

演化计算^{*1)}

康立山 陈毓屏

(武汉大学软件工程国家重点实验室)

EVOLUTIONARY COMPUTATION

Kang Lishan Chen Yuping

(National Software Engineering Laboratory, Wuhan University)

Abstract

This is an introduction to the new research area of the evolutionary computation. First, the history of evolutionary computation is described. Then, the evolutionary computation from imitating life, a very important component of the evolution, as well as the genetic algorithms are introduced. Another component is the evolutionary computation from simulating physical phenomena, such as annealing process. Some research directions and important references in this field are mentioned.

从1987年由钱学森提出的“极度并行”计算的概念,到四年后(1991年底)IBM、CRAY等公司开始转轨,像Thinking Machines公司一样研制极度并行(massively parallelism)计算系统。现在又过去了三年,许多问题变得更明朗了,CRAY-T3D, IBMSP₂, TM CM-5E以及NEC SX-4,相继问世,现在真正进入了并行计算时代。正如我们七年前在第一届全国并行算法会议上预计的那样,1992年后,并行计算进入了它的成熟期:并行计算的标准化时期(包括并行算法与软件)。例如,软件方面的高性能Fortran (HPF)提供了一种标准化语言(如CM-Fortran)为极度并行计算系统在科学计算中的应用开辟了道路。1993年D. Hillis发出了“向并行过渡”的呼吁(见:世界科学,1994,第5期)。在极度并行化的今天,除了并行计算机本身的问题,从计算复杂性到计算语言和系统软件等要重新研究外,他认为最激动人心的是那些崭新的应用,如格子气流体流动,神经网络及演化计算。本文着重介绍与新学科,计算智能(Computational Intelligence)有关的新理论,演化计算(Evolutionary Computation)。

本文首先介绍演化计算的由来。接着介绍仿生演化计算和拟物演化计算。最后列举演化计算的一些研究方向。

* 1994年9月22日收到。

1) 国家自然科学基金、“863”计划资助项目。

(一) 演化计算的由来

从其根源来看有四个:

① 德国的演化策略 (Evolutionary Strategy, 简写成 ES) 派

代表人物: I. Rechenberg H-P. Schwefel

他们1963年在 Berlin 工业大学当学生时提出 ES. Birlin 工业大学成立了仿生与演化技术实验室。现在 Rechenberg 所在 Birlin 工业大学与 Schwefel 所在的 Dortmund 大学成了欧洲的 ES 研究中心。从1990年起,每两年召开一次国际性的学术会议“Parallel Problem Solving from Nature” (1990, 1992, 1994)。

代表专著: I. Rechenberg, Evolutions strategie, Frommann Holzboog, 1973.

H-P. Schwefel, Numerical Optimization of Computer Models, Wiley, 1981.

该学派主要研究函数优化。

② 美国的演化程序设计派 (Evolutionary Programming, 简称 EP)

代表人物: L. J. Fogel 与 D. B. Fogel

他们父子开了公司, Natural Selection, Inc. 与 ORINCON Corporation

该学派成立了 Evolutionary Programming Society, 从1992年起每年召开年会, 并出版年会论文集, 例如1994. Proceedings of the Third Annual Conference on Evolutionary Programming, World Scientific.

代表作: L. J. Fogel 等, Artificial Intelligence through Simulated Evolution. John Wiley & Sons, 1966.

D. B. Fogel, Evolving Artificial Intelligence, PhD thesis, University of California, San Diego 1992.

该学派最初是研究人工智能, 现在也用于函数优化与训练神经网络等。

③ 美国的遗传算法派 (Genetic Algorithm, 简称 GA)

代表人物: J. H. Holland, Michigan 大学;

D. E. Goldberg, Illinois 大学;

K. De Jong, George Mason 大学。

中心在 Michigan 大学, 成立了遗传算法实验室, 并成立了“国际遗传算法学会”。

从1985年起每两年召开一次 International Conference on Genetic Algorithms, 已出版了五届会议论文集(1985, 1987, 1989, 1991, 1993)。

