

蚁群算法基本原理及其应用综述

程艳燕

(南京信息工程大学 江苏 南京 210044)

摘要:蚁群算法(ACA)是一种广泛应用于优化领域的仿生进化算法。从ACA发展背景着手,分析比较国内外ACA研究团队与发展情况,立足于基本原理,分析其数学模型,介绍了六种经典的改进模型,对其优缺点进行分析,简要总结其应用领域并对其今后的发展、应用做出展望。

关键词:蚁群算法;优化;改进;应用

中图分类号:TP273

文献标识码:A

0 引言

专家发现单个蚂蚁只具有一些简单的行为能力,但整个蚁群却能完成一系列复杂的任务,这种现象是通过高度组织协调完成的。1991年,意大利学者M.Dorigo首次提出一种新型仿生算法ACA,研究了蚂蚁的行为,提出其基本原理及数学模型,并将之应用于寻求旅行商问题(TSP)的解。

通过实验及相关理论证明,ACA有着有着优化的选择机制的本质,而这种适应和协作机制使之具有良好的发现能力及其它算法所没有的优点,如较强的鲁棒性、分布式计算、易与其他方法结合等;但同时也不应忽略其不足,如搜索时间较长,若每步进行信息素更新,计算仿真时所占用CPU时间过长;若当前最优路径不是全局最优路径,但其信息素浓度过高时,靠公式对信息素浓度的调整不能缓解这种现象,会陷入局部收敛,无法寻找到全局最优解;转移概率过大时,虽有较快的收敛速度,但会导致早熟收敛,所以正反馈原理所引起的自催化现象意在强化性能好的解,却容易出现停滞现象。笔者综述性地介绍了ACA,对一些已有的提出自己的想法,并对其应用及发展前景提出了展望。

1 蚁群算法概述

ACA源自于蚁群的觅食行为。S. Goss的“双桥”实验说明蚂蚁总会选择

距食物源较短的分支。蚂蚁之间通过信息素进行信息的传递,捷径上的信息素越多,吸引的蚂蚁越多,形成正反馈机制,达到一种协调化的高组织状态,该行为称集体自催化。目前研究的多为大规模征兵,即仅靠化学追踪的征兵。

1.1 蚁群算法的基本原理

为便于研究提出以下基本假设:蚂蚁间通过信息素和环境进行间接通信;蚂蚁对环境的反应由其内部模决定;蚂蚁个体是独立的,但群体却呈现出一种随机性。

蚂蚁通过适应和协作两个阶段的调整从无序到有序,得到最优解,完成对路径的搜索。对路径的选择,重点在转移概率,即某时刻蚂蚁 k 在城市 i 选择城市 j 的概率的大小

$$p_{ij}^k(t) = \begin{cases} \arg \max_{u \in allowed_k} \{[\tau(r, u)^\alpha] \cdot [\eta(r, u)^\beta]\} (q \leq q_0) \\ \frac{\tau_{ij}^\alpha(t) \eta_{ij}^\beta(t)}{\sum_{s \in allowed_k} \tau_{is}^\alpha(t) \eta_{is}^\beta(t)} (j \in allowed_k) \end{cases} \quad (1)$$

其中, q_0 和 q 分别为 $[0, 1]$ 上的参数和均匀分布的随机数,其大小决定了利用先验知识与探索新路径之间的相对重要性。若 $q \leq q_0$,则转移概率选取上面一个式子,即按照先验知识选择最好的边,否则,按照转移概率选择一条边,式1又被称为伪随机比例规则; $j \in allowed_k$ 为蚂蚁 k 下一步允许选择的城市; $\eta_{is}(t)$ 为能见度因子; u 为禁忌表; α 、 β 分别反映蚂蚁在运动过程中所积累的

