

王 丽 著

智能算法的 改进与案例分析

ZHINENG SUANFA DE
GAIJIN YU ANLI FENXI



北京邮电大学出版社
www.buptpress.com

智能算法的改进与案例分析

王 丽 著



北京邮电大学出版社

• 北京 •

内 容 简 介

本书是编者从事算法研究与应用的经验总结,每种算法的改进及应用案例来源于编者近几年的研究成果。

群智能优化算法是人工智能领域方兴未艾的研究方向,而支持向量机作为一种具有代表性的机器学习方法,一直是数据挖掘领域的研究热点。本书第2~5章分别介绍了粒子群算法、人工鱼群算法、蚁群算法和群集蜘蛛优化算法的基本理论、改进思想、程序实现和案例分析,使读者比较全面地了解四种典型群智能优化算法的相关理论知识、改进思想及其应用。第6章系统介绍了支持向量机的理论基础、建模流程和应用案例。

本书理论联系案例,对智能算法的原理、实现步骤、改进思想和应用等进行了全面且详细的介绍,可供高年级本科生作为开阔视野、增长知识的材料阅读,也可供自动化、人工智能、计算机科学、信息科学、系统工程、生物信息等相关领域的研究生、教师、科研人员以及工程技术人员参考使用。

图书在版编目(CIP)数据

智能算法的改进与案例分析 / 王丽著. -- 北京: 北京邮电大学出版社, 2019.1

ISBN 978-7-5635-5080-7

I. ①智… II. ①王… III. ①人工智能—算法—研究 IV. ①TP18

中国版本图书馆 CIP 数据核字(2017)第 079095 号

书 名 智能算法的改进与案例分析

著 者 王 丽

责任编辑 沙一飞

出版发行 北京邮电大学出版社

社 址 北京市海淀区西土城路 10 号(100876)

电话传真 010-82333010 62282185(发行部) 010-82333009 62283578(传真)

网 址 www.buptpress3.com

电子信箱 ctrd@buptpress.com

经 销 各地新华书店

印 刷 北京建宏印刷有限公司

开 本 787 mm×960 mm 1/16

印 张 10.75

字 数 208 千字

版 次 2019 年 1 月第 1 版 2019 年 1 月第 1 次印刷

ISBN 978-7-5635-5080-7

定价: 54.00 元

如有质量问题请与发行部联系

版权所有 侵权必究

前 言

随着计算机技术的快速发展,为了在一定程度上解决大空间、非线性、全局寻优、组合优化等复杂问题,人工智能领域中许多智能优化方法不断涌现。群智能算法的研究是生物学、数学、计算机、管理学等多个学科的交叉。群智能算法通过生物学来观察动物行为,再用数学方法建立抽象模型,然后用计算机编程来模拟算法的结果,最后通过管理学的思想来制定、改善、协调群体行为。群智能算法因简单性、分布式、鲁棒性、易扩展性和广泛的适用性等优点,已经在多个领域获得了广泛的应用,例如组合优化、信号处理、图像处理、生产调度、任务分配、模式识别、自动控制和机械设计等。但是,没有一种万能的算法可以解决所有的问题或在各种问题中均表现出色。所以,近几年群智能算法层出不穷,针对某一种算法的改进和参数设置更是多种多样。本书第2~5章分别介绍了4种具有代表性的群智能优化算法(粒子群算法、人工鱼群算法、蚁群算法以及群集蜘蛛算法)的理论基础、原理、算法流程和改进方法,并给出了具体的MATLAB仿真案例。通过探索不同的算法模型和改进策略在求解优化问题中的优劣,逐渐丰富群智能算法的求解模式,对于今后采取新的模式解决现实问题甚至反过来理解现实世界的运行规律都具有很大的意义。同时,随着传感器技术、存储技术和网络技术的迅猛发展,数据挖掘成为大数据时代人工智能领域的另一个研究热点。作为数据挖掘中的一个重要分支——支持向量机,不仅可以应用在分类、预测、模式识别中,也可以应用于回归拟合问题中。本书第6章介绍了支持向量机的分类机和回归机的理论基础和建模流程,并以两种网络可靠性的预测作为案例与其他智能算法预测结果对比,从而体现出支持向量机的良好性能。

由于编者水平有限,书中对智能算法的理论介绍肯定存在挂一漏万的现象,案例分析难免有不妥之处,诚挚地希望各位专家和读者批评指正。联系方式:15388542405@163.com。

