

基于近似动态规划的微粒群系统参数优化研究

康琦^{1,2} 汪镭^{1,2} 安静^{1,3} 吴启迪^{1,2}

摘要 从系统最优控制的角度对微粒群参数的动态优化问题进行探讨. 针对离散动态规划的“维数灾”问题, 将群体启发式随机搜索机制引入动态规划的最优策略求解, 提出了一种群体智能近似动态规划模式; 基于该模式给出简化的确定型微粒群反馈控制系统参数优化的近似计算方法, 并扩展应用于具有随机变量的微粒群系统; 仿真计算得到了微粒群加速因子的近似最优动态规律, 并将所得策略与一种时变加速因子 (Time-varying acceleration coefficients, TVAC) 策略进行了函数优化性能的比较与分析, 初步实验结果表明该近似动态规划模式可有效地用于微粒群系统参数的动态优化设置.

关键词 群体智能, 近似动态规划, 微粒群系统, 参数优化

DOI 10.3724/SP.J.1004.2010.01171

Approximate Dynamic Programming Based Parameter Optimization of Particle Swarm Systems

KANG Qi^{1,2} WANG Lei^{1,2} AN Jing^{1,3} WU Qi-Di^{1,2}

Abstract From the perspective of optimal control, parameter dynamic optimization of particle swarm optimization (PSO) is addressed in this paper. This work is based on a type of simplified PSO and corresponding convergence conditions. First, to overcome the “curse of dimensionality”, a novel swarm approximate dynamic programming (SADP) is proposed by introducing the heuristic stochastic search mechanism of swarm intelligence. Second, grounded on SADP, parameter dynamic optimization and computation are studied in detail for a deterministic PSO feedback system and a stochastic PSO system, respectively. Further, numerical experiments are performed to show the effectiveness of SADP in parameter dynamic optimization of PSO systems through computing optimal dynamics of acceleration coefficients, as well as comparing the optimized strategies with a time-varying acceleration coefficients (TVAC) strategy based on several benchmarks.

Key words Swarm intelligence, approximate dynamic programming, particle swarm system, parameter optimization

微粒群优化 (Particle swarm optimization, PSO) 是由 Kennedy 和 Eberhart 于 1995 年提出的一种群体启发式全局优化技术^[1], 其基本思想源于对鱼群、鸟群等群居生物社会行为的模拟. 由于其概念简明、实现方便, 迅速得到了国际进化计算研究领域的认可和广泛关注^[2], 在算法的自适应性^[3-4]、参

数的优化设计^[4-5]、群体拓扑与组织进化^[6-7] 以及混合算法^[8-10] 等方面, 进行了大量的改进和仿真分析工作, 得到了较为丰富和系统的结果. 与此同时, 由于在解决复杂组合优化类问题方面具有优越性能, PSO 在工程设计与优化^[11]、电力系统优化^[12-13]、交通规划^[14] 和生产管理与调度优化^[8-15] 等领域得到了成功的应用.

PSO 中, 一定数量的微粒在寻优空间中结伴飞行, 每个微粒在自身运动状态和群体优化信息的影响下不断调整速度和方向, 使得群体在协作中逐步逼近寻优目标. D 维空间中, 定义微粒 i 的位置坐标为 \mathbf{x}_i ; 速度 \mathbf{v}_i 表示每次迭代微粒移动的距离. 微粒 i 在第 d ($d = 1 - D$) 维子空间中的飞行速度 v_{id} 和位置 x_{id} 根据式 (1) 和 (2) 进行调整:

$$v_{id}(t+1) = \omega v_{id}(t) + c_1 \text{rand}_1 [p_{id} - x_{id}(t)] + c_2 \text{rand}_2 [p_{gd} - x_{id}(t)] \quad (1)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (2)$$

其中, ω 为惯性权重 (Inertia weight); c_1, c_2 为加速因子 (Acceleration coefficients); rand_1 和 rand_2 分别依均匀分布在 $[0, 1]$ 产生随机数. p_{id} 表示微粒 i 的