信息和启发信息在蚂蚁选择路径中的相对重要性; $\tau(\cdot)$ 为信息素浓度的函数。根据不同的模型,信息素做不同的调整,如全局更新规则和局部更新规则。

2 蚁群算法的发展

蚁群优化(ACO)是研究受到真实蚁群行为启发的智能系统,常用于解决离散优化问题。启发式ACO是1991年由Dorigo和Gambardella提出定义的。

2.1 国外蚁群算法的发展概况

2.1.1 有关蚁群算法的研究团队

从ACO提出至今,越来越多的专家投身于蚁群算法的研究之中,其中较为突出的有以下四个:

(1)瑞士卢加诺IDSIA。1988年建立的IDSIA是非营利性研究人工智能研究所,2000年成为公共研究机构,隶属于卢加诺大学的信息学院和瑞士意大利语区高等专业学院的科技创新系,主要负责人为Luca, Lepori, Carlo和Schmidhuber。其中一研究主题是人工蚂蚁,该多代理方法是受基于信息素交流的生物蚂蚁启发而来,由前高级研究员Dorigo和联合负责人Luca领导研究的。其人工蚁和局部搜索算法的结合已经成为解决某些图形优化任务的方法选择,如车辆路径和网络路径,其迅速发展促成许多商业应用和关于人工蚂蚁的专门会议。

(2)比利时布鲁塞尔IRIDIA。IRIDIA是布鲁塞尔自由大学人工智能研究实

验室,主要在理论上进行深入研究以及计算智能应用于优化。在 Dorigo 和 Bersini 领导下,其主要研究领域为群智能、组合问题的求解和连续空间优化问题的启发式、生物网络原理性研究以及商业智能应用四个方面,而有关于 ACA 的方向为群智能和元启发式。

IRIDIA 在群智能的 ACO 和群机器人这两方面处于世界领先地位,对于具体元启发式的研究聚焦于 ACO,主要研究点是研发一套合理的实验方法论,一套经验学习和元启发式构建的发展工具的应用,特别着重研究的是发展能够设计和完善随机局部搜索算法和元启发式算法的方法论。

近期研究 ACA 项目有:对 ACA 的基础理论研究、复杂系统的智能计算方法、使用生物启发和软件计算的医学成像、自组织的蚁群、走向型机器人和群智能系统的通信策略。

(3)奥地利维也纳大学经济与统计商学院。由 Richard 和 Artner 等组成的团队,主要研究遗传算法、项目管理、最优控制、ACO 和电子装配等研究课题。其中,关于 ACO 方面的工作由 Richard 和 Karl 负责,从 1999 年开始至今。

(4)德国莱比锡大学并行计算与复杂系统。Martin 领导的数学与计算机科学学院计算机科学研究所的并行计算与复杂系统团队,关于 ACA 的主要研究是由人类前沿科学组织自主计划的自然系统优化,以及东风集团项目中的一些关于系统、模型和硬件算法等。

2.1.2 有关蚁群算法的国际会议

随着人们对 ACA 越来越重视,相关会议也组织起来,来自世界各地的专家对 ACA 及其应用展开研究讨论,其中以 Dorigo 为大会总主席的 ANTS 最为权威。

1998 年在比利时布鲁塞尔召开第一届 ACA 研讨会:从蚁群到人工蚁,每隔两年召开一次蚁群优化和群智能国际会议。期间,2000 年召开第二届 ACA 国际专会:从蚁群到人工蚁。另外,自 2005 年在美国加利福尼亚州帕萨迪纳威斯汀召开了 IEEE 群智能讨论会,2006 年、2007 年分别在美国印第安州印第安纳波利斯和美国夏威夷檀香山希尔顿村延续召开此会。除以上较为权威的会议,还有很多相关的国际会议,

