人工智能-神经网络-计算
IV. ①TP183

中国版本图书馆CIP数据核字(2010)第094248号

责任编辑:李鹏奇 张中兴 / 责任校对:张 琪
责任印制:张克忠 / 封面设计:耕者设计工作室

科 学 出 版 社 出 版

北京东黄城根北街16号
邮政编码:100717

<http://www.sciencep.com>

北京市农林印务有限公司印刷

科学出版社发行 各地新华书店经销

*

2010年6月第 一 版 开本:B5(720×1000)

2010年6月第一次印刷 印张:15

印数:1—4 000 字数:248 000

定价:29.00元

(如有印装质量问题,我社负责调换)

《科学计算及其软件教学丛书》编委会

(以姓氏笔画为序)

主任：石钟慈

副主任：王兴华 宋永忠

编委：马富明 王仁宏 白峰杉 孙文瑜

余德浩 何炳生 何银年 张平文

陆君安 陈发来 陈仲英 林鹏

郭本瑜 徐宗本 黄云清 程晋

《科学计算及其软件教学丛书》序

随着国民经济的快速发展,科学和技术研究中提出的计算问题越来越多,越来越复杂.计算机及其应用软件的迅猛发展为这些计算问题的解决创造了良好的条件,而培养一大批以数学和计算机为主要工具,研究各类问题在计算机上求解的数学方法及计算机应用软件的专业人才也越来越迫切.

1998年前后,教育部着手对大学数学专业进行调整,将计算数学及其应用软件、信息科学、运筹与控制专业合并,成立了“信息与计算科学专业”.该专业成立之初,在培养目标、指导思想、课程设置、教学规范等方面存在不少争议,教材建设也众说纷纭.科学出版社的编辑曾多次找我,就该专业的教材建设问题与我有过多次的讨论.2005年11月在大连理工大学召开的第九届全国高校计算数学年会上,还专门讨论了教材编写工作,并成立了编委会.在会上,编委会就教材编写的定位和特色等问题进行了讨论并达成了共识.按照教育部数学与统计学教学指导委员会起草的“信息与计算科学专业教学规范”的要求,决定邀请部分高校教学经验丰富的教师编写一套教材,定名为“科学计算及其软件教学丛书”.该丛书涵盖信息与计算科学专业的大部分核心课程,偏重计算数学及应用软件.丛书主要面向研究与教学型、教学型大学信息与计算科学专业的本科生和研究生.为此,科学出版社曾调研了国内不同层次的上百所学校,听取了广大教师的意见和建议.这套丛书将于今年秋季问世,第一批包括《小波分析》、《数值逼近》等十余本教材.选材上强调科学性、系统性,内容力求深入浅出,简明扼要.

丛书的编委和各位作者为丛书的出版做了大量的工作,在此表示衷心的感谢.我们诚挚地希望这套丛书能为信息与计算科学专业教学的发展起到积极的推动作用,也相信丛书在各方面的支持与帮助下会越出越好.

石钟慈
2007年7月

前 言

中国人把计算机(computer)译为电脑,是希望用电脑来解放人脑.

探讨计算智能,提高电脑解决人脑任务的能力,把人们从繁重的脑力劳动(智能计算)中解放出来,即以电脑的计算智能来帮助人类解决大脑要解决的智能问题,一直是计算机科学家在思考的一个问题.

早在 1950 年, Turing 就梦想着电脑具有人脑能力(智能)的一天的到来. 到那时, 计算机就可以通过所谓 Turing 测试, 即 T-Test, 或简称 TT. 2003 年, Feigenbaum 提出了一种度量电脑是否具有人脑智能的新标准, 他称其为 Feigenbaum 测试, 即 F-Test, 简称 FT. 当今的电脑能通过 TT 吗? 按照 Turing 的预测, 今天的电脑本应该能通过 TT 的, 但事实上并非如此. 至于通过 FT 更是遥遥无期的事. 我们比较倾向于 Koza 提出的计算机可与人的智能媲美的 8 条具体标准(A, B, C, D, E, F, G, H). 就具体问题而言, 它正在一步步地通过这种测试, 我们把它称为 Koza 测试, 即 K-Test, 或简称 KT. 也就是说, 与人脑相比, 电脑能力的提高是一个漫长的渐近过程.