编 者

2016年10月

目 录

第 1 章 绪论	1
1.1 群智能算法	1
1.2 支持向量机	2
第 2 章 粒子群算法	4
2.1 理论基础	4
2.2 基本粒子群算法	4
2.2.1 算法流程	4
2.2.2 函数寻优	7
2.3 一种非线性改变惯性权重的粒子群算法	11
2.3.1 算法改进思想	11
2.3.2 仿真实验	13
2.3.3 结果分析	15
2.4 案例分析	15
2.4.1 改进粒子群算法优化 BP 网络	15
2.4.2 PSO-BP 算法预测 LAN/WLAN 网络可靠性	18
2.4.3 PSO-BP 算法预测导线耦合串扰	36
第 3 章 人工鱼群算法	41
3.1 理论基础	41
3.2 基本人工鱼群算法	42
3.2.1 算法流程	42
3.2.2 函数寻优	45
3.3 基于视野递减反馈策略的改进人工鱼群算法	50
3.3.1 算法改进思想	51

3.3.2	仿真实验	52
3.3.3	结果分析	57
3.4	案例分析	58
3.4.1	改进人工鱼群算法求解路径优化问题	58
3.4.2	改进人工鱼群算法求解旅游线路问题	64
第4章	蚁群算法	77
4.1	理论基础	77
4.2	基本蚁群算法	77
4.2.1	算法流程	77
4.2.2	研究现状	79
4.3	基于粒子群和蚁群的混合连续优化算法	80
4.3.1	算法改进思想	80
4.3.2	仿真实验	84
4.3.3	结果分析	87
4.4	案例分析	88
4.4.1	蚁群算法求解中国旅行商的问题	88
4.4.2	蚁群算法求解三维路径规划问题	94
第5章	群集蜘蛛优化算法	105
5.1	理论基础	105
5.2	基本群集蜘蛛优化算法	105
5.2.1	算法流程	105
5.2.2	研究现状	107
5.3	自适应多种群回溯群居蜘蛛算法	108
5.3.1	算法改进思想	108
5.3.2	仿真实验	112
5.3.3	结果分析	116
5.4	案例分析	117
5.4.1	自适应多种群回溯群居蜘蛛算法求解 TSP 问题	117
5.4.2	基于离散群居蜘蛛算法的分簇 WSNs 路由优化	123
第6章	支持向量机	133
6.1	理论基础	133
6.2	支持向量机原理	136

6.2.1 支持向量分类机	136
6.2.2 支持向量回归学习机	140
6.2.3 Libsvm 软件包	143
6.2.4 建模流程与实现步骤	143
6.3 案例分析	147
6.3.1 基于 SVM 的 LAN/WLAN 集成网络可靠性的预测模型	147
6.3.2 基于 SVM 的野战地域通信网的可靠性预测	156
参考文献	161

第1章 绪 论

生命在长期进化的过程中,由最初的纷繁复杂发展成现在的和谐有序,都是演化与优化的体现。人类很早就从中得到启发,分析自然界现象并将其运用到现实生活中。例如,被野草划伤而发明锯子,受蝙蝠夜行启发而发明雷达,仿蜘蛛依靠液压结构行走而发明液压步行机等。随着人们对生命本质的不断了解,生命科学正以前所未有的速度迅猛发展,人工智能的研究也开始摆脱经典逻辑计算的束缚,大胆探索起新的非经典计算途径。在这种背景下,社会性动物(蚁群、鱼群、蜂群、群集蜘蛛等)的自组织行为引起了人们的广泛关注,许多学者对这种行为进行数学建模并用计算机对其进行仿真,这就产生了所谓的“群智能”。群智能算法(swarm intelligence algorithm, SIA)已经成为人工智能的一个重要分支。同时,随着计算机技术、传感器技术、存储技术和网络技术的迅猛发展,使得数据的复杂性膨胀趋势日益加剧,信息技术发展的瓶颈已经不再是数据的获取、传输和存储,而是数据的加工、分析和利用。如何从这些复杂数据中获取信息并转化为知识,即数据挖掘,是大数据时代人工智能领域的另一个研究热点。数据挖掘是指从数据中抽取模式、关联、变化、异常及有意义的结构信息,其价值在于利用数据挖掘技术改善模型。机器学习作为数据挖掘的有力方法,其关注的重点是计算机程序如何随着经验的积累自动地提高其任务处理性能的行为。其中,支持向量机(support vector machines, SVM)是一种通用、有效的机器学习方法。由于支持向量机具有扎实的理论基础、优秀的泛化能力,常用来解决分类和回归问题,在诸如手写数字识别、人脸图像识别、时间序列预测、非线性函数估计等领域得到成功的应用。