说明 ACA 在国际范围内得到越来越多的重视,研究亦广泛展开。

2.2 国内蚁群算法的发展概况

国内对该算法研究最早的是张纪会博士和徐心和教授。1999 年 3 月,他们首次简单引进 ACA,从其基本原理、模型、伪码流程进行说明,对 Oliver30 TSP 问题分析说明,但未对基本模型中的参数进行详细地理论说明,且停止条件的界定较含糊,主要靠经验决定。其后引入的变异机制加快了收敛速度,使得到较优解的周期缩短,并从计算机仿真层面上证明其有效性。

2001 年,陈焯引入遗传算法中用到的杂交算子来改善蚁群搜索速度慢、容易陷入局部最优。但随路程的增长,每次的代数显著增加,计算量较大。同年 8 月,郝晋等通过将转移概率设置为转移系数,结合扰动策略以防止漏选最优路径,能够节约计算时间,且能够很好的克服算法容易陷入局部最优的缺陷,计算精度也有提高。但在关于倒指数的扰动因子和均匀分布的随机数的选择并未解决,仍以实验为主要获取手段。而后李艳君等对自适应 ACA 进行了进一步研究,对信息素采取自适应更新,应用于连续空间优化问题的求解,并进行了证明。

2002 年,王颖等对自适应 ACA 作出了改进,获得了很好的结果。该 ACA 在进化代数减少的情况下,解的质量也得到了提高,在一定程度上实现了收敛速度与解的质量的平衡。但分析其复杂度可知,蚂蚁的数目与问题规模不相上下,蚂蚁数目增多会使收敛速度过快,为防止该现象而将信息素挥发系数设置得比一般情况低一个数量级,而相关系数 α 、 β 、 ρ 等则由实验设定。当蚂蚁数量与问题规模相当时,实验次数与时间是不容忽视的一个问题。ACA 除在原理层面进行模型改进,在应用方面也有一定发展,如张宗永等将 ACA 作出改进,配合随机分布技术,应用到分析上海市整个内河航道和集装箱运输的过程中,对内河航道进行规划,最终得出合理的分配方案并提出了满足最优分配的河道改造的建议。

2003 年,陈峻等令人工蚁模仿真实蚂蚁的感觉和知觉行为,设置合理绝对感觉阈值以克服蚂蚁在初始选择时容

易失去解的多样性,改进选择策略以自适应的修改路径上的信息量,通过对不同规模和不对称 TSP 的仿真说明算法具有较好的收敛性和稳定性。新提出的启发式搜索方法——智能 ACA,通过取消外激素、自动调整选择最优路径的比例,改变了选择目标城市的依据和引入扰动,仿真结果说明在减少计算量的同时,可取得更好的搜索结果,但也指出通过实验确定相关的物理系数不利于算法的推广。但该文仅针对 TSP,对其他问题能否应用仍不明确。

2004 年,黄国锐等提出采用不同于传统基本 ACA 所采用的蚁周模型,它采用了更贴近于真实蚂蚁行为的蚁量模型,建立信息素扩散模型,使相距较近的蚂蚁之间能够更好地进行协作,文中仿真结果表明该算法的有效性。然而文中虽在达到收敛所需进化代数上较基本 ACA 有了很大的改进,减少约 4 倍,但最短路径长度的减少并不明显,且参数的设定仍是以试验为手段,缺乏理论支撑。

针对基本 ACA 容易陷入局部最优、收敛慢等缺陷,许多新模型陆续提出,如基于云模型的 ACA、对信息素的限制、回归型的等,甚至还有不少研究者试图从新的角度重新审视并尝试性研究 ACA,较为新颖的有从蚁群社会的“多态性”出发,试图以更贴近真实世界中蚂蚁行为来研究 ACA,发现更适应较大规模的问题,以及将着眼于蚁群整体的研究角度转换到关注蚂蚁个体的速度对算法的影响。为从根本上解决 ACA 不足,其收敛性的分析也在不断展开,如用动态分阶段的方式,而具体影响 ACA 的参数也越来越得到关注,如有相关讨论,但参数如何设定并未有理论依据,如何建立通用标准来对参数进行最优设置仍是难点。在研究的应用范围方面也从一开始的离散域扩展到连续域,连续域中有关于收敛性的研究,以及新模型的设计也在进行着。