虽然目前电脑尚不具有上述标准的智能, 但是通过模拟生物和自然智能来求解复杂问题在众多自然科学和应用领域已经获得了令人瞩目的成功, 这些成功导致了计算智能的诞生. 计算智能研究的就是通过模拟生物和自然智能来求解复杂问题的方法和技术. 目前, 计算智能研究领域正在蓬勃发展, 已经形成了人工智能研究领域的一个重要研究方向.

随着科学技术的不断发展, 人类需要求解的问题的规模越来越大, 也越来越复杂. 寻求求解复杂问题的有效方法成为科学和工程等领域的迫切需要. 我们感到有必要编写一本有关计算智能的基础性教材, 目的是向高等院校相关专业的高年级本科生或低年级研究生, 系统地介绍计算智能的基础知识, 为其进一步学习和研究计算智能打下良好的基础.

本书介绍了计算智能研究领域的 4 个典型代表: 演化计算、群体智能算法、人工神经网络和 Fuzzy 计算. 这些方法都起源于对生物和自然系统的模拟. 演化计算模拟自然进化, 群体智能算法模拟生物群体的社会行为, 人工神经网络模拟生物的神经系统, 而 Fuzzy 系统起源于对生物体与环境相互作用的研究.

全书共分为 4 部分. 第一部分为演化计算, 介绍了演化计算的 4 个主要分支: 遗传算法、遗传程序设计、演化策略和演化规划. 第二部分为群体智能算法, 介绍了群体智能的两个主要代表: 粒子群优化和蚁群优化. 第三部分为人工神经网络, 介绍了人工神经网络的基本概念和学习算法. 第四部分为 Fuzzy 计算, 介绍了 Fuzzy 计算的基本理论及其应用.

在选材和叙述上, 我们尽量做到阐明各种方法的基本原理、基本方法和基本技术, 便于教学, 并通过具体的应用实例加深对内容的理解.

本书可作为系统讲授计算智能的教材, 也可选用部分内容单独讲授某种计算智能技术.

由于作者水平所限, 书中难免有不足之处, 欢迎读者批评指正.

作 者

2009 年 4 月

目 录

前言

第一部分 演化计算

第 1 章 演化计算导引	3
1.1 演化计算	3
1.2 演化算法的基本结构	4
1.3 演化算法的设计	5
1.4 演化算法的特点.....	10
1.5 演化算法的性能评估.....	11
第 2 章 遗传算法	15
2.1 遗传算法的基本结构.....	15
2.2 一个例子.....	15
2.3 遗传算法的实现技术.....	27
2.4 遗传算法的理论基础.....	42
习题	52
第 3 章 遗传算法在优化中的应用	53
3.1 无约束优化.....	53
3.2 约束优化.....	55
3.3 组合优化.....	62
习题	72
第 4 章 遗传程序设计	74
4.1 遗传程序设计框架.....	74
4.2 程序的表示.....	74
4.3 程序归纳.....	77
4.4 遗传程序设计的实现技术.....	78
4.5 应用实例.....	85
习题	91
第 5 章 演化策略	92
5.1 演化策略的基本结构.....	92

5.2 演化策略的实现技术·····	94
5.3 应用实例·····	100
习题·····	101
第6章 演化规划 ·····	102
6.1 演化规划的基本结构·····	102
6.2 演化规划的实现技术·····	104
6.3 应用实例·····	107
习题·····	107

第二部分 群体智能算法

第7章 粒子群优化 ·····	111
7.1 PSO算法的基本结构·····	111
7.2 PSO算法的实现·····	113
7.3 应用实例·····	116
习题·····	116
第8章 蚁群优化 ·····	118
8.1 ACO算法的原理·····	118
8.2 ACO算法·····	119
8.3 应用实例·····	121
习题·····	127

第三部分 人工神经网络

第9章 人工神经网络的基本概念 ·····	131
9.1 人工神经网络的特点·····	131
9.2 人工神经网络的基本原理·····	133
9.3 人工神经网络的基本结构模式·····	140
9.4 人工神经网络互联结构·····	141
9.5 神经网络模型分类·····	144
9.6 人工智能与人工神经网络·····	145
9.7 人工神经网络的应用领域·····	152
习题·····	153
第10章 人工神经网络的学习算法 ·····	155
10.1 几种基本的学习算法介绍·····	155
10.2 几种典型神经网络简介·····	159