收稿日期 2009-05-25 录用日期 2010-03-17
Manuscript received May 25, 2009; accepted March 17, 2010
国家自然科学基金 (70531020, 70871091), 教育部科学研究重大项目 (306023), 上海市科委科研计划项目 (08DZ1140500, 09DZ1141300), 教育部第 37 批留学回国人员科研启动基金, 同济大学青年优秀人才培养行动计划 (2009KJ028) 资助
Supported by National Natural Science Foundation of China (70531020, 70871091), Major Scientific Projects of Ministry of Education of China (306023), Science and Technology Commission of Shanghai Municipality (08DZ1140500, 09DZ1141300), Scientific Research Foundation for the Returned Overseas Chinese Scholars, Education of Ministry of China, and Program for Young Excellent Talents in Tongji University (2009KJ028)
1. 同济大学电子与信息工程学院 上海 201804 2. 同济大学教育部嵌入式与服务计算重点实验室 上海 201804 3. 上海应用技术学院机械与自动化工程学院 上海 200235
1. College of Electronics and Information Engineering, Tongji University, Shanghai 201804 2. Key Laboratory of Embedded System and Computer-service of Ministry of Education, Tongji University, Shanghai 201804 3. College of Mechanical and Automation Engineering, Shanghai Institute of Technology, Shanghai 200235

最佳位置记录, 而 p_{gd} 记录的是群体最佳位置; 两者与微粒当前位置之差在施加一定程度的随机化之后, 被分别用于引导微粒朝其最佳位置和群体最佳位置方向运动. 为防止微粒飞离有效寻优空间, 通常设定最大速度值 v_{\max} , 向量 \mathbf{v}_i 被限制在 $[-v_{\max}, v_{\max}]$.

Eberhart 和 Shi 在 *IEEE Transactions on Evolutionary Computation: Special Issue on PSO (2004)* 卷首语中指出, PSO 研究的主要方向及热点包括 PSO 算法分析、群体拓扑结构、参数选择与优化、与其他演化计算的融合及 PSO 算法应用等. 特别是在参数选择与优化方面, 研究者从不同的角度做了大量的工作, 积累了较为丰富的结果. 作为 PSO 的重要参数, 惯性权重 ω 影响微粒群的搜索范围和精度, 通常较大的 ω 值有利于群体搜索更大的空间, 而较小的 ω 值有助于在当前解空间附近挖掘更好的解. 因此, 在不同阶段根据需要动态设置 ω 可使微粒群获得更佳的寻优性能. Shi 和 Eberhart^[16] 提出了时变惯性权重 (Time-varying inertia weight, TVIW) 和随机惯性权重 (Random inertia weight, RANDIW) 的 PSO 算法, TVIW 允许 ω 随迭代进程线性减少, 群体搜索范围随之逐渐缩小, 目的是让微粒群充分探索新区域之后能对当前最优解邻域进行细致地搜索; RANDIW 采用 ω 随机取值的方式来保持种群的多样性, 以提高算法对动态问题的求解能力. 不过, 微粒群寻优应该是一种非线性动态过程, 线性或随机参数调整策略难以体现这一特征. 为此, Shi 等^[17] 采用模糊规则来自适应动态调整 ω ; 张长胜等^[18] 基于一种可控制吸引和排斥动态自适应切换的群体多样性保持模式, 采用 ω 动态变化的策略来提高算法的收敛速度, 取得了不错的效果. Han 等^[19] 则从闭环控制的角度来研究参数 ω 的动态优化问题, 通过设置 PID 控制器动态调节 ω , 以保证群体的多样性和较好的寻优能力. 除 ω 外, 加速因子 c_1, c_2 对于平衡全局搜索和局部开发能力也起着重要作用, 研究者对如何设置两者之间的关系、如何动态调整取值策略等进行了大量的仿真与分析工作, 提出了许多有效的参数选择与优化策略. 例如, Kennedy 与 Eberhart^[1] 建议采用恒定的加速因子 $c_1 = c_2 = 2.0$, 该策略被广泛采用; Suganthan^[20] 采用一种线性增加 c_1, c_2 的方法来改进算法寻优性能; Zhan 等^[3] 提出一种系统化的参数自适应调整模式, 利用进化状态估计技术对群体分布信息进行估计, 并根据进化状态 (探索、开发、收敛和跳出等) 来自适应调整 ω 和 c_1, c_2 等参数; Asanga 等^[4] 在 TVIW 的基础上, 提出一种时变加速因子 (Time-varying acceleration coefficients, TVAC) 的策略来提升算法性能; 在 TVIW 和 TVAC 的基础上, Ko 等^[13] 提出一种非线性时变进化 (Nonlinear time-