代表著作: J. H. Holland, Adaptation in Natural and Artificial Systems, University Michigan Press (1975), 1992年 MIT Press 再版时作了扩充与修订。

D. E. Goldberg, Genetic Algorithms in Search Optimization and Machine Learning, Addison-Wesley, MA, 1989.

该学派开始是用 GA 来设计与实现健壮 (Robust) 自适应系统, 现已广范应用于搜索、优化与机器学习等。

④ 美国的遗传程序设计派 (Genetic Programming, 简称 GP)

代表人物: J. R. Koza 博士, Stanford 大学

代表著作: J. R. Koza, Genetic Programming, MIT Press, 1991. 1994 年又出版了第二册。

GP 是延伸 GA 到计算机程序领域和遗传地繁殖计算机程序的群体去求解问题。用于解决系统识别、最优控制、博弈、优化及计划等问题。最近开发的自动函数定义的灵活性使 GP 能动态地分裂一个问题为较简单的一些子问题, 解出这些子问题, 再组装原问题。实验证据显示自动函数定义减少求解问题所需的计算努力和产生一个更简单、更可理解的整体解答。

所有以上四种学派, 最主要的是 ES 与 GA, 都是从“仿生”的演化中得到启发而发展起来的, 四种方法都可以用来进行搜索、优化与学习, 解决智能计算中的问题。而且在它们的发展过程中彼此交织在一起, 已是难分难解了。从 1993 年起, 他们共同主办了一个杂志: 《Evolutionary Computation》, MIT Press. 并于今年 6 月 27—29 日在美国召开了会议 The First IEEE Conference on Evolutionary Computation. 至此, 在国际上正式形成了一门多学科高度交叉的新学科。

(二) 演化计算

我们觉得演化计算还有一个重大的分枝——拟物演化计算, 其中最重要的是基于 Boltzmann 演化策略的模拟退火算法 (Simulated Annealing Algorithm, 简称 SA), 而且它已与 GA 彼此交织在一起了。我国的黄文奇教授在“拟物”算法方面作了不少很有意义的工作。

只有将“仿生”与“拟物”两个方面都考虑到了, 才算是真正的完整的演化计算, 即 Computation from Nature.

(1) 仿生演化计算 (Evolutionary Computation from Imitating Life)

大自然的演化过程造就了一个完美无比的人类。人类在改造自然的同时, 也不断完善自身。人类的根本任务是向大自然学习, 掌握其基本演化规律以促进大自然加速向着有利于人类的方向演化。

演化计算就是学习和借鉴大自然的演化规律来解决我们面临的各种计算问题, 包括从计算系统的设计到计算系统的应用。

仿生演化策略主要有两种:

• Darwin 策略: 它出现在生物遗传过程中

1. 具有最大适应性的优良品种的自繁殖;
2. 错误繁殖导致的变异过程;
3. 自繁殖精确度的增加。

• Haeckel 策略

随着细胞组织复杂性的增长, 其发展的一个生命周期由若干阶段组成即少年期, 生长与学习期, 自繁殖与死亡期。

根据这些演化策略而设计的算法, 最典型的是遗传算法 (GA) 和演化策略 (ES), 所

以我们分别介绍一下 GA 与 ES.

J. H. Holland 七十年代初期在设计与实现自适应系统时将问题的可行解编码成二进制串(染色体)表示群体的个体,其中每一位表示一个基因,而将问题的目标函数模拟为个体对环境的适应性(fitness).并仿效遗传学中的杂交(crossover)、变异(mutation)与自然淘汰(natural selection)的法则设计了遗传算子,即杂交算子、变异算子与淘汰规则.

标准的 GA 由下述步骤组成:

Step1 初始化.随机地产生 K 个个体(固定长度的二进制串)组成第一代群体 $M(0)$.