习题	174
----	-----

第四部分 Fuzzy 计算

第 11 章 Fuzzy 计算的基本理论	179
11.1 Fuzzy 集合	179
11.2 隶属函数	182
11.3 Fuzzy 集合的特征	186
11.4 Fuzzy 集合的运算	188
11.5 Fuzzy 关系	191
习题	195
第 12 章 Fuzzy 计算的应用	196
12.1 Fuzzy 推理系统	196
12.2 Fuzzy 控制器设计	210
12.3 Fuzzy 聚类	214
习题	228
参考文献	230

第一部分 演化计算

演化计算是模拟生物进化而发展起来的一类通用的问题求解方法。演化计算采用群体搜索技术以及优胜劣汰的自然选择来指导学习和确定搜索方向,具有自组织、自适应和自学习等特点。目前演化计算技术已被广泛应用于科学、工程、经济学和社会学等领域,并取得了很好的效果。

本部分系统地介绍了演化计算的4个主要分支:遗传算法、遗传程序设计、演化策略和演化规划。本部分由第1~6章构成。第1章介绍演化计算的基本概念;第2章介绍遗传算法;第3章介绍遗传算法在优化中的应用;第4章介绍遗传程序设计;第5章介绍演化策略;第6章介绍演化规划。

第 1 章 演化计算导引

生物的进化是一种优化过程,而演化计算是模拟生物进化过程发展起来的一类通用的问题求解方法.本章主要介绍演化计算的基本思想、基本概念,演化算法的基本结构和主要构成成分,演化算法的设计和性能评估.

1.1 演化计算

生物进化是指一个种群经过漫长的时间所发生的累积变化,这些变化是由于生物体的基因变异或在繁殖期间以不同方式重组基因所产生的,而且这些变化可以被遗传到生物体的后代.

生物的进化可以看成是一个优化过程,而优化的结果是产生能够很好地适应环境的生物体.现在地球上的种类繁多、结构复杂的生物都是通过漫长的由简单到复杂、由低级到高级的进化过程而得到的优化结果.生物的进化过程也可以看成是在众多可能性中搜索“解”的一种方法.在生物中,众多可能性的集合是可能遗传序列的集合,而所要求的“解”则是能够适应环境的生物体.例如,对细菌入侵身体问题的一个令人惊奇的进化解就是哺乳动物的免疫系统.

受生物进化的启发,人们想到生物的进化机制可以被用来设计求解各类复杂问题的算法,而演化计算就是模拟生物进化而发展起来的一类通用问题求解方法.

将生物进化原则应用到问题求解的思想可以追溯到 20 世纪 40 年代.早在 1948 年,英国数学家 A. Turing 就提出了演化搜索的思想.在 20 世纪 50 和 60 年代,若干计算机科学家独立地研究了一些演化系统,这些系统将进化作为工程问题的优化工具.20 世纪 60 年代,将进化作为优化的思想在不同的地方得到了三种不同的实现.在美国,J. H. Holland 提出了遗传算法(genetic algorithm),L. J. Fogel 等提出了演化规划(evolutionary programming).与此同时,德国的 I. Rechenberg 和 H. P. Schwefel 提出了演化策略(evolutionary strategy).20 世纪 90 年代初期,美国 Stanford 大学的 J. R. Koza 又提出了遗传程序设计(genetic programming).目前,人们用演化计算(evolutionary computation)表示模拟生物进化过程求解问题的整个研究领域,而遗传算法、遗传程序设计、演化策略和演化规划是演化计算研究领域的 4 个主要研究方向.

1.2 演化算法的基本结构

演化计算所涉及的算法称为演化算法. 虽然有许多不同类型的演化算法, 但所有演化算法的一个共同特点是求解问题的过程也就是模拟大自然生物进化的过程.

自然界的一个生物种群通过繁殖产生后代, 同代或子代中的生物偶尔会发生变异, 由于食物和空间等资源有限, 种群中的生物之间必然引起竞争, 其结果是适者生存, 不适者消亡. 适者不仅获得了生存的机会, 而且能够继续繁殖后代, 自然界的生物就是根据这种自然选择法则不断地进化.

演化算法模仿自然进化过程, 在求解问题的过程中, 保持一个个体的种群, 每个个体表示问题的一个可能解. 个体适应环境的程度用一个适应函数判断, 每个个体按照适应函数来度量该个体作为问题解的好坏程度, 而度量值称为该个体的适应值, 适应值越大, 个体越好. 基于个体的适应值, 某些较好的个体被选择, 通过重组、变异等遗传算子的作用, 产生一些新的个体, 这些个体与种群中原来的个体竞争, 形成下一代种群, 继续这个过程, 直到终止条件被满足.