varying evolution, PSO-NTVE) 策略来改进微粒群的寻优性能, 该方法分别定义了 ω 和 c_1, c_2 的非线性进化系数 α, β, γ , 并采用正交矩阵技术来确定 α, β, γ 的最优组合; de Oca 等^[21] 从组元差异的角度分析参数设置对算法优化性能的影响, 将每个参数的一种优化设置模式看作 PSO 组元, 提出了一种包含若干 PSO 组元的优秀改进算法 Frankenstein's PSO. 此外, 有研究者从群体规模^[7, 22]、速度限幅 v_{\max} ^[23] 和最优值更新速率^[24] 等方面来研究改善算法性能.

总体而言, PSO 参数选择与优化方法层出不穷, 形式多样、各具千秋. 但这些方法都是通过人为设定某种规则来控制相应参数的变化规律, 并且大多是基于经验的, 或是通过实验试凑法获得的, 所得结果的可移植性较差. 不难理解, 如果能够从微粒自身运动行为分析的角度来解析参数对算法的影响, 就有可能从中发现某种与微粒群运动特性相吻合的参数优化控制规律. 那么, 能否按照该思路研究 PSO 参数的优化设置问题呢? 如果将微粒群寻优看作一个动态系统, 其中相关参数构成影响系统演化动态的内部因子, 那么在保证系统稳定或收敛条件下, 对参数采取合理有效的动态规划与控制可以产生优化的系统演化输出, 贝尔曼最优化原理为之提供了一种行之有效的途径. 于是, 作者产生借助最优控制原理进行微粒群系统参数优化的想法. 因此, 本文尝试从系统最优控制的视角, 利用近似动态规划的思想对离散 PSO 系统参数的动态优化设置问题进行探讨, 以期得到一种研究微粒群参数优化的新思路.

1 简化 PSO 动态系统

为深入了解微粒群优化的行为模式, 解析相应参数与算法性能的关系, 研究者对微粒群的运动和演化规律进行了一定程度的近似假设与简化处理, 分别将微粒群描述为几类典型的动态系统: 线性时不变系统、线性时变系统、反馈控制系统以及随机系统等. Kennedy^[25] 首先对微粒群的运动行为进行了分析, 计算模拟了不同参数设计下的微粒运动轨迹. Ozcan 和 Mohan^[26] 对单个微粒在一维空间的运动进行分析, 证明其轨迹满足幅值与频率可调的正弦波规律. 之后, 研究者通过分析微粒群的收敛性来求解 PSO 参数的设置条件. 在与文献 [26] 相同的假设前提下, Clerc 和 Kennedy^[27] 对 PSO 的不同参数组合模型下单个微粒的离散运动轨迹进行分析, 得到一个可以用于控制系统收敛趋势的参数集合和通用算法模型. 在同样的假设条件下, Trelea^[28] 则针对标准 PSO 模型提出系统的稳定区间和不稳定区间, 并对呈微粒的运动轨迹进行全面分析, 得到了群体运动轨迹收敛、谐波振荡和锯齿形时对应的参

数空间. van den Bergh^[29] 从 PSO 全局收敛和局部收敛等行为的分析得到保证 PSO 收敛的参数设置. 潘峰等^[30-31] 将粒子运动的一般性描述图解为历史状态加权的形式, 分析了标准 PSO 的优化机理, 并对 PSO 参数的选择与微粒群早熟收敛情况进行了运动特性分析, 给出了系统渐近稳定的充分条件.