$t := 0$;

Step2 计算适应度.对每一个体 $m \in M(t)$,计算其适应度 $u(m)$,其中 $u(x)$ 是目标函数;

Step3 计算选择率.对 $M(t)$ 中的每个个体 m ,确定其选择概率 $P(m)$,它是与 $u(m)$ 成比例的;

Step4 生产下一代.从 $M(t)$ 中按 $P(m)$ 随机地选择个体配对杂交(按事先规定的杂交率),并经过变异(按事先规定的变异率)随机地改变某些个体的某些基因,而产生下一代 $M(t+1)$,然后返回 Step2.

停机规则通常由达到指定的代数 t_{\max} 为止.

或简写为程序:

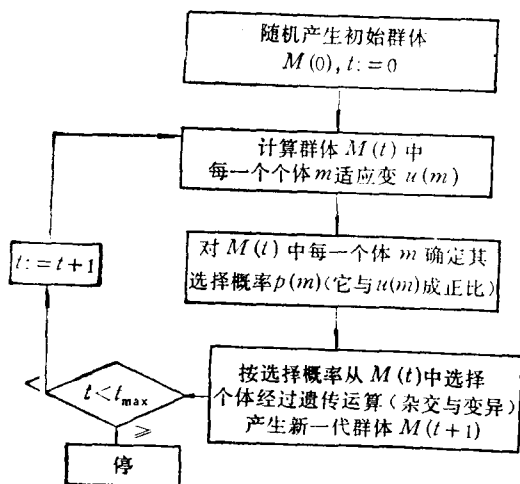
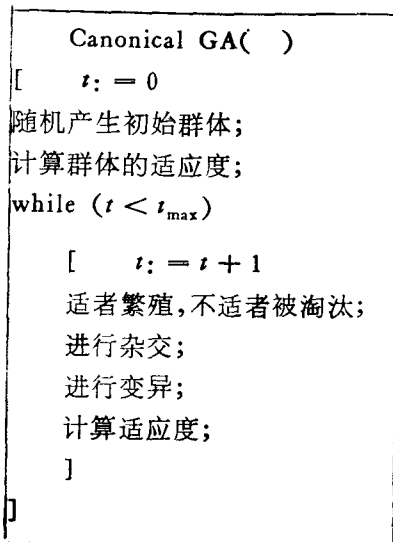


图1 标准遗传算法框图



关于 GA 的一个综合性介绍,读者可以参看[1].

武汉大学软件工程国家重点实验室从 1988 年起在 863 计划资助下开始进行计算智能基础理论——演化计算方面的研究.探索“拟物”[3]与“仿生”[4]演化计算的基本规律以及它们的结合[5,6,7],特别是它们的下述三大特点:

1. 极度并行性,即其群体可以任意分裂成相对独立的子群体,形成具有各种粒度的并行算法;

2. 通用性,它可用来解决各种类型的优化、搜索与学习等问题,有利于建立各种类型的软件环境;

3. 灵活性,易于形成各种并行计算系统(包括分布式系统、并行机系统以及人工神经网络系统等)的自适应与自组织算法.

上述这些特点,使它们有可能成为智能计算(或神经计算)的主要计算方法^[9].

1. Rechenberg 于 1963 年提出演化策略 (ES). ES 与 GA 的主要区别在于:

1. ES 主要用于求解函数优化问题,因此群体中的个体的表示为实值向量,而 GA 可广泛用于搜索、优化(包括组合优化)与学习等问题,其群体中的个体(染色体)既可以是串,也可以是树、图或别的表示形式.

2. 在演化过程中,ES 主要用变异算子,而 GA 主要用杂交算子,只小概率地使用变异算子.

3. 在淘汰过程中,GA 主要按个体的适应性比例进行选择,而 ES 则通过同一代或父子两代按锦标赛淘汰制进行选择.

以求解下述函数优化问题为例:

$$\max_{x \in Q} f(x)$$

ES 可描述如下:

群体中个体表示为一对实值向量: $v = (x, \sigma)$, 其中 $x \in R^n$ 表示搜索空间 Q 中的一个点, 向量 $\sigma \in R^n$ 是标准偏差. 由于生物变异中观测到小的变化比大的变化出现频繁. 故采用

变异: $x(t+1) = x(t) + N(0, \sigma)$ 其中 $N(0, \sigma)$ 是均值为 0, 标准偏差为 σ 的一个独立的随机 Gauss 数的向量.