演化算法的基本结构如下:

算法 1.1 演化算法的基本结构

```

procedure Evolutionary_Algorithm
begin
   $t \leftarrow 0$ ;
  initialize( $P(t)$ ); evaluate( $P(t)$ );
  while(not termination condition)do
     $P'(t) \leftarrow$ parent_select( $P(t)$ );
     $P''(t) \leftarrow$ recombine( $P'(t)$ );
     $P'''(t) \leftarrow$ mutate( $P''(t)$ );
    evaluate( $P'''(t)$ );
     $P(t+1) \leftarrow$ survivor_select( $P(t) \cup P'''(t)$ );
     $t \leftarrow t+1$ ;
  end while
  return the best solution;
end

```

在算法 1.1 中, t 表示演化代数, 其初始值为 0, $P(t)$ 表示第 t 代种群. 演化算法首先产生一个初始种群 $P(t) = \{x_1^t, x_2^t, \dots, x_N^t\}$ ($t=0$), 其中, N 是一个常数, 称为种群规模. 初始种群 $P(0)$ 通常是随机产生的, 但也可以用其他启发式算法得到, 初始种群中的每个个体 x_i^0 表示所求解问题的一个可能解. 初始种群 $P(0)$ 的产生通过过程 initialize($P(t)$) 来实现. 过程 evaluate($P(t)$) 对种群 $P(t)$ 中的每个个体

进行适应值度量. 个体的适应值度量可以是简单地计算一个适应函数, 也可以是一个复杂的模拟过程. 过程 $\text{parent_select}(P(t))$ 根据种群 $P(t)$ 中个体的适应值选择出一些个体作为父体. 一般来说, 适应值较大的个体被选择的概率较大, 所选择的个体形成一个中间种群 $P'(t)$. 过程 $\text{recombine}(P'(t))$ 对中间种群 $P'(t)$ 中的个体以一定的概率进行重组, 其目的是期望通过对适应值较大的个体进行重组得到适应值更大的个体, 重组后得到中间种群 $P''(t)$. 过程 $\text{mutate}(P''(t))$ 对中间种群 $P''(t)$ 中的个体以一定的概率进行变异, 其目的是引进新的个体, 以保持种群的多样性. 过程 $\text{survivor_select}(P(t) \cup P''(t))$ 确定 $P(t) \cup P''(t)$ 中的哪些个体存活下来, 以形成下一代种群. 从初始种群 $P(0)$ 开始, 演化算法通过选择、重组、变异、存活选择等过程得到后代种群, 然后再对后代种群重复上述过程, 直到某种终止条件满足. 当算法终止后, 将所得到的适应值最大的个体作为问题的解.

演化算法的流程图可用图 1.1 来说明.

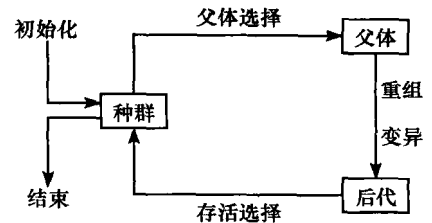


图 1.1 演化算法流程图

1.3 演化算法的设计

演化算法的基本结构虽然简单, 但在用演化算法解决实际问题时能否取得成功却依赖于演化算法的实现细节. 实现方法的不同导致不同的演化算法. 设计演化算法时, 通常需要详细指明以下构成成分:

- (1) 个体的编码;
- (2) 适应函数;
- (3) 父体选择策略;
- (4) 变化算子;
- (5) 存活选择策略;
- (6) 参数设置;
- (7) 种群的初始化;
- (8) 终止准则.

1.3.1 个体的编码

演化算法求解问题的过程就是模拟生物进化的过程. 设计演化算法的第一步是对问题的可能解进行编码, 其目的是为了能够有效地执行遗传操作. 演化算法不是直接作用在问题的解空间上, 而是交替地作用在编码空间和解空间上. 它在编码

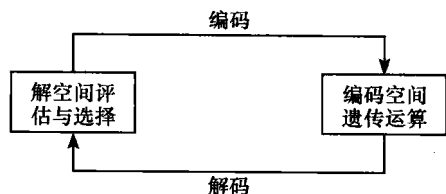


图 1.2 编码空间与解空间

空间中对个体进行遗传操作,而在解空间中对问题的解进行评估和选择,如图 1.2 所示.

编码是一个从问题的解空间到编码空间的映射.