以上研究从保证系统收敛性的角度进行了参数分析, 得到了相应参数的设置范围, 这是参数优化研究的基础. 因此, 本文基于最优控制的 PSO 系统参数动态优化也将建立在这些工作之上.

将 PSO 视为一个反馈控制动态系统, 并保持 ω 恒定. 该系统可表示为一个时不变线性系统与时变增益输出反馈控制的组合, 系统状态空间描述为

$$\mathbf{X}_{t+1} = \mathbf{A}\mathbf{X}_t + \mathbf{B}\mathbf{u}_t \quad (3)$$

$$\mathbf{y}_t = \mathbf{C}\mathbf{X}_t \quad (4)$$

其中, $\mathbf{A} = \begin{bmatrix} 1 & \omega \\ 0 & \omega \end{bmatrix}$, $\mathbf{B} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$, $\mathbf{C} = \begin{bmatrix} 1 & 0 \end{bmatrix}$, $\mathbf{u}_t = -\varphi_t(\mathbf{x}_t - \tilde{\mathbf{p}}_t) = \mathbf{F}_t(\varphi_t)\tilde{\mathbf{X}}_t$ 表示控制输入信号, $\tilde{\mathbf{p}}_t = \frac{\phi_1 \cdot \mathbf{p}_t + \phi_2 \cdot \mathbf{g}\mathbf{b}_t}{\phi_1 + \phi_2}$.

基于状态空间描述的 PSO 反馈控制动态系统, Kadiramanathan 等^[32] 进行了系统稳定性分析, 并证明了 PSO 系统渐近稳定的充分条件为

$$|\omega| < 1, \omega \neq 0, \Phi = \sup(\varphi_t) < \frac{2(1 - 2|\omega| + \omega^2)}{1 + \omega} \quad (5)$$

本文针对 PSO 动态系统状态空间描述, 在不等式 (5) 给出的系统稳定约束下, 运用最优控制原理对 PSO 系统参数进行动态优化计算, 寻找相应参数的最优动态轨迹, 即参数的最佳动态设置策略.

为了求解上述 PSO 系统的最优控制问题, 定义系统的性能指标:

$$J_{i,N}(\mathbf{X}_i) = \min \left\{ \text{Obj}(\mathbf{X}_N) + \sum_{t=i}^{N-1} \left[\text{Obj}(\mathbf{X}_t) + \frac{1}{2} \mathbf{u}_t^T \mathbf{u}_t \right] \right\} \quad (6)$$

其中, $\text{Obj}(\cdot)$ 为待优化的目标函数, N 表示最大迭代步数.

2 基于近似动态规划的 PSO 系统参数优化

动态规划^[33] 由美国学者贝尔曼于上世纪 50 年代创立, 它是一种求解决策过程最优化的数学方法, 其基本思想为贝尔曼最优化原理. 动态规划可有效地区用于最优控制问题的求解. 对于某些理想的系统,

例如具有二次效用函数并且目标值为 0 的线性系统, 可通过求解 HJB (Hamilton-Jacobi-Bellman) 方程来获取最优代价与最优控制律. 然而, 目标函数通常是非线性及多模的, HJB 方程的理论解很难直接求得^[34]. 离散动态规划进行具体问题求解时, 需要在每个离散时刻计算相应的数值解, 当状态空间维数增加时, 计算复杂度呈指数增加 (“维数灾”)^[35]. 针对 “维数灾” 问题, 研究者提出了许多近似的解决方法^[34-36]. 通用方法是基于 Cost-to-go 函数近似^[34], 设计一个固定的结构体系, 如多项式近似、仿射、多变量自适应衰减仿射和神经网络等, 并利用离散状态空间的采样数据进行结构参数的训练, 称为 “近似动态规划 (Approximate dynamic programming, ADP)”. 在利用动态规划进行最优控制求解的计算机实现时, 无论离散动态系统, 还是连续系统, 都不可能求得一个最优的通解, 而是要进行离散化处理, 并对一系列离散数据来近似拟合其动态轨迹. 在每个阶段寻找最优控制量时, 需要对所有控制量进行计算比较才能获得理论上的最优解. 然而, 这种理论最优解是无法获得的, 需要采取某种替代方法来获取近似的最优解. 由前述可知, 作为群体智能的典型实现模式, 微粒群算法概念简单、易于计算机实现且搜索速度快, 特别适合复杂的组合优化问题. 针对离散动态规划的计算特征和需要, 易想到微粒群启发式随机寻优机制可用于海量控制策略的近似优化求解. 因此, 本文将群体启发式随机优化引入近似动态规划设计, 提出一种群体智能近似动态规划模式 (Swarm approximate dynamic programming, SADP), 并用于 PSO 系统参数的动态优化求解.