1.1 群智能算法

近十余年来,群智能算法作为新型的仿生类进化算法已成为众多研究者关注的焦点。生物学家观察研究发现,群居昆虫群体可以看作是一个分布式系统,个体的行为都很简单,智能也不高,但当它们一起协同工作时,整个系统却呈现出一种

高度结构化的群体组织,能够“凸显”出非常复杂(智能)的行为特征,完成一些远远超出单个个体能力负荷的复杂工作。例如,单只蚂蚁的能力极其有限,但当这些简单的蚂蚁组成蚁群时,却能完成筑巢、觅食、迁徙、清扫蚁巢等复杂行为。蚁群在觅食过程中总能找到一条从蚁巢到食物源的最短路径;记忆力只有 7 s 的简单小鱼可以随机游弋,也可以集体觅食,群聚和追尾,最终呈现出水域中食物最多之处必定鱼类数目最多的现象;一群行为显得盲目的蜂群能造出精美的蜂窝;群居生活的蜘蛛物种个体间保持有复杂的协作行为准则,根据雌雄执行捕食、交配、蜘蛛网设计及群体协作等。群智能算法就是模拟自然界生物群体的这些行为而构造的随机搜索方法。具有代表性的群智能算法主要有:粒子群优化算法(particle swarm optimization, PSO)、人工鱼群算法(artificial fish swarm algorithm, AFSA)、蚁群算法(ant colony optimization, ACO)和群集蜘蛛优化算法(social spider optimization algorithm, SSO)等。

由于生物群体中相互合作的个体是分布的,不存在直接的中心控制,因此,群智能更适合当前发展中的网络并行环境,并且具有较强的鲁棒性,即不会由于某一个或某几个个体出现故障而影响群体对整个问题的求解。单个个体能感知局部信息,但不能直接获得全局信息。个体行为简单、实现方便。个体间可以通过某种方式进行通信,整个群体协调合作完成某项工作。现实世界中有很多问题都可以归结为优化问题,如使用神经网络优化时的参数调整、网络路由优化和路径规划等。其中有相当重要的一部分优化问题,由于问题本身是动态的、多模型、无精确数学描述或无梯度信息的,所以无法使用传统数学方法来解决;而且传统基于梯度的优化算法普遍要求目标函数导数连续,具有计算复杂、串行求解等特征,所以群智能算法以其较强的鲁棒性,分布式计算机制,算法实现简单等特点,在函数优化、路径优化、网络路由、数据挖掘、机器人路径规划、无线传感器网络性能优化等领域获得了广泛的应用,并取得了较好的效果。近年来,大量的研究学者对群智能算法进行了深入的研究,有的针对某一种群智能算法参数设置改进,有的针对不同类型的优化问题对群智能算法改进,并将它们运用到了实际工程应用中。

1.2 支持向量机

20 世纪 90 年代迅速兴起的数据挖掘技术又称为知识库知识发现。数据挖掘就是从大规模数据库中搜索出有用的模式和重要规律的技术。数据挖掘受到很多学科领域的影响,其中数据库、机器学习、统计学影响最大。数据库提供数据管理技术,机器学习和统计学习提供数据分析技术。由于统计学往往注重理论而忽略了实际应用,因此,统计学领域提供的很多技术通常都要在机器学习领域进一步研

究,变成有效的机器学习算法后才能进入数据挖掘领域,从这个意义上来讲,机器学习是数据挖掘的支撑技术之一。

支持向量机是建立在统计学习理论和结构风险最小化原理基础上的机器学习方法,它在解决小样本、非线性和高维模式识别问题中表现出许多特有的优势,并在很大程度上克服了“维数灾难”和“过学习”等问题。此外,它具有坚实的理论基础,简单明了的数学模型。概括而言,SVM 具有其他机器学习方法难以比拟的优点,具体列举如下。

① 基于统计学习理论的 SVM 是基于小样本的统计学习方法,避免了传统方法追求样本趋于无穷时的渐近性能,更符合实际应用问题。

② 相对于传统的以经验风险最小化为准则的机器学习方法,SVM 遵循了结构风险最小化原则,即学习模型不追求训练样本分类错误最小,而是使得训练样本分类错误的概率上界最小,有效避免了过学习问题。