全国各高校及研究机构也对该算法展开了研究,如徐锋、杜军平的《改进蚁群算法在旅游路线规划中的应用研究》等。1999 年从首次引入 ACA 至今,相关研究蓬勃发展,右图为相关的论文题名和关键字的论文数量增长表。

国内对于 ACA 的研究也越来越深

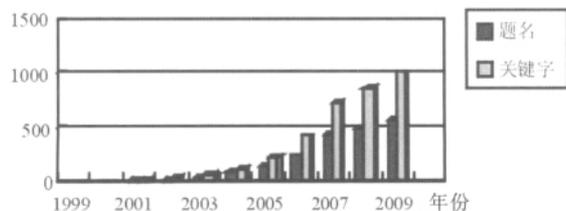


图1 研究 ACA 论文数增长表

入,基于各类模型的 ACA 层出不穷,研究域也从离散域扩展到连续域,同时 ACA 在不断与其他算法结合以克服本身的缺陷。但国内的研究起步较晚,对于影响收敛性的 α 、 β 等相关参数至今无法确定一套相关的理论来进行设置,只能通过反复试验来大致确定一个参数范围,且研究较多地停留在理论仿真,应用到实践中仍较少,而国外对于这些范围的研究已经较为成熟。

2.3 蚁群算法的改进型

ACA 的收敛速度和最优解的全局性是一对矛盾体,收敛速度过快,会导致早熟,陷入局部最优解,而当信息素更新不及时或算法计算量过大时,则导致收敛速度过慢而应用不现实,为克服这些问题,相应的改进的 ACA 不断提出。

(1)带精英策略的蚁群系统(AS)。在带精英策略的 AS 中,为了使到目前为止所找出的最优解在下一个循环中对蚂蚁更有吸引力,在每次循环之后给予最优解额外的信息素量。它信息素的更新为:

$$\tau_{ij}(t+1) = \rho\tau_{ij}(t) + \Delta\tau_{ij} + \Delta\tau_{ij}^* \quad (2)$$

其中, $\Delta\tau_{ij}^*$ 表示经蚂蚁引起路径 (i, j) 上的信息素量的增加; σ 是精英蚂蚁的个数; L^* 为所找出的最优解的路径长度。这种算法收敛速度快,而且计算耗时短,但如果 σ 过大,搜索会局限于极值周围,导致搜索的早熟收敛。

(2)基于优化排序的蚂蚁系统。基于优化排序的 AS 针对带精英策略的 AS 的缺点,通过排序很好地抑制了早熟,尤其当初始状态各解的差异性不大时,效果显著,但其中,第 μ 只最优蚂蚁引起的路径 (i, j) 上信息素量的增加

$$\Delta\tau_{ij} = \sum_{k=1}^{\sigma} \Delta\tau_{ij}^{\mu} \quad (3)$$

$\Delta\tau_{ij}^{\mu}$ 为第 μ 只最优蚂蚁引起的路径 (i, j) 上信息素量的增加; $\Delta\tau_{ij}^*$ 由精英蚂蚁引起的路径 (i, j) 上的信息素量的增加。

(3)最大—最小蚁群系统(MMAS)。MMAS 是由德国学者 T. Stuetzle 和 H. Hoos 提出的,其目的主要在于防止过早的算法停滞现象,性能改进在以下几方面:

- ①只允许其中的一个路径更新信息素。该路径通常是最优路径,可以是所有或当前周游中的最优路径。即只有一只蚂蚁进行信息素更新,而用于信息素更新的蚂蚁可能是迭代最优蚂蚁,也可能是全局最优蚂蚁;
- ②为了避免搜索的停滞,在每个解的元素上的信息素轨迹量的值被限制在一个固定范围内 $[\tau_{min}, \tau_{max}]$;
- ③为了尽可能的扩大搜索范围,信息素轨迹初始值设为 τ_{max} 。