2.1 基于群体智能的近似动态规划

经文献检索, 群体智能与动态规划相结合的研究并不多, 而且大多是利用各自的特点分别求解复杂优化问题中不同的子问题, 或者从不同的层面进行优化计算. 在利用群体智能进行近似动态规划方法改进方面, 所做工作也仅是利用微粒群算法对某种近似动态规划模式进行神经网络自适应训练^[37]. 鉴于离散动态规划问题与 TSP 问题的相似性, Sitarz^[38] 利用蚁群算法求解一种多准则离散动态规划问题. Lu^[39] 利用启发式动态规划模式来动态调整 PSO 参数, 如惯性权重 ω 、加速因子 c_1, c_2 等. 与以上工作不同, 本文将群体启发式搜索机制用于动态规划的控制策略寻优, 给出一种新型的近似求解思路.

对离散时间系统 $\mathbf{x}_{k+1} = T(\mathbf{x}_k, \mathbf{u}_k)$, 给定初始代价 V_0 , 定义 SADP 的迭代方程为

$$V_{k+1}(\mathbf{x}) = \min_{\mathbf{u}_k \in \Omega} \{ r(\mathbf{x}, \mathbf{u}_k) + V_k(T(\mathbf{x}, \mathbf{u}_k)) \} \quad (7)$$

其中, $r(\mathbf{x}, \mathbf{u}_k)$ 表示对状态 \mathbf{x} 施加控制 \mathbf{u}_k 后的代价返回, 包含对微粒适应度的评价。

SADP 中, 采用微粒群寻优来代替传统求解 HJB 方程, 分别在每一个控制时刻利用 PSO 逐步确定最优控制序列, 从而找到系统最优控制律与最优性能指标的关系. 对于 n 维空间, 用个体表示控制向量 $\mathbf{u}(i) = (u_1(i), \dots, u_n(i))$, 其中的元素定义为实数. 对于单变量控制问题, 每个微粒代表一个控制量. 群体启发式随机寻优机制的引入, 有效避免了试图对所有控制量进行计算来获取理论最优解, 而是在动态系统的有效策略范围内寻找一个近似最优的控制律. 对于定义在高维空间的最优控制问题, 由于 PSO 所具有的随机搜索和群体协作优化的特性, SADP 的计算复杂度较理论最优解的求解方法将大大减少, 优点也将更为明显.

首先给出以下定义:

定义 1 (可行状态空间 Ω_{AS}). 在 n 维空间中, 有效轨迹必须同时满足其可到达的空间 $\Omega_R \subset \mathbf{R}^n$ 和可控空间 $\Omega_C \subset \mathbf{R}^n$. 即从当前状态出发, 经过任意有限时间一定可以到达系统的平衡点 (原点), 且当前状态需从其给定的初始状态集转移得到. 因此, 可行状态集 Ω_{AS} 定义为 $\Omega_{AS} = \Omega_R \cap \Omega_C \subset \mathbf{R}^n$.

定义 2 (有效策略空间 Ω_{AP}). 对应于系统的可行状态空间 Ω_{AS} , 所采用的控制策略构成了一个 l 维有效策略空间 $\Omega_{AP} \subset \mathbf{R}^l$.