③ SVM 本质上是求解一个凸二次优化问题,理论上可以得到问题的全局最优解,克服了其他机器学习方法“局部极值”的问题。

④ SVM 借助核方法将原始空间的非线性可分问题通过映射变换到高维空间,转化为高维特征空间中的线性可分问题,利用核函数避开了求解显式的非线性映射函数,在保证模型泛化性能的同时,解决了维度灾难问题。

⑤ SVM 所得到的学习模型简单,仅由少数的支持向量决定,与非支持向量数据无关,具有优秀的自适应能力和良好的鲁棒性。

虽然支持向量机作为一种基于统计学习理论的数据挖掘新方法,近年来被广泛应用于实际问题中,但还是有很多问题没有解决,如核函数的选取问题,什么样的数据采用什么样的核函数;在大数据时代如何处理大规模数据的问题以及如何进一步提高其预测性能;针对大规模、非平衡、高维、多类、在线等复杂数据挖掘问题,如何建立相应的 SVM 模型。

第2章 粒子群算法

2.1 理论基础

粒子群算法是美国社会心理学家肯尼迪(Kennedy)和电气工程师埃伯哈特(Eberhart)于1995年提出的一种演化计算技术,其基本思想受他们早期对鸟类群体行为研究结果的启发,并利用了生物学家弗兰克·赫普纳(Frank Heppner)的生物群体模型。鸟群在迁徙过程中,在他们找到有食物的地方之前,既有分散又有群集的特点。鸟群无时无刻在相互传递着信息,这种个体之间的信息交换在群居生物群体之间都存在,如蜜蜂、蚂蚁。对于鸟群来说,在他们从一个地方到另一个地方的迁徙过程中,总是有那么一只鸟对食物的嗅觉较好,对食源的大致方向有较好的洞察力,从而,这只鸟就拥有对食源的较好信息。所以,在这种“好消息”的指引下,最终引导鸟群“一窝蜂”的奔向食源,达到了在食源的群集。解群指得是一个鸟群一地到另一地的迁徙,相当于解群的进化。“好消息”相当于解群每次迭代进化中的最优解,食源相当于全局最优解。粒子和鸟的区别是:在空间中的任意一点,只能有一只鸟,而粒子数目可以有多个。自粒子群算法提出以来,由于它的计算快速性和算法本身的易实现性,引起了国际上相关领域众多学者的关注和研究,其研究大致可以分为算法的改进、算法的分析以及算法的应用。

2.2 基本粒子群算法

2.2.1 算法流程

粒子群算法与其他进化类算法相类似,也采用“群体”与“进化”的概念,同样也

是依据个体(粒子)的适应值大小进行操作。所不同的是,粒子群算法不像其他进化算法那样对于个体使用进化算子,而是将每个个体看作是在 n 维搜索空间中的一个没有重量和体积的粒子,并在搜索空间中以一定的速度飞行。飞行速度由个体的飞行经验和群体的飞行经验进行动态调整。

设 $x_i = (x_{i1}, x_{i2}, \dots, x_{id})$ 为粒子的当前位置; $v_i = (v_{i1}, v_{i2}, \dots, v_{id})$ 为粒子的当前飞行速度; $p_i = (p_{i1}, p_{i2}, \dots, p_{id})$ 为粒子所经历的最好位置,也就是粒子所经历过的具有最好适应值的位置,称为个体最好位置(对应解空间中的个体极值);群体中的粒子数为 s ;群体中所有粒子所经历过的最好位置为 p_{gbest} ,称为全局最好位置(对应解空间中的全局极值)。

基本粒子群算法的进化方程为

$$v_{id}^{k+1} = v_{id}^k + c_1 \text{rand}() (p_{id} - x_{id}^k) + c_2 \text{rand}() (p_{gbest}^k - x_{id}^k) \quad (2.1)$$

式(2.1)中等式右边第一项为粒子先前的速度;第二项为“认知”部分,因为它仅考虑了粒子自身的经验,表示粒子本身的思考;第三项为“社会”部分,表示粒子之间的社会信息共享。如果基本粒子群算法的速度进化方程仅包含先前速度和认知部分,即

$$v_{id}^{k+1} = v_{id}^k + c_1 \text{rand}() (p_{id} - x_{id}^k)$$

则其性能变差。主要原因是不同的粒子间缺乏信息交流,即没有社会信息共享,粒子间没有交互,使得一个规模为 N 的群体等价于运行了 N 个单个粒子,因而得到最优解的概率非常小。若速度进化方程中仅包含先前速度和社会部分,即

$$v_{id}^{k+1} = v_{id}^k + c_2 \text{rand}() (p_{gbest}^k - x_{id}^k) \quad (2.2)$$