该法的轨迹更新规则为

$$\tau_{ij}(t+1) = \rho\tau_{ij} + \Delta\tau_{ij}^{best} \quad (4)$$

其中, $\Delta\tau_{ij}^{best} = 1/f(s^{best})$, $f(s^{best})$ 表示迭代最优解或全局最优解的值。这也保证了在初始的搜索中,能够保证最优解的多样性。

尽管在算法中对信息素浓度进行了最大和最小值的限制,但此限制还不足以在较长的运行时间里持久消除停滞现象,故采用信息素的平滑机制:按照比例更新的方法来调整信息素的浓度,使信息素浓度的增加值正比于 τ_{max} 与边 (i, j) 上的信息素浓度的差值。基本做法是:限定信息素浓度允许值的上下限,且采用了平滑机制,也就是不同点中的③。

在当前路径上的信息素浓度远远高于其它路径时,会导致过度收敛,为了降低运行时间,可增加候选解集合的选择概率,不仅能降低算法的运行时间,还可提高该算法的性能。这些改进模式可以通过对一些较大规模的巡游问题求解加以证明。

在全局修正规则中,只让实现最好周游的蚂蚁释放信息素。它和改进的状态转移规则相结合的搜索模式,保证了蚂蚁在优秀父辈完成的周游领域内进行更多搜索,这就使得相应的求解速度大大提高。

(4)最优—最差蚂蚁系统。鉴于 AS 搜索效率低和质量差的缺点,最优—最差蚂蚁系统增强了搜索过程的指导性,使蚂蚁的搜索更集中于到当前循环位置所找出的最好路径的领域内,既避免

的早熟收敛,又缩短的搜索时间,其模型为

$$\tau(r, s) = (1 - \rho) \cdot \tau(r, s) - \varepsilon \cdot \frac{L_{worst}}{L_{best}} \quad (5)$$

其中, ε 为该算法引入的一个参数; L_{worst} 、 L_{best} 分别表示当前循环中最差和最优蚂蚁的路径长度; $\tau(r, s)$ 表示城市 r 和城市 s 之间的信息素轨迹量。

(5)随机扰动蚁群算法。ACA 的参数往往是通过反复试验,由经验凑得的,但对算法的计算效率和收敛性会产生不利影响。ACA 的主要依据是信息正反馈原理和某种启发式算法的结合,式 1 充分说明了这点。如果放到某条路径上信息素越多且路径越短,则该路径被选中的概率越大,此时转移概率基本不变,成为了“转移系数”,由于蚂蚁总是选择转移系数最大的路径,这个值就具有一定的确定性,而此时,随机扰动蚁群算法与基本蚁群算法相同,也就不可避免会出现停滞现象,由此提出了可变的扰动因子。

随机扰动蚁群算法的过程具体分为“粗调”和“细调”两部分。“粗调”是事先设定信息素浓度和转移概率,再调整系数 α 和 Q ,而 α 和 Q 的大小分别决定了某一路径的信息量对蚂蚁选择路径的影响程度和路径上信息量的更新程度,而“细调”则相反。确定性选择导致蚂蚁总是选择转移系数最大的路径,而随机选择导致计算转移系数时具有较强的随机性,正是两者的共同作用才使算法具有更强的全局搜索能力。

(6)动态蚁群算法。动态蚁群算法相对于传统蚁群算法和最大—最小蚁群算法,其主要的改进在于:前者在迭代过程中,在选择目标城市的标准时,没有使用一个固定的标准,这样就有利于减少进化停滞现象。后者两者依据信息素和启发函数来选择目标城市,如何依据这两个因子选择城市对算法的性能是非常关键的。蚁群算法对信息素的强度没有限制,因而容易陷入局部最优解,MMAS 对信息素的强度给予了一定的范围限制,大大改善了算法的性能,而动态蚁群算法的挥发因子是动态变化。其信息素浓度越高,则挥发因子越大;浓度越低,则挥发因子越小。这样实际上就对信息素的浓度也进行了限制,使信息素浓度不可能无限增大,也不可

