

文章编号: 1001-0920(2007)02-0127-05

动态环境中的进化算法

王洪峰¹, 汪定伟¹, 杨圣祥²

(1. 东北大学 信息科学与工程学院, 沈阳 110004; 2. 莱斯特大学 计算机科学学院, 莱斯特)

摘要: 目前关于进化算法(EA)的研究主要局限于静态优化问题,然而很多现实世界中的问题是动态的,对于这类时变的优化问题通常并不是要求 EA 发现极值点,而是需要 EA 能够尽可能紧密地跟踪极值点在搜索空间内的运行轨迹.为此,综述了使 EA 适用于动态优化问题的各种方法,如增加种群多样性、保持种群多样性、引入某种记忆策略和采用多种群策略等.

关键词: 动态环境; 非静态; 进化算法; 遗传算法

中图分类号: TP301 **文献标识码:** A

Evolutionary algorithms in dynamic environments

WANG Hong-feng¹, WANG Ding-wei¹, YANG Sheng-xiang²

(1. College of Information Science and Engineering, Northeastern University, Shenyang 110004, China; 2. Department of Computer Science, University of Leicester, Leicester, U K. Correspondent: WANG Hong-feng, E-mail: hfwang@mail.neu.edu.cn)

Abstract: Evolutionary algorithms (EAs) are widely and often used for solving stationary optimization problems where the fitness landscape or objective function does not change during the course of computation. However, the environments of real world optimization problems may fluctuate or change sharply. If the optimization problem is dynamic, the goal is no longer to find the extrema, but to track their progression through the search space as closely as possible. All kinds of approaches that have been proposed to make EAs suitable for the dynamic environments are surveyed, such as increasing diversity, maintaining diversity, memory-based approaches, multi-population approaches and so on.

Key words: Dynamic environment; Non-stationary; Evolutionary algorithm; Genetic algorithm

1 引言

近几十年来,进化算法(EA)得到了众多学者的广泛关注已成为优化算法研究的一个热点,但大多关于 EA 的研究主要局限于静态优化问题.然而现实世界中很多优化问题是动态、时变的,它们会因为目标函数、环境参数或者约束条件的变化而随时产生变化.对于这类时变的优化问题通常并不仅仅要求 EA 发现极值点,而更需要 EA 能够尽可能紧密地跟踪极值点在搜索空间内的运行轨迹,这就需要算法能够持续地适应非静态环境中解的变化.

实际上,将进化算法应用于动态环境的研究可以追溯到 1966 年^[1],但是直到 20 世纪 80 年代中期才成为众多学者的研究热点^[2].近年来许多国际会议(如 GECCO2002, WCCI2002 和 CEC2003 等)都有进化算法在动态环境中应用方面的论文发表,特

别是在 CEC2004 会议上为动态进化优化方法开辟了相关的专题讨论.国内学者康立山等^[3]在动态系统方面也曾作过一些研究工作.

传统进化算法的目标是使种群逐渐收敛,最终获得一个满意解.这样会使种群失去多样性,而种群的多样性恰恰是有效探索整个可行空间的必要条件.因此传统进化算法在进化后期会失去对环境变化的适应能力,这是进化算法在动态环境中所面临的主要挑战.近些年来,许多学者使用了各种方法来解决这个问题,这些方法大体上可以分成下面 4 种:

1) 采取修改某些 EA 算子的策略,使 EA 能够适应环境的变化.

2) 始终避免种群收敛,保持种群的多样性,这是因为一个发散的种群能够更容易适应环境中的变化.

收稿日期: 2005-11-16; 修回日期: 2006-02-07.

基金项目: 国家自然科学基金重点项目(70431003).

作者简介: 王洪峰(1979-),男,辽宁辽阳人,博士生,从事动态进化计算方法的研究;汪定伟(1948-),男,江西彭泽人,教授,博士生导师,从事生产计划与调度、人工生命计算等研究.