定义 3 (最优策略网络 (OPN) 与最优策略序列网络 (OPSN)). OPN 用于记录微粒群获得的最优控制策略, 由一定数量的离散学习单元 (信息位) 组成. OPSN 按照一定的次序记录群体得到的最优控制序列, 是 OPN 的子集, 即 $OPSN \subset OPN$.

定义 4 (最优代价网络 (OCN)). 存储每一步控制策略付出的代价 $\tilde{J}(x)$. 求解贝尔曼方程 $V(\mathbf{x}_t) = \min_{\mathbf{u}_t \in \Omega_{AP}} r(\mathbf{x}_t, \mathbf{u}_t) + V(T(\mathbf{x}_t, \mathbf{u}_t))$ 时, 状态 $\mathbf{x}_{t+1} = T(\mathbf{x}_t, \mathbf{u}_t)$ 的 $V(T(\mathbf{x}_t, \mathbf{u}_t))$ 值可根据 $\tilde{J}(\mathbf{x})$ 求得.

定义 5 (线性插值近似). 令 $\tilde{J}(x_a)$, $\tilde{J}(x_b)$ 分别记录从状态 x_a , x_b 出发的系统最优代价. 如果 \bar{x} 表示从状态 x 的一个状态转移, 且位于 x_a 和 x_b 之间, 那么从状态 x 出发的代价值 $\tilde{J}(\bar{x})$ 可计算如下:

$$\tilde{J}(\bar{x}) = \tilde{J}(x_a) + \frac{\bar{x} - x_a}{x_b - x_a} [\tilde{J}(x_b) - \tilde{J}(x_a)] \quad (8)$$

在控制策略 $\tilde{\mathbf{u}}(x)$ 和代价网络 $\tilde{J}(x)$ 中, 记录了有限个数的离散状态值. 状态转移得到的新状态并不能保证与 $\tilde{J}(x)$ 中的状态完全匹配. 通过对有限离散状态的代价值进行线性插值近似, 可以得到最优代价网络 $\tilde{J}(x)$ 中所有状态的近似值. 每一步迭代采样状态越多, 那么经过线性插值拟合得到的曲线将更光滑且更接近于真实值.

2.2 PSO 反馈控制系统参数动态优化

基于群体智能近似动态规划模式的 PSO 参数优化过程可描述如下:

步骤 1. 初始化状态向量 $\mathbf{X}_0 \subset \mathbf{R}^n$: 在一维可行状态空间内进行随机初始化, $\mathbf{X}_0 = (X_0(1), X_0(2), \dots, X_0(n))$, 包含 n 个采样状态; 并设置初始的代价函数 $V(\mathbf{X}_0) = 0^n$; 相应地, 对于连续状态 $\mathbf{X} \subseteq \Omega_{AS}$ 的最优代价网络 \tilde{J} 的初始值为 $\tilde{J}_0(\mathbf{X}) = 0$ (网络初始化).

步骤 2. 执行状态转移操作 $\mathbf{X}_t \rightarrow \mathbf{X}_{t+1}$: 在阶段 t 中, 对每一个采样状态 $\mathbf{X}_t(s)$, $s = 1, 2, \dots, n$, 根据以下 PSO 动态方程 (状态方程) 求解状态转移:

$$\mathbf{X}_{t+1}(s) = A\mathbf{X}_t(s) + B\mathbf{u}_t(s) \quad (9)$$

$$\mathbf{u}_t(s) = -\varphi_t(x_t - \bar{p}) = \mathbf{F}_t(\varphi_t)\mathbf{X}_t(s) \quad (10)$$

执行控制策略 $\mathbf{u}_t(s) \subset \Omega_{AP}$ 将会产生状态转移.

步骤 3. 控制策略的启发式随机优化: 对由式 (9) 和 (10) 描述的 PSO 动态反馈控制系统, 利用 PSO 进行控制策略 $\mathbf{u}_t(s) \subset \Omega_{AP}$ 的动态优化计算, 寻找最优控制策略.