能为零。

(7)带奖惩策略的蚁群算法。带奖惩策略的蚁群算法采取奖励每次循环中的较优解而惩罚一般性的解,限制信息素量的范围,将挥发系数设为相对较大值以抵消信息素两极分化影响路径选择的随机性,同时兼顾了提高收敛速度与避免早熟收敛。通过模拟实验,该方法读对解决复杂组合优化问题有很好的效果。但是对于奖励的蚂蚁数目和信息素挥发系数的取值只能通过实验来确定,当遇到较大规模的问题时,计算量将会非常大,不易于实践。

此外,自适应 ACA 也是应用较为广泛的一种蚁群算法,一文中的实验表明由于采用自适应选择和动态调整策略,算法的性能明显得到改善,该方法不仅能够加快收敛速度,节省搜索时间,而且能够克服停滞行为的过早出现,有利于发现更好的解。这对于求解大规模优化问题是十分有利的。笔者对自适应蚁群算法进行了进一步研究,对信息素采取自适应更新,应用于求解连续空间优化问题的,并进行了证明。

除了上述改进型,还有其他如具有感觉和知觉特征的 ACA、基于去交叉局部优化策略的 ACA、基于信息素扩散的 ACA、基于模式学习的小窗口 ACA、带聚类处理的 ACA、基于云模型理论的 ACA、基于信息熵的改进 ACA 等。据需要可以针对 ACA 模型中的不同参数进行调整,甚至创新。

3 蚁群算法的应用与展望

3.1 蚁群算法的应用概况

随着蚁群算法研究的深入,其应用也愈来愈广泛,从宏观上看,主要分为静态组合优化中的应用和动态组合优化中的应用,下面罗列了一些较为具体的应用。此外,国家经济分配问题的协作智能搜索方法、解决反应堆再生器的多目标优化问题等。以下是国内一些论文中出现的关于 ACA 的应用范围中的特例。

3.2 蚁群算法展望

近年研究 ACA 的研究人员和研究成果成几何级数增长,蚁群算法表现出了其强大的生命力和广阔的前景。

在模型改进方面,注意思维变换,比如尝试逆向思维,试图建立一个具有

表 1 蚁群算法的应用形式

优化问题求解	基于 ACA 的交通过程建模及规划问题求解	计算机领域	机器人设计及控制
组合优化问题求解领域、调度问题求解领域、规划问题求解领域、约束优化问题求解、连续空间优化问题求解、二次指派问题求解、着色问题求解	交通过程建模、交通过程优化及导航、车辆路线规划问题求解	专家系统问题求解(人工智能研究领域)、计算机图形学领域、数据挖掘和数据高层综合问题求解、计算机网络管理和移动计算	机器人设计、机器人控制及协调、移动式机器人导航
电力系统应用	通信领域	工程应用领域	
电力系统优化、电力系统负荷分配及调度、发电规划及调度问题求解、电力系统故障分析	路由选择及其负载、资源配置问题、移动计算、天线阵列控制、智能网络控制	制造过程控制及优化、工程设计问题求解、工程设计、其它工程应用领域(采矿、河道航行、设施布局、地下房屋建造顺序优化、水利分配系统)	