3) EA 中引入某种记忆策略,使之能够重用以前的进化信息,这类方法适用于周期变化的环境。

4) 采用多种群策略,将整个种群分成若干个小种群,其中一部分用于追踪当前的极值点,另一部分继续搜索整个空间,以发现新的极值点。

2 环境变化后修改 EA 算子

当环境中的某个变化被探测到之后,EA 采用简单的重启是处理动态的一种最直接的方式。然而如果环境的变化相对较小,那么新的极值点与旧的极值点之间可能具有一定的联系。因此通常在 EA 重启时将旧种群(变化前)中的某些信息通过某种方式传递给新种群(变化后),例如将旧种群的某些个体传递给新的初始种群作为其中的一部分。

Cartwright 等^[4]证明了当 EA 重启时,旧种群中的部分个体应该保存下来,因为它们也能够新的环境中被重用。Reeves 等^[5]提出了利用前一阶段 EA 搜索到的解而得到的新初始种群会使重新开始的 EA 更快地搜索到新问题的最优解,但并没有给出算法的具体实现过程。Bierwirth 等^[6]建议校正种群中所有个体,使之满足新问题的要求,并把它们作为求解新问题的 GA 的初始种群。在与随机初始化的方法相比较^[7]时会发现,虽然解的质量稍有不同,但是速度却大为提高。Lin 等^[8]针对一个 JSSP 问题也提出了一种直接校正的方法,他们利用 Giffler-Thompson 算法把新工件加入到旧调度中,结果发现这种方法显示出十分明显的效果。Pico 等^[9]在考虑最小化拖期的处理器调度问题时采用了一种固定编码长度的置换方法。Krishnakumar^[10]提出了一种 Micro Genetic Algorithm,该算法种群很小,一旦发现收敛,则保留最好的个体,其余个体随机初始化。

然而采用重启策略也会给 EA 带来一定的挑战,它需要 EA 能够在开发和探索之间达到一种平衡:如果过多的信息被保留,那么由于种群中个体的相似性会使种群过早地收敛;反之,如果过多的信息被丢弃,那么会花费更多的时间去寻找新的极值点。

为了保持这种平衡,Cobb^[11]在传统 GA 中采用了一种过度变异的策略。当探测到环境的变化后立刻猛烈增大变异率,使得趋于收敛的种群发散。在后来的研究中,Hypermutation 被进一步检验^[12],结果发现环境变化的频率会对变异率的增大幅度有一定的影响,频率越大,Hypermutation 越强烈。Vavak 等^[13]提出一种称为可变局部搜索的变异算子:最初,只是采用较小的变异(如仅仅允许二进制编码最后几位发生变异),当种群的性能在一定时期内没有得到改善时,逐步增大局部搜索的范围(变异范围越来越大)。Gerratt 等^[14]将爬坡算法与 EA 结合在一

起提出了进化随机搜索算法,实验表明该算法比一般 GA 和爬坡算法更能适用于动态优化问题。Simões 等^[15]将细胞转化机制引入到 GA 中,提出了 TGA 和 ETGA 算法,在 0-1 时变背包问题中,TGA 和 ETGA 表现出比采用过度变异策略的 GA 更好的性能。Renato 等^[16]提出了一种基因独立变异的 GA 算法,即染色体的每一位基因具有独立的变异率,实验表明这种策略在非静态优化问题中表现出良好的性能。

3 始终保持种群的多样性

Grefenstette^[17]提出一种称为随机移民的方法,即在每一代,种群中的部分个体都会被随机产生的个体所替代。Cobb 等^[18]比较了 3 种不同变异方案在动态环境中应用:固定变异率、过度变异和随机移民的方法。结果发现,采用过度变异的 GA 在缓慢变化的环境中(低强度)表现最好,如果环境变化较大,则随机移民的方法表现会更好一些。Eriksson 等^[19]提出了一种采用“life-time adaptation”策略的 EA 算法,即在 EA 运算的整个过程中,对个体估值之前首先要采用某种适应性调节的策略对个体进行调节,以此来保证种群的多样性。

一些研究者发现在遗传运算时通过修改选择过程也能够保证种群的多样性。在静态优化问题中有一种常用的方法是使用共享策略,共享最初设计用于在并行计算过程中研究 multi-model 函数中的若干个局部最优点,其基本思想是处于环境中相同区域的个体将共享其适值,利用这种策略会使居住于低人口密度区域的个体获得比高人口密度区域的个体更多的重视,进而使得所有个体分散到适值曲线的各个峰上。