则粒子没有认识能力,也就是“只有社会(social-only)”的模型。这样,粒子在相互作用下,有能力到达新的搜索空间,虽然它的收敛速度比基本粒子群算法更快,但对于复杂问题,则容易陷入局部最优点。

从上述粒子进化方程可以看出, c_1 是调节粒子飞向自身最好位置方向的步长, c_2 是调节粒子向全局最好位置飞行的步长。为了减少在进化过程中粒子离开搜索空间的可能性, v_{id} 通常限定在 $[-x_{\max}, x_{\max}]$ 内,则可设定 $v_{\max} = kx_{\max}$, $0.1 \leq k \leq 1$ 。粒子在解空间内不断跟踪个体极值与全局极值进行搜索,直到达到规定的迭代次数或满足规定的误差标准为止。粒子在各维度飞行的速度不能超过算法设定的最大速度 $v_{\max}$ 。较大的 $v_{\max}$ 可以保证粒子种群的全局搜索能力,较小的 $v_{\max}$ 可以使粒子种群的局部搜索能力加强。

基本 PSO 算法流程可描述为以下过程。

① 首先初始化粒子群,即设置相关初始参数,如种群规模 s ,每个粒子的初始速度 v ,初始位置 x ,最大循环迭代次数 loop ,个体极值 p_i ,全局极值 p_{gbest} 。

② 计算各微粒的适应度函数值 partf 。

③ 对每个微粒,将其适应度值与个体极值 p_i 相比较,如果当前适应度值更优,则用当前适应度值更新 p_i ,将每个微粒的适应度值与全局极值 p_{gbest} 相比较,如果当前群体中最好的适应度值较好,则将其置为新的 p_{gbest} 。以求最小值为例,若 $partf < p_i$ 则 $p_i = partf$, 否则 p_i 不变;若 $partf < p_{gbest}$ 则 $p_{gbest} = partf$, 否则 p_{gbest} 不变。

④ 根据式(2.1)更新各微粒的位置和速度。

⑤ 比较当前迭代次数是否达到 loop 或满足最小误差精度,如果达到此结束条件,则此次迭代所得全局极值即为所求最优值,算法结束;否则返回步骤②重新计算粒子的适应度。

算法流程如图 2-1 所示。

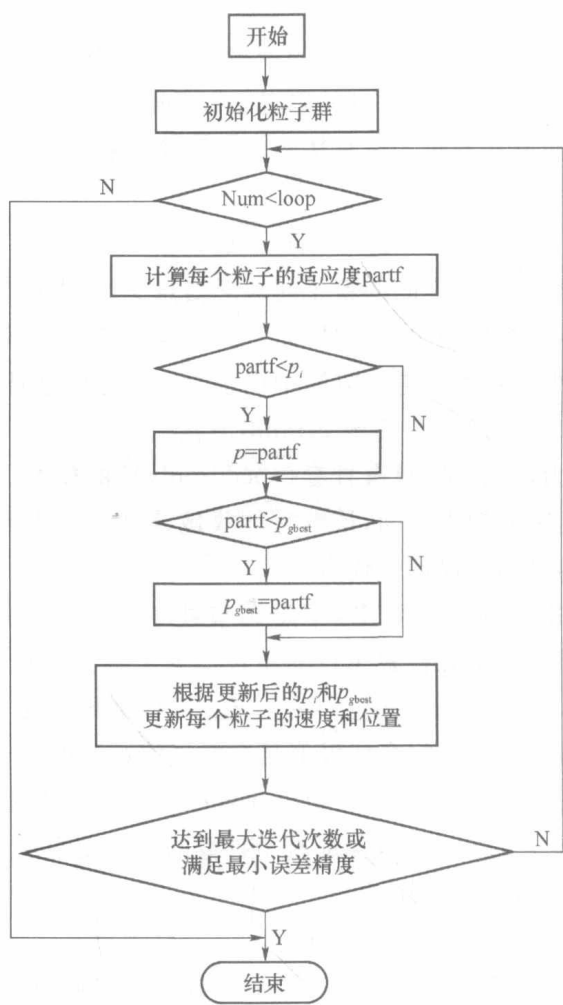


图 2-1 基本 PSO 算法流程图

为了改善基本 PSO 算法的收敛性能,埃伯哈特在 1998 年的 IEEE(电气和电子工程师协会)国际进化计算学术会议上首次在速度进化方程中引入惯性权重,对每一次迭代,粒子 i 在 d ($1 \leq d \leq D$) 维空间的运动遵循如下方程。