表 2 ACA 的应用范围举例

经典求解	TSP
任务调度与分配	柔性装配系统调度、多目标资源受限项目调度、配货发送、物流配送点选址(生鲜农产品配送中心选址)、JSP、智能公交调度、应急物资调度、网格任务调度
路径问题	物流配送路径问题(配送中心多目标选址问题)、迷宫路径规划问题、车辆路径问题、最短路径、动态路径、多 AUV 路径规划、自动仓库路径优化、常规医疗器械配送路径优化、复杂地形下的路径规划
系统构建与优化	传播电力系统故障重构研究、电力系统优化
网络分布与优化	输配气管网优化、铁路路网优化路径规划、BP 网络优化、集中网站的 WEB 搜索优化策略
其他	火灾人群疏散仿真、军事应用中的火力分配、战时雷达器材运输调度、分类问题中的 KNN 文本分类、资源优化中的传感器资源优化、具体分析中的求解蛋白质折叠问题、地图矢量化等

普遍适应性的 ACA 模型,而不仅仅是针对某一类特定的问题。对于从自然界真实蚁群的行为研究,进一步剖析深层原因,提出更加智能化的蚁群混合行为模型。此外,很重要的一点就是摆脱已有框架,提出更具有创新性的 ACA 模型,尤其是与智能融合方面。

在理论分析方面,由于 ACA 是一种概率搜索算法,从数学理论上对其正确性与可靠性进行严密的证明相对于确定性算法而言,是比较困难的。而 Gutjahr W J 首次利用有向图论对 ACA 进行其收敛性方面的证明。现阶段 ACA 的众多模型并没有归结于共同的框架之下,并且对于 ACA 在连续域的研究相对离散域而言还相对较少,所以对于收敛性的证明和理论分析值得进一步深入研究。

而对于蚁群方法的并行实现而言,ACA 本身隐含着一定的并行性,单只蚂蚁在一次循环中独立于其他蚂蚁。因此从本质上说,ACA 应以分布式的协同优化计算方式为特征,而在串行计算机上

对 ACA 的并行模拟并不能真正体现蚁群算法的本质特征,因此,进一步的研究工作还应该开展 ACA 的并行机实现。从目前的发展趋势来看,最优并行化将会成为发展重点。

关于硬件实现,ACA 能够处理的电路规模不及常规电路综合方法。算法的运行时间、硬件功能以及电路规模之间的协调平衡。同时,对蚁群算法的仿真硬件产品的可靠性、可测试性、普适性和鲁棒性问题也需要进行进一步的研究,尤其是实际应用时的稳定性。

然而最关键的突破点是如何从实际的现实问题中抽象出其理论模型,从而更加清晰明确地与 ACA 结合应用,在应用中,如何进行相关产品的批量生产以及市场应用的推广,特别是在技术规范以及国家标准方面未雨绸缪也是十分必要的。

4 结语

ACA 被广泛用于理论与实际应用的各个领域,从最基本的 ACA 初用于

解决旅行商问题,到分析解决当下的交通布线、货物运送、蛋白质折叠等问题。ACA 应用领域之广,效率之高空前,这和其本身所具有的一些特征是密不可分的。

ACA 提出之际,其创始人对它的发展指出了四个的主要的研究方向:所提出模型和一般自催化过程的数学理论公式;通过研究这种算法能够很好的用于解决问题的类型,该方法的一般性的评价;所提出的模型在人工智能领域的研究意义,尤其是在模式识别和机器学习领域;开发所提出模型的内在并行性,以此绘制并行体系结构的远景。

在以上方向的指导下,针对传统 ACA 收敛速度慢、易陷入局部最优及早熟收敛的缺陷和早期的一些算法模型、收敛性等问题所存在的不确定性,研究者们不断提出新的改进模型,新近提出的量子 ACA 大大加速了各自算法的收敛速度,使得并行计算和鲁棒性得到了强化,但是由于研究时间不长,缺乏坚实的理论基础,而且其可行性是建立在实验仿真的基础上。至今,蚁群算法仍然没有一个普适性的数学模型,尤其是在信息素更新方面,它的主要参数的手段也是建立在实验仿真基础上,从而导致理论基础相对薄弱。某种蚁群优化算法的延拓性在实践应用上对一般 ACA 并不适用,也就是缺乏普适性,而且 AS、MMAS 等的收敛性的结果并不能预测最优解能何时找到。