Andersen^[20]检验了共享策略对 GA 追踪运动极值点能力的影响,结果发现在缓慢变化环境中采用共享策略能够显著地增强 GA 对极值点的追踪能力,同时也发现 GA 采用表现型共享策略会比采用基因型共享策略稍好一些。文献[21]将一种称为 Minimal Representation Size Clustering 的共享策略应用于自动机器人路径识别问题中,通过对各条路径的并行调查使得机器人能够快速适应环境中的变化。

Mori 等^[22]提出的 Thermodynamical Genetic Algorithm(TDGA)的核心思想是通过一个称作“自由能量”的变量 F 来直接控制种群的多样性,并给出了 F 的计算方法。

此外在一些文献中介绍了使用特殊的染色体表现型来保持种群的多样性方法,对偶 GA^[23](DGA)就是一类这样的方法,它在染色体位串中引入了一

个元基因(通常是第一位基因),当元基因的位值为 0 时,没有什么影响,如果位值为 1,则后面所有的位值将取反. Yang^[24] 采纳了 DGA 的思想,提出一种原对偶遗传算法(PDGA),与 DGA 不同的是 PDGA 染色体中没有引入元基因,而是在进行交叉变异等遗传运算之前首先对种群中的部分染色体进行对偶运算,文献[25]给出了对偶运算的染色体选择方法,并且发现在某些动态函数中 PDGA 取得了比 DGA 更好的性能.

4 基于记忆的方法

所谓记忆就是允许 EA 能够存储曾经获得的好解,并在需要的时候重用这些解,利用记忆能够使得环境变化时 EA 迅速地作出反应. 通常记忆可以分为两种:利用冗余表示的隐式记忆和显式记忆,直接引入某种记忆机制,给出某种存储策略和重用策略. 一般而言,记忆策略较适用于类似周期变化的环境,此外利用冗余表示还能够使种群收敛速度放慢,丰富基因池中染色体的种类.

最具代表性的隐式记忆策略是二倍体的表示方法. Goldberg 等^[2] 提出了基于二倍体和基因显性机制的遗传算法,在动态背包问题的应用中,该方法比简单 GA 表现出更好的性能. Ng 等^[26] 提出了一种四等位基因的二倍体方案,为了更快地适应环境的变化,采用一种基因显性变化机制,当一个个体的适值减少超过 20% 时所有的等位基因位值都发生转换(即从显性转化成隐性,反之亦然),并在实验中发现这种二倍体方法要优于单倍体和三等位基因的二倍体方法. 文献[27]介绍了另一种二倍体的 GA 算子,为了保持种群的多样性,在这种 GA 中采用了 3 种策略:适应性的基因显性机制、减数分裂的繁殖策略和考虑个体年龄的替代方法. 在对一种动态背包问题的实验中,这种二倍体的 GA 取得了较好的效果.

Hadad 等^[28] 提出了多倍体的方法,将一个统治变量作为个体的一部分,在动态背包问题中,该方法能够获得与二倍体方法一样好的性能,而且发现多倍体的方法更适用于变化频率较高的环境. Ryan^[29] 使用了一种附加多倍体方法,附加的基因用于决定个体表现型的某个特性,在某些动态问题中,这种方法取得了优于 Ng 和 Wong 方法的结果.

从目前的研究来看,多倍体的方法在周期性变化环境中能够比较有效地适应环境的变化,但是在非周期性变化环境或者变化周期的状态数较多的环境中是否能够有效应用值得研究.

此外, Dasgupta 等^[30] 提出了一种基因分级结构的方法. 在这种表示方法中,上层基因能够对下层基

因进行动员或者解除动员,其复杂的分级结构基因会比多倍体的方案具有更多的冗余信息. 在时变背包问题中,使用了相对简单的两层基因结构,取得了比简单 GA 更好的效果.

尽管上述各种隐式记忆的方法能够使 EA 间接地存储一些有效信息,但并不确定算法能够有效地使用这些信息. 而显式记忆则能够直接存储某些特殊信息,并在后期的进化中将有用的信息重新引入到种群中.

Louis 等^[31] 对一个 open shop 动态调度问题采用了以下方法:每迭代一段时间,当前种群的最好解都会被保存下来,当环境发生变化后,GA 重新初始化时将记忆中的部分个体(5%~10%)传递给种群,其余个体随机产生. 结果发现,采用这种策略的 GA 比完全随机初始化的 GA 能够更快地获得更好的结果,但是当传递过多的个体(50%~100%)时,这种方法是失败的.