$$v_{id}^{k+1} = \omega v_{id}^k + c_1 \text{rand}() (p_{id} - x_{id}^k) + c_2 \text{rand}() (p_{g\text{best}} - x_{id}^k) \quad (2.3)$$

$$x_{id}^{k+1} = x_{id}^k + v_{id}^k \quad (2.4)$$

其中, ω 为惯性权(inertia weight),它使粒子保持运动的惯性,使其有能力探索新的区域; c_1 和 c_2 为加速常数(acceleration constants),它们使每个粒子向 p_{best} 和 $p_{g\text{best}}$ 位置加速运动; $\text{rand}()$ 为(0,1)范围内变化的随机数。

2.2.2 函数寻优

根据 PSO 算法原理,在 MATLAB 7.0.1 中编程实现基于 PSO 算法的函数极值寻优算法。

设置 PSO 算法的运行参数,程序代码如下。

```
%% 清空环境
clc
clear

%% 参数初始化
%粒子群算法中的两个参数
c1 = 1.49445;
c2 = 1.49445;

maxgen=300;    %进化次数
sizepop=20;    %种群规模

Vmax=0.5;
Vmin=-0.5;
popmax=2;
popmin=-2;
```

以非线性函数 $f(x) = \frac{\sin \sqrt{x^2 + y^2}}{\sqrt{x^2 + y^2}} + e^{\frac{\cos 2\pi x + \cos 2\pi y}{2}} - 2.71289$ 为例寻优,函数图像如图 2-2 所示。

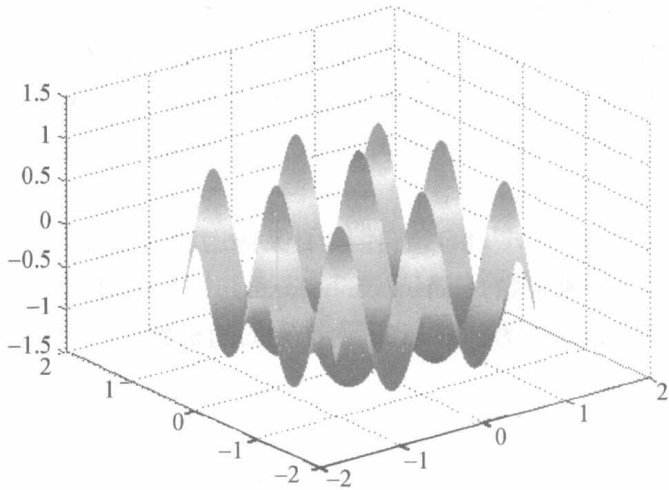


图 2-2 函数图形

适应度函数程序代码如下。

```
function y = fun(x)
%函数用于计算粒子适应度值
%x input 输入粒子
%y output 粒子适应度值
y=sin( sqrt(x(1).^2+x(2).^2) )./sqrt(x(1).^2+x(2).^2)+exp((cos(2 * pi * x
(1))+cos(2 * pi * x(2)))/2)-2.71289;
```

随机初始化粒子位置和粒子速度,然后根据适应度函数计算粒子适应度值,程序代码如下。

```
%% 产生初始粒子和速度
for i=1:sizepop
%随机产生一个种群
pop(i,:)=2 * rands(1,2); %初始种群
V(i,:)=0.5 * rands(1,2); %初始化速度
%计算适应度
fitness(i)=fun(pop(i,:)); %适应度
end
```