参考文献

- Alberto Coloni, Macro Dorigo etc. Distributed Optimization by Ant Colonies. Elsevier Publishing[M], 1991
- 陈峻.基于分布均匀度的自适应蚁群算法[J].软件学报,2003(14)
- 王剑.蚁群算法的理论与应用[J].机电工程,2003(20)
- 张纪会.自适应蚁群算法[J].控制理论与应用,2000(2)
- 李士勇.蚁群算法及其应用[M].哈尔滨:哈尔滨工业大学出版社,2006
- 张纪会.一种新的进化算法——蚁群算法[J].系统工程理论与实践,1999(3)
- 张纪会.具有变异特征的蚁群算法[J].计算机研究与发展,1999(10)
- 陈焯.带杂交算子的蚁群算法[J].计算机工程,2001(12)
- 郝晋.具有随机扰动特性的蚁群算法[J].仪器仪表学报,2001(8)
- 李艳君.连续空间优化问题的自适应蚁群系统算法[J].模式识别与人工智能,2001(12)
- 王颖.一种自适应蚁群算法及其仿真研究[J].系统仿真学报,2002(1)
- 张宗永.蚁群算法的改进及其应用[J].上海交通大学学报,2002(11)
- 陈峻.具有感觉和知觉特征的蚁群算法[J].系统仿真学报,2003(10)
- 曹浪财.智能蚁群算法——蚁群算法的改进[J].计算机应用研究,2003(10)
- 黄国锐.基于信息素扩散的蚁群算法[J].电子学报,2004(5)
- 段海滨.基于云模型理论的蚁群算法改进研究[J].哈尔滨工业大学学报,2005(1)
- 柯良军.有限级信息素蚁群算法[J].自动化学报,2006(3)
- 许殿.回归蚁群算法[J].西安电子科技大学学报(自然科学版),2005(12)
- 徐精明.多态蚁群算法[J].中国科学技术大学学报,2005(2)
- 印峰.个体速度差异的蚁群算法设计及仿真[J].智能系统学报,2009(12)
- 邢桂华.动态分阶段蚁群算法及其收敛性分析[J].控制与决策,2007(6)
- 彭沛夫.具有杂交、变异因子的自适应蚁群算法最优 PID 参数[J].计算机工程与应用,2006(6)
- 吴春明.蚁群算法中系统初始化及系统参数的研究[J].电子学报,2006(8)
- 李士勇.求解连续空间优化问题的量子蚁群算法[J].控制理论与应用,2008(4)
- 赵云涛.用于连续域优化的蚁群算法及其收敛性研究[J].系统仿真学报,2008(8)
- 李士勇.蚁群算法及其应用[M].哈尔滨:哈尔滨工业大学出版社,2006(9)
- 陈峻.基于分布均匀度的自适应蚁群算法[J].软件学报,2003(8)
- 吴启迪.智能蚁群算法及其应用[M].上海:上海科技教育出版社,2004(4)
- 张飞君.奖惩蚁群算法[J].计算机工程与应用,2010(10)
- 段海滨.蚁群算法原理及其应用[M].北京:科学出版社,2005(12)
- 赵俊生.一种改进的量子蚁群算法及其应用[J].计算机应用于软件,2010(7)
- Christian Blum, Ant Colony Optimization: Introduction and Recent Trends[J]. Physics of Life, 2005(10)
- IRIDIA. Ant Colony Optimization——Artificial Ants as A Computational Intelligence Technique[J]. IRIDIA—Technical Report Series, 2006(6)

(责任编辑 高平 杜倩倩)

The Basic Principle and Applications of ACA

Abstract: ACA is a bionics evolutionary arithmetic which is diffusely used by optimized area. This paper starts from the background of ACA, then analyzes its math model, introduces its six classical ameliorated models, construes its merit and defect, summarizes its application area and development.

Key words: aca; optimize; ameliorate; application