Branke^[32] 将整个种群分成两个部分:记忆种群和搜索种群. 一个种群从记忆获取信息,不断对记忆进行更新;另一个种群负责搜索整个空间,并把搜索的结果按照一定的策略存储到记忆中,但不从记忆中获取任何信息. 而且为了保证探索能力,当环境每次变化后搜索种群都进行重新初始化. 这种基于策略的 GA 在移动峰函数中能够获得比无记忆的 GA 更好的性能.

Trojanowski 等^[33] 通过引入额外记忆对每个个体进行扩展,使其能够获得双亲的信息. 在进一步的研究^[34] 中,采用了一种多样性策略(每次变化后种群中 85% 的个体被随机产生的个体替代),实验表明引入多样性策略后算法获得了较好的改善.

Ramsey 等^[35] 采用了另外一种记忆策略,将 CBR 的思想引入 EA 中,使用一种知识库存储“成功”的个体,当种群重新初始化时将知识库中部分个体重新引入. Claus 等提出了一种动态的记忆模型,文献[36,37]表明这种记忆模型在温室控制和电话网最优路径选择问题中取得了较好的应用.

5 多种群的方法

基于记忆的方法中存在的最大问题就是所存储的信息,比如搜索到的峰的位置,会随着环境的变化而变得过时无用. 解决这类问题的一种方法是在搜索空间中几个可行区域上安排一些小的子种群,并且让这些子种群追随各自区域上峰的变化,从某种程度上说,这类多种群方法扮演的是一种自适应的记忆.

自组织侦察群^[38] 方法的基本思想是当一个峰被发现之后,种群就会分离出一个“子种群”来“监

视”这个峰,而种群中的其余个体(称为“主种群”)展开,继续搜索新的峰. Oppacher 等^[39]提出一种 Shifting Balance GA,整个种群分成核心种群和许多小的殖民地种群,核心种群的任务是探索极值点,小殖民地种群负责在适值曲线上几个孤立的区域进行搜索,即能够保证探索性.当适值曲线变化频繁而变化幅度较小时,这种算法明显优于常规 GA.文献[40]还讨论了一种更为有效的计算个体与核心种群之间距离的方法. Ursem^[41]提出一种多种族遗传算法,在该算法中,子种群的分组利用一种“峰谷探测过程”来确定.

文献[42]介绍的 Hierarchical Distributed Evolutionary Algorithm 也是一种多种群方法,在 HDEA 中包含宏观和微观两个进化过程.前者实际上是各个子种群对整个空间的探索过程,通过一个共享适值函数来保证整个种群的多样性;后者体现在个体对其所在的局部区域进行有效开发.通过对3种不同类型的动态函数的测试,能够证明该算法在动态环境中表现出良好的性能.

6 结 语

动态优化问题正逐渐成为进化算法的一个新的研究领域,本文对各种适用于动态环境的进化算法进行了分类综述.与传统算子相比,各种动态 EA 算子更注重保持种群的多样性,这是因为种群多样性是算法能够适应环境变化的必要条件.虽然越来越多的专家学者开始关注动态进化算法的研究,但目前这方面工作还处于初始阶段,许多问题值得研究:1)目前文献中关于各种动态进化算法的应用研究大多局限于一些简单的测试函数,而实际应用方面的研究较少.因此,把各种动态 EA 算子应用到复杂的现实问题中会成为研究热点之一.2)关于算法比较的研究还较少,比较各种 EA 算子的性能及其各自的优势也将成为引人关注的问题.3)关于动态进化计算方法的相关理论分析,如收敛性分析、计算复杂性分析、构造合适的动态测试函数、算法动态性能评价指标等问题也将成为值得研究的方向.