如果 $f(x(t+1)) > f(x(t))$, 且 $x(t+1) \in Q$, 则

$$\text{变异成功: } (x(t), \sigma) := (x(t+1), \sigma)$$

杂交: 群体中所有的个体有相同的交配概率, 随机地选一对双亲:

$$(x^i, \sigma) = ((x_1^i, x_2^i, \dots, x_n^i), (\sigma_1^i, \sigma_2^i, \dots, \sigma_n^i)),$$

$$(x^j, \sigma) = ((x_1^j, x_2^j, \dots, x_n^j), (\sigma_1^j, \sigma_2^j, \dots, \sigma_n^j)).$$

离散杂交生产一个儿子:

$$(x, \sigma) = ((x_1^{a_k}, x_2^{a_k}, \dots, x_n^{a_k}), (\sigma_1^{a_k}, \sigma_2^{a_k}, \dots, \sigma_n^{a_k})),$$

其中 $a_k = iorj$ (按 1/2 概率), 或平均杂交生产一个儿子:

$$(x, \sigma) = \left(\left(\frac{x_1^i + x_1^j}{2}, \dots, \frac{x_n^i + x_n^j}{2} \right), \left(\frac{\sigma_1^i + \sigma_1^j}{2}, \dots, \frac{\sigma_n^i + \sigma_n^j}{2} \right) \right).$$

这样一来,参数 σ 是随着演化过程而变化的。

通常引入下述记号来表示不同的演化策略:

$(\mu + \lambda) - ES$: μ 个个体生产 λ 个儿子,然后父子两代(共 $\mu + \lambda$ 个个体)进行竞争,决赛出 μ 个优胜个体组成新一代。

$(\mu, \lambda) - ES$: μ 个个体生产 λ 个儿子,且 $\lambda > \mu$,然后子代中 λ 个个体进行竞争,决赛出 μ 个优胜个体组成新一代。

随着演化策略的发展,许多新的变异、杂交与淘汰规则出现在各种应用中。演化策略的收敛性也有了较深入的研究。由于 ES 主要研究函数优化问题,所以相对于 GA 而言,它的数学理论基础要容易建立一些。

值得指出的是,现在已经建立了一些 GA 与 ES 的程序设计环境,它们大致可以分为三类:面向算法的系统,面向问题的系统以及工具包[2]。图 2 给出了部分软件系统的名称及其作者的示意图。