步骤 3.1. 微粒编码与初始化. 对于采样状态 $\mathbf{X}_t(s)$, 定义微粒群 $PS(s)$. 微粒 i ($i = 1, 2, \dots, N$) 的位置 $x_i(s)$ 表示该阶段对应状态 $\mathbf{X}_t(s)$ 处施加的控制策略, 并在有效策略空间 Ω_{AP} 中进行随机初始化. 对于一个 l 维控制系统, 微粒的位置描述为 $x_i(s, j) = u_{t,i}(s, j)$, $j = 1, 2, \dots, l$, 表示在可行空间 Ω_{AP} 中的第 j 维控制分量; 微粒的速度 $v_i(s)$ 定义为控制量的调整幅度, 即 $v_i(s) = \Delta u_{t,i}(s)$. 通过调整速度值实现在空间 Ω_{AP} 中的控制策略寻优.

步骤 3.2. 在阶段 t , 分别对采样状态 $\mathbf{X}_t(s)$, $s = 1, 2, \dots, n$ 在控制的有界可行空间 $\mathbf{u}_t(s) \in \Omega_{AP}$ 内执行群体寻优. 微粒 i 的速度及位置同步更新:

$$\begin{cases} \Delta u_{t,i}(s, k+1) = \omega \Delta u_{t,i}(s, k) + \\ \quad c_1 r_1(s, k) [p_{t,i}(s, k) - u_{t,i}(s, k)] + \\ \quad c_2 r_2(s, k) [g_{t,i}(s, k) - u_{t,i}(s, k)] \\ u_{t,i}(s, k+1) = u_{t,i}(s, k) + \Delta u_{t,i}(s, k+1) \end{cases} \quad (11)$$

步骤 3.3. 对于微粒 i , 根据式 (12) 计算适应度值:

$$FI(u_{t,i}(s, k)) = \min_{u_{t,i}(s, k) \in \Omega_{AP}} \{r(\mathbf{X}_t(s)), u_{t,i}(s, k) + V_t(T(\mathbf{X}_t(s), u_{t,i}(s, k)))\} \quad (12)$$

式中 $r(\mathbf{X}_t(s), u_{t,i}(s, k)) = Obj(\mathbf{X}_t(s)) + \frac{1}{2} u_{t,i}^T(s, k) \times u_{t,i}(s, k)$. 其中, $Obj(\cdot)$ 代表目标函数, 并且

$V_t(T(\mathbf{X}_t(s), \mathbf{u}_{t,i}(s, k))) = \tilde{J}_t(T(\mathbf{X}_t(s), \mathbf{u}_{t,i}(s, k)))$, 其值从最优代价网络 $\tilde{J}_t(\mathbf{X})$ 中获得.

步骤 3.4. 计算每个微粒的适应度 $FI(\mathbf{u}_{t,i}(s, k))$, 并对个体最优位置进行更新:

$$p_{t,i}(s, k+1) = \begin{cases} p_{t,i}(s, k), & FI(\mathbf{u}_{t,i}(s, k+1)) \geq FI(p_{t,i}(s, k)) \\ \mathbf{u}_{t,i}(s, k+1), & FI(\mathbf{u}_{t,i}(s, k+1)) < FI(p_{t,i}(s, k)) \end{cases} \quad (13)$$

群体的全局最优位置 $g_t(s, k)$ 计算如下:

$$g_t(s, k) = \arg \min_{p_{t,i}} FI(p_{t,i}(s, k)), \quad 1 \leq i \leq n \quad (14)$$

步骤 3.5. 令 $k = k + 1$, 执行寻优迭代, 直到 $k = C_{\max}$ 为止. 对于采样状态 $\mathbf{X}_t(s)$, $s = 1, 2, \dots, n$, 最优控制策略 $\mathbf{u}_t^*(s) = g(s, C_{\max})$, 记为 $\tilde{\mathbf{u}}_t(s)$ 并更新最优控制策略网络 $\tilde{\mathbf{u}}_t = [\tilde{u}_t(1) \ \tilde{u}_t(2) \ \dots \ \tilde{u}_t(n)]$.