参考文献(References)

- [1] Fogel L J, Owens A J, Walsh M J. Artificial intelligence through simulated evolution [M]. New York: John Wiley, 1966.
- [2] Goldberg D E, Smith R E. Nonstationary function optimization using genetic algorithms with dominance and diploidy[C]. Proc of the 2nd Int Conf on Genetic Algorithms. Lawrence Erlbaum Associates, 1987: 59-68.
- [3] 曹宏庆,康立山,陈毓屏.动态系统的演化建模[J].计算机研究与发展,1999,36(8):923-931.
- (Cao H Q, Kang L S, Chen Y P. A hybrid evolutionary modeling algorithm for dynamic systems [J]. J of Computer Research and Development, 1999, 36(8): 923-931.)
- [4] Cartwright H M, Tuson A L. Genetic algorithms and flowshop scheduling: Towards the development of a real-time process control system[C]. Proc of the AISB Workshop on Evolutionary Computing. San Francisco: Morgan Kaufmann Publishers, 1994: 277-290.
- [5] Reeves C, Karatza H. Dynamic sequencing of a multi-processor system: A genetic algorithm approach [C]. Proc of 1st Int Conf on Artificial Neural Nets and Genetic Algorithms. San Francisco: Morgan Kaufmann Publishers, 1993: 491-495.
- [6] Bierwirth C, Kopfer H, Mattfeld D C, et al. Genetic algorithm based scheduling in a dynamic manufacturing environment [C]. Proc of IEEE Conf on Evolutionary Computation. Piscataway: IEEE Service Center, 1995: 439-443.
- [7] Bierwirth C, Mattfeld D C. Production scheduling and rescheduling with genetic Algorithms[J]. Evolutionary Computation, 1999, 7(1): 1-18.
- [8] Lin S C, Goodman E D, Punch W F. A genetic algorithm approach to dynamic job shop scheduling problems [C]. Proc of the 7th Int Conf on Genetic Algorithm. San Francisco: Morgan Kaufmann Publishers, 1997: 481-488.
- [9] Pico C A G, Wainwright R L. Dynamic scheduling of computer tasks using genetic algorithms[C]. Proc of the 1st IEEE Conf on Evolutionary Computation. Piscataway: IEEE Service Center, 1994: 829-833.
- [10] Krishnakumar K. Micro-genetic algorithms for stationary and non-stationary function optimization [C]. Intelligent Control and Adaptive Systems Proc of the SPIE. Philadelphia, 1989: 289-296.
- [11] Cobb H G. An investigation into the use of hypermutation as an adaptive operator in genetic algorithms having continuous, time-dependent nonstationary environment [R]. Washington: Naval Research Laboratory, 1990.
- [12] Morrison R W, Jong de K A. Triggered hypermutation revisited [C]. Proc of Congress on Evolutionary Computation. Piscataway: IEEE Service Center, 2000: 1025-1032.
- [13] Vavak F, Fogarty T C, Jukes K. Adaptive combustion balancing in multiple burner boiler using a genetic algorithm with variable range of local search [C]. Proc of the 7th Int Conf on Genetic Algorithm. San Francisco: Morgan Kaufmann Publishers, 1997: 719-726.