根据初始粒子适应度值寻找个体极值和群体极值,根据式(2.3)和式(2.4)更新粒子位置和速度,根据新粒子的适应度值更新个体极值和群体极值,程序代码如下。

```
%% 个体极值和群体极值
[bestfitness bestindex]=max(fitness);
zbest=pop(bestindex,:); %全局最佳
```



```

gbest=pop; %个体最佳
fitnessgbest=fitness; %个体最佳适应度值
fitnesszbest=bestfitness; %全局最佳适应度值
%% 迭代寻优
for i=1:maxgen
for j=1:sizepop
%速度更新
V(j,:)=V(j,:)+c1*rand*(gbest(j,:)-pop(j,:))+c2*rand*(zbest-
pop(j,:));
V(j,find(V(j,:)>Vmax))=Vmax;
V(j,find(V(j,:)<Vmin))=Vmin;
%种群更新
pop(j,:)=pop(j,:)+V(j,:);
pop(j,find(pop(j,:)>popmax))=popmax;
pop(j,find(pop(j,:)<popmin))=popmin;
%适应度值
fitness(j)=fun(pop(j,:));
end
for j=1:sizepop
%个体最优更新
if fitness(j) > fitnessgbest(j)
best(j,:)=pop(j,:);
fitnessgbest(j)=fitness(j);
end
%群体最优更新
if fitness(j) > fitnesszbest
zbest=pop(j,:);
fitnesszbest=fitness(j);
end
end
yy(i)=fitnesszbest;
end

```

PSO 算法反复迭代 300 次,画出每次迭代最优个体适应度值变化图形,如图 2-3 所示,程序代码如下。

```

plot(yy)
title('最优个体适应度','fontsize',12);
xlabel('进化代数','fontsize',12);ylabel('适应度','fontsize',12);

```

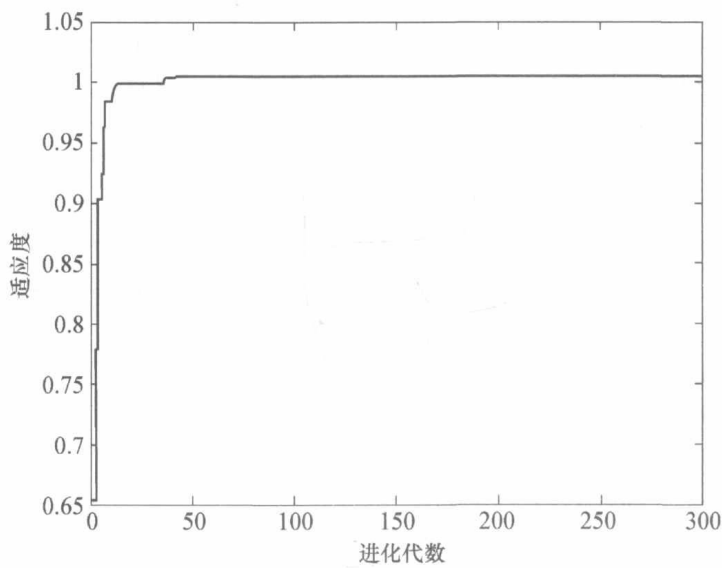


图 2-3 最优个体适应度值

最终得到的最优个体适应度值 (fitnesszbest) 为 1.005 4, 对应的粒子位置 (zbest) 为 $(-0.000\ 367\ 4, -0.000\ 560\ 49)$ 。PSO 算法寻优得到的最优值接近函数实际最优值, 这说明 PSO 算法具有相对较强的函数极值寻优能力。PSO 算法认为, 群体中的每个个体都可以从邻近个体的发现和以往经验中受益, 其理论基础主要包括三个基本因素: 刺激的评价, 与邻近的比较和对目前最优近邻的模仿。虽然 PSO 算法收敛快, 但是由于基本 PSO 算法依靠的是群体之间的合作与竞争, 粒子本身没有变异机制, 因而单个粒子一旦受某个局部极值约束后, 本身很难跳出局部极值的约束, 此时需要借助其他粒子的成功发现。事实上, PSO 算法的寻优能力主要来自于粒子之间的相互作用和相互影响, 如果从算法中去除粒子之间的相互作用和相互影响, 则 PSO 算法的寻优能力就会非常有限。算法在运行的初始阶段, 收敛速度较快, 但运行一段时间后, 速度开始减慢甚至停滞。当所有粒子的速度几乎为 0 时, 整个群体失去了进一步进化的能力, 可以认为算法已经收敛。而在许多情况下 (如多峰函数寻优), 算法并没有收敛到全局极值, 使得后期收敛速度明显变慢, 无法继续优化, 产生早熟收敛, 此时粒子群高度聚集, 严重趋向同一化 (失去了多样性), 粒子群会长时间或永远跳不出聚集点。因此, 很多学者都致力于提高 PSO 算法的性能。