例如,若一个优化问题的解空间由整数构成,那么一种编码方式是采用整数的

二进制编码,如整数 18 编码为 10010.

借用生物的术语,解的编码通常也称为染色体,或基因型,或个体.编码空间也称为基因型空间或搜索空间,演化算法的搜索是在基因型空间中完成的.一个染色体解码后所对应的解称为该染色体的表现型.解空间也称为表现型空间,演化算法对个体的评估和选择是在表现型空间中完成的.

在对基因型空间中的一个染色体进行评估时,需要将该染色体解码为其表现型,这就要求编码映射是可逆的,即对编码空间中的一个元素,解空间中至多有一个元素映射到它.

设计演化算法非常重要的一步是找到一种合适的编码方式.一种好的编码使我们能够设计出高效的遗传算子.演化算法的有效性和复杂性在很大的程度上依赖于所用的编码机制.

1.3.2 适应函数

在演化算法中,适应函数是区别种群中个体好坏的唯一方法,是进行选择的依据,直接影响到演化算法的收敛速度和效率.

在概念上,适应函数是一个从编码空间到实数域的函数.适应函数的作用是对基因型空间中的每个染色体指定一个实数值,称之为适应值,使得我们可以区分染色体的好坏.一般来说,一个染色体的适应值越大,表明该染色体越好.反之,一个染色体越好,其适应值也就越大.但在有些情况下,也可以这样定义适应函数,使得适应值越小,个体越好.

因为演化算法在算法终止后通常取适应值最大的个体作为问题的最优解或近似最优解,所以,尽管有多种方法设计适应函数,但一般要求最优解具有最大的适应值.除此之外,通常还要求个体的适应值与个体的质量之间有密切的联系,即个体适应值之间的差异能够反映出与个体质量的差异,或者个体质量的差异能够通过其适应值反映出来.为了说明这一点,考虑一个简单的优化问题

$$\max \sum_{i=1}^n x_i, \quad x_i \in \{0,1\}.$$

该问题的一个解为一个 n 维向量 (x_1, x_2, \dots, x_n) . 若选用问题的解本身作为其编码

的话,一种选择是直接目标函数 $f(x) = \sum_{i=1}^n x_i$ 作为适应函数. 显然,向量 (x_1, x_2, \dots, x_n) 中 1 的个数越多,该向量就越接近最优解,其适应值也就越大,反之亦然. 特别地,最优解 $(1, 1, \dots, 1)$ 的适应值最大. 另一方面,也可以取 $f(x) = \prod_{i=1}^n x_i$ 作为目标函数,这时最优解的适应值还是最大,但除了最优解 $(1, 1, \dots, 1)$ 外,其他所有解的适应值均为 0. 第一个适应函数允许逐步改进解的质量,而第二个适应函数则无此功效.

在技术上,适应函数通常是一个从解空间到实数域的函数. 演化算法按如下过程对基因型空间的一个染色体进行评估:先将该染色体解码为其对应的表现型,然后用适应函数对表现型进行评估,并将评估值作为该染色体的适应值.

在许多情况下,演化算法求解的问题是一个优化问题,这时适应函数可以直接从目标函数得到或简单地由目标函数变换而成.

1.3.3 父体选择策略

演化算法通过从当前代种群产生下一代种群的方式,使得种群不断演化,从而逐步逼近问题的最优解.

类似于生物,产生下一代个体的一个重要手段是对当前代中的个体进行重组. 父体选择是重组的基础. 父体选择的目的是期望较好个体的重组能够得到更好的后代. 若一个个体被选择通过重组产生新的个体,那么该个体称为一个父体.

并不是最好的个体总是被选择为父体. 这是因为选择压力过大,搜索会过早终止,算法容易收敛到局部最优. 当然,若选择压力过小,虽然可以保持种群的多样性,增加算法收敛到全局最优的概率,但算法收敛的速度缓慢.

保持适当的选择压力是父体选择策略应考虑的一个重要问题. 一般来说,算法的初始阶段宜采用较低的选择压力,这有利于扩展搜索空间,而在算法的终止阶段宜采用较高的选择压力,这有利于找到最好解.

有许多不同的父体选择策略. 通常,父体选择是随机的,使得具有较大适应值的个体被选择作为父体的概率较大,具有较小适应值的个体被选择作为父体的概率较小.

1.3.4 变化算子

变化算子的作用是从已有的染色体中产生新的染色体,这相当于在解空间中产生新的解.

变化算子主要包括重组算子和变异算子.