- [14] Gerratt S M, Walker J H. Genetic algorithms: Combining evolutionary and 'non'-evolutionary methods in tracking dynamic global optima[C]. Proc of the Genetic and Evolutionary Computation Conf. San Francisco: Morgan Kaufmann Publishers, 2002: 359-366.
- [15] Simões A, Costa E. Using GAs to deal with dynamic environments: A comparative study of several approaches based on promoting diversity[C]. Proc of the Genetic and Evolutionary Computation Conf. San Francisco: Morgan Kaufmann Publishers, 2002: 698.
- [16] Tin'os R, Carvalho A. A genetic algorithm with gene dependent mutation probability for non-stationary optimization problems[C]. Proc of 2004 Congress on Evolutionary Computing. Piscataway: IEEE Service Center, 2004: 1278-1285.
- [17] Grefenstette J J. Genetic algorithms for changing environments [C]. Parallel Problem Solving from Nature. Brussels, 1992: 137-144.
- [18] Cobb H G, Grefenstette J J. Genetic algorithms for tracking changing environments [C]. Proc of 5th Int on Genetic Algorithms. San Francisco: Morgan Kaufmann Publishers, 1993: 523-530.
- [19] Eriksson R, Olsson B. On the performance of evolutionary algorithms with life-time adaptation in dynamic fitness landscapes[C]. Proc of 2004 Congress on Evolutionary Computing. Piscataway: IEEE Service Center, 2004: 1293-1300.
- [20] Andersen H C. An investigation into genetic algorithms, and the relationship between speciation and the tracking of optima in dynamic functions [D]. Brisbane: Queensland University of Technology, 1991.
- [21] Hocaoglu C, Sanderson A C. Planning multi-paths using speciation in genetic algorithms[C]. Proc of the 3rd IEEE Int Conf on Evolutionary Computation. Piscataway: IEEE Service Center, 1996: 378-383.
- [22] Mori N, Kita H, Nishikawa Y. Adaptation to a changing environment by means of the thermodynamical genetic algorithm [C]. Parallel Problem Solving from Nature. Berlin: Springer Publishers, 1996: 513-522.
- [23] Collard P, Escazut C, Gaspar A. An evolutionary approach for time dependant optimization[J]. Int J on Artificial Intelligence Tols, 1997, 6(4): 665-695.
- [24] Yang S. The primal-dual genetic algorithm[C]. Proc of the 3rd Int Conf on Hybrid Intelligent System. IOS Press, 2003.
- [25] Yang S. Non-stationary problem optimization using the primal-dual genetic algorithm [C]. Proc of the 2003 Congress on Evolutionary Computation. Piscataway: IEEE Service Center, 2003: 2246-2253.
- [26] Ng K P, Wong K C. A new diploid scheme and dominance change mechanism for non-stationary function optimization [C]. Proc of 6th Int Conf on Genetic Algorithms. San Francisco: Morgan Kaufmann Publishers, 1995: 159-166.
- [27] Uyar A, Harmanci A. Preserving diversity in changing environments through diploidy with adaptive dominance [C]. Proc of the Genetic and Evolutionary Computation Conf. San Francisco: Morgan Kaufmann Publishers, 2002.
- [28] Hadad B S, Eick C F. Supporting polyploidy in genetic algorithms using dominance vectors[C]. Proc of the 6th Int Conf on Evolutionary Programming. San Francisco: Morgan Kaufmann Publishers, 1997: 223-234.
- [29] Ryan C. Diploidy without dominance[C]. Proc of 3rd Nordic Workshop on Genetic Algorithms. 1997: 63-70.
- [30] Ryan C, Collins J J. Polygenic inheritance — A haploid scheme that can outperform diploidy[C]. Proc of the 5th Int Conf on Paraller Problem Solving from. Berlin: Springer Publisher, 1997: 178-187.
- [31] Louis S J, Xu Z. Genetic algorithms for open shop scheduling and re-scheduling[C]. ISCA 11th Int Conf on Computers and Their Applications. Piscataway: IEEE Service Center, 1996: 99-102.
- [32] Branke J. Memory enhanced evolutionary algorithms for changing optimization problems [C]. Proc of the 1999 Congress on Evolutionary Computation. Piscataway: IEEE Service Center, 1999: 1875-1882.
- [33] Trojanowski K, Michalewicz Z, Xiao J. Adding memory to the evolutionary planner/ navigator [C]. Proc of the 1997 Congress on Evolutionary Computaton. Piscataway: IEEE Service Center, 1997: 483-487.
- [34] Trojanowski K, Michalewicz Z. Searching for optima in non-stationary environments [C]. Proc of the 1999 Congress on Evolutionary Computation. Piscataway: IEEE Service Center, 1999:1843-1850.
- [35] Ramsey C L, Grefenstette J J. Case-based initialization of genetic algorithms [C]. Proc of 5th Int Conf on Genetic Algorithms. San Francisco: Morgan Kaufmann Publishers, 1993:89-91.
- [36] Bendtsen C N, Krink T. Dynamic memory model for non-stationary optimization [C]. Proc of the 2002 Congress on Evolutionary Computatoin. Piscataway: IEEE Service Center, 2002: 145-150.

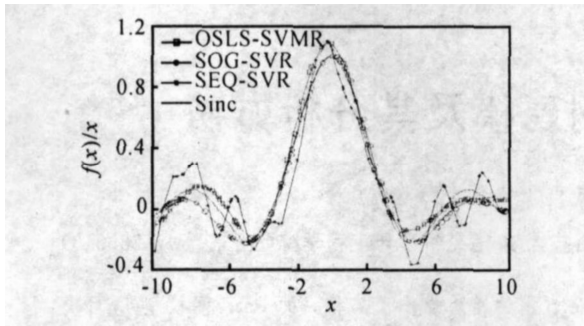


图 7 噪声方差为 0.5 回归的实际效果图

OS-LSSVMR 相对 SOG-SVR 效果较好.