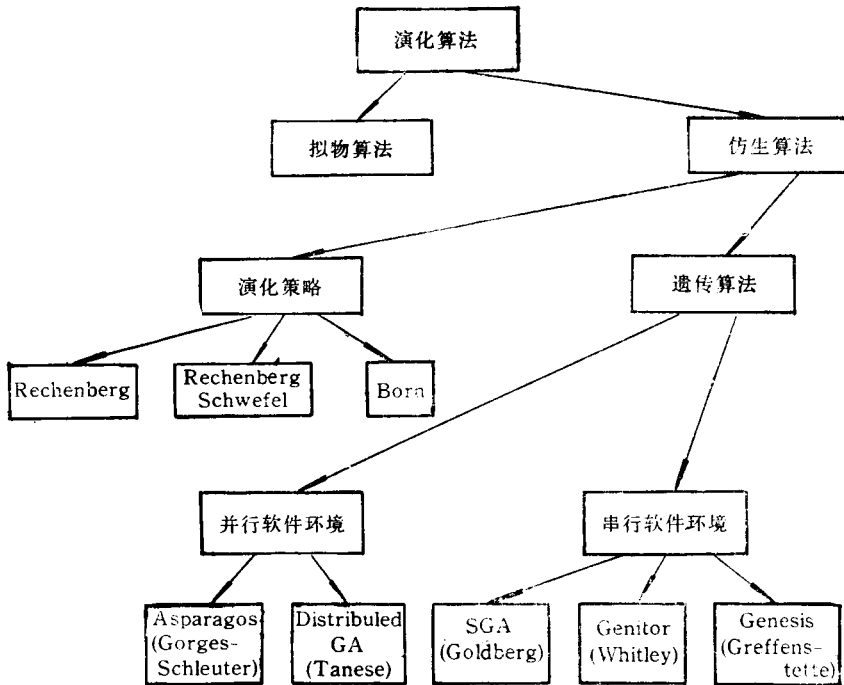


图 2 GA 与 ES 软件环境示意图

(2) 拟物演化计算

1953年, N. Metropolis 等在研究封闭盒子中的浓稠气体的行为时,提出了一种有效的随机抽样方法计算状态方程。他应用的是 Boltzmann 演化策略。

1983年 Kirkpatrick, Gelatt 与 Vecchi 等重新燃起对这种算法的兴趣,而提出了模拟退火算法 (Simulated Annealing, 简称为 SA)。见他们的论文:

Optimization by Simulated Annealing, Science, Vol. 220, No. 4598, May(1983)
671--681

若要解优化问题:

则 SA 可描述如图 3:

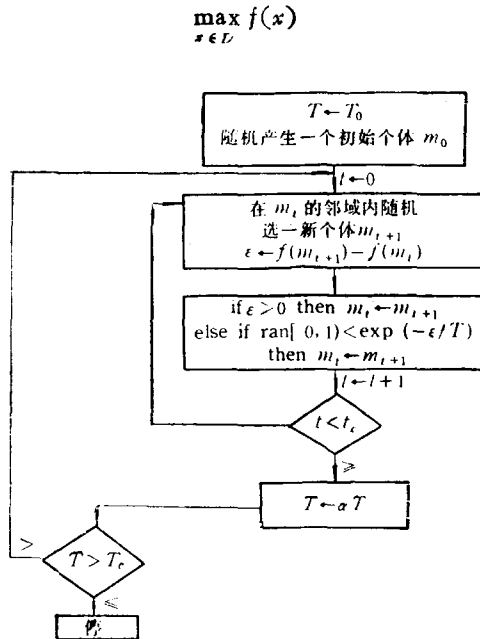


图 3

关于 SA 的详细过程、应用及其收敛性等, 请参看[4]

(三) 演化计算的研究方向

- EC 的数学基础
- 适合于 EC 的问题特征
- EC 的实现
- 演化优化
- 演化机器学习
- 演化人工生命
- 演化神经网络
- EC 用于图像处理与模式识别
- EC 用于机器人与自治系统
- EC 用于科学、工程与经济问题
- 演化 Fuzzy 系统

参 考 文 献

- [1] M. Srinivas and L. M. Patnaik, Genetic algorithms: A Survey, Computer, 27: 6 (1994), 17-26.
- [2] J. L. Ribeiro Filho, P. C. Treleaven and C. Alippi, Genetic-Algorithm programming environments, Computer, 27: 6 (1994), 28-43.
- [3] 刘勇, 康立山, 陈毓屏, 非数值并行算法(第二册)遗传算法, 科学出版社, 1995.
- [4] 康立山, 谢云, 尤矢勇, 罗祖华, 非数值并行算法(第一册)模拟退火算法, 科学出版社, 1994.
- [5] Liu Yong, Kang Lishan and D. J. Evans, The annealing evolution algorithm, Parallel Algorithms and Applications, Vol. 3(1994), 145-150.
- [6] He Jun, Kang Lishan and Chen Yongjun, Convergence of genetic evolution algorithms for optimization, Parallel Algorithms and Applications, Vol. 4 (1994).
- [7] 康立山, 陈毓屏, Jain Macleod, 陈璐鹏, 求解 TSP 的演化算法, 智能计算机基础研究'94, 清华大学出版社, 1994, 23-35.
- [8] 何军, 印鉴, 关于一般模拟退火算法的收敛性, 智能计算机基础研究'94, 清华大学出版社, 1994, 37-91.
- [9] 康立山, 陈毓屏, 遗传并行算法发展动态, 自然杂志, 15: 12(1992), 899-905.