步骤 3.6. $t = 0, 1, \dots$, 最优代价函数:

$$V_{t+1}(\mathbf{X}_t(s)) = V_t[T(\mathbf{X}_t(s), \mathbf{u}_t^*(s))] + \{Obj(\mathbf{X}_t(s)) + \frac{1}{2} \mathbf{u}_t^*(s)^T \mathbf{u}_t^*(s)\} \quad (15)$$

其中, $V_t(T(\mathbf{X}_t(s), \mathbf{u}_t^*(s))) = \tilde{J}_t(T(\mathbf{X}_t(s), \mathbf{u}_t^*(s)))$.

在阶段 t , 依次运用最优控制序列 $\mathbf{u}_{t-1}^*(s) = \{u_0^*(s), u_1^*(s), \dots, u_{t-1}^*(s)\}$ 来计算最优代价函数值 $V_t(\mathbf{X}_t(s))$, 记录为 $J_t(\mathbf{X}_t(s))$. 储存 $\mathbf{u}_{t-1}^*(s)$ 并更新最优控制序列网络 $\tilde{\mathbf{u}}$. 对所有状态 \mathbf{X} , 通过线性插值来近似求解最优代价网络 $\tilde{J}_t(\mathbf{X})$.

步骤 4. 令 $t = t + 1$, 重复执行步骤 2 和步骤 3, 直到满足 $V_{t+1}(\mathbf{X}) - V_t(\mathbf{X}) < \varepsilon$ 为止, ε 是一个任意正实数, 相应地, $\tilde{J}_t(\mathbf{X}) \rightarrow \tilde{J}(\mathbf{X})$; 或满足条件 $t = T_{\max}$ 时, 终止算法. 输出最优控制序列 $\mathbf{u}_t^* = [\underline{u}_t^*(1) \ \underline{u}_t^*(2) \ \dots \ \underline{u}_t^*(n)]^T$.

步骤 5. 对最优控制序列 \mathbf{u}_t^* 进行数据拟合与分析, 得出相应参数的优化结果, 用来指导 PSO 系统的参数优化设置.

2.3 随机 PSO 系统参数的群体智能近似动态优化

前述的 PSO 收敛性分析与参数优化研究均舍弃了标准算法中的随机因子. 但由于随机性的缺失, 所得结果偏离了微粒群系统的本质. 金欣磊等^[40] 在已有文献的假设前提和随机系统理论基础上, 从随机过程角度出发对 PSO 进行分析推导, 把带随机因素的线性时变系统在概率意义下转换成线性定常系统加以分析研究, 给出了其动力学系统依均方收敛的一个充分条件. Jiang 等^[41] 对标准 PSO 给出了一个包含随机量的收敛性分析框架, 将微粒在每一步进化中的位置看作一个随机变量, 利用随机过程理论进行分析, 得到了 PSO 系统的随机收敛条件

以及相应的参数选择策略. Fernández 等^[42] 将 PSO 系统看作是一种随机阻尼质量-弹簧系统的离散化模式, 针对一类时延模型推导出其一阶和二阶稳定区域和相应的谱半径, 得到了相关参数的最优范围. Kadiramanathan 等^[32] 将微粒动态描述为一个非线性反馈控制系统, 摒弃参数不含随机量的假设, 对随机微粒动态进行稳定性分析. 这些工作对于认识微粒群行为和计算模式具有重要意义.

对于随机系统的最优控制问题, 可利用统计方法转化为确定系统的控制问题, 再利用动态规划进行最优控制计算. Chen 等^[43] 提出了一种随机动态规划方法, 首先对测量数据进行标准 Kalman 滤波, 对未知的状态向量进行估计, 然后将估计向量反馈值重新进行状态测量. 本文基于这种思路, 采用 SADP 对具有随机变量的 PSO 系统进行参数的动态优化.

对于带随机变量的 PSO 系统:

$$\mathbf{X}_{t+1} = A\mathbf{X}_t + B\mathbf{u}_t(\vartheta) \quad (16)$$

$$\mathbf{u