6 结 语

最小二乘支持向量机回归将传统支持向量机回归的不等式约束变成等式约束,使传统支持向量机的二次规划优化方法变成了解线性方程的优化方法,然而这种方法不具有稀疏性.在线稀疏最小二乘支持向量机回归方法采用样本字典的方法大大减少了支持向量数目,使最小二乘支持向量机具有较好的稀疏性.与 SOG-SVR 相比,在线稀疏最小二乘支持向量机回归在收敛速度上具有稍好的效果,但需调整的参数较少,使用起来相对简单,因而可进一步应用于在线实时建模与控制等方面的研究.

参考文献(References)

[1] Syed N A, Liu H, Sung K K. Incremental learning with support vector machines [C]. Proc of Workshop on Support Vector Machines at the Int Joint Conf on Artificial Intelligence. Sweden, 1999: 313-321.
 [2] Cauwenberghs G, Poggio T. Incremental and decremental support vector machines learning [C]. Advances in Neural Information Systems. Cambridge: MIT Press, 2001: 409-415.

[3] Ma Junshui, Theiler James, Perkins Simon. Accurate on-line support vector regression [J]. Neural Computation, 2003, 15: 2683-2703.
 [4] Martin Mario. On-line support vector machine regression [C]. European Conf on Machine Learning. Berlin: Springer-Verlag, 2002: 282-294.
 [5] Frieß T T, Cristianini N, Campbell C. The kernel adatron algorithm: A fast and simple learning procedure for support vector machines [C]. Proc of 15th Int Conf Machine Learning. Morgan: Kaufmann Publishers, 1998: 188-196.
 [6] Vijayakumar S, Wu S. Sequential support vector classifiers and regression [C]. Proc Int Conf on Soft Computing. Genoa, 1999: 610-619.
 [7] Jyrki Kivinen, Alexander J Smola, Robert C Williamson. Online learning with kernels [J]. IEEE Trans on Signal Processing, 2004, 52(8): 2165-2176.
 [8] Suykens J A K, Lukas L, Vandewalle J. Sparse approximation using least squares support vector machines [C]. IEEE Int Symposium on Circuits and Systems. Geneva, 2000: 11757-11760.
 [9] Yaakov E, Shie M, Ron M. Sparse online greedy support vector regression [C]. Proc of European Conf on Machine Learning. Berlin: Springer-Verlag, 2002: 84-96.
 [10] Down T, Gates K E, Masters A. Exact simplification of support vector solutions [J]. J of Machine Learning Research, 2001, 2: 293-297.
 [11] 黄琳. 系统与控制理论中的线性代数 [M]. 北京: 科学出版社, 1984.
 (Huang L. Linear algebra of system and control theory [M]. Beijing: Science Press, 1984.)

(上接第 131 页)

[37] Bendtsen C N, Krink T. Phone routing using the dynamic memory model [C]. Proc of the 2002 Congress on Evolutionary Computation. Piscataway: IEEE Service Center, 2002: 992-997.
 [38] Branke J, Kaubler T, Schmidt C, et al. A multi-population approach to dynamic optimization problems [C]. Adaptive Computing in Design and Manufacturing 2000. Berlin: Springer-Verlag, 2000: 299-308.
 [39] Oppacher F, Wineberg M. The shifting balance genetic algorithm: Improving the GA in a dynamic environment [C]. Proc of Genetic and Evolutionary Computation Conf. San Francisco: Morgan Kaufmann Publisher, 1999: 504-510.
 [40] Wineberg M, Oppacher F. Enhancing the GA's ability to cope with dynamic environments [C]. Proc of Genetic and Evolutionary Computation Conf. San Francisco: Morgan Kaufmann Publisher, 2000: 3-10.
 [41] Ursem R K. Multinational GA optimization technique in dynamic environments [C]. Proc of Genetic and Evolutionary Computation Conf. San Francisco: Morgan Kaufmann Publisher, 2000: 19-26.
 [42] Oh S K, Lee C Y, Lee J J. A new distributed evolutionary algorithm for optimization in nonstationary environments [C]. Proc of the 2002 Congress on Evolutionary Computation. Piscataway: IEEE Service Center, 2002: 1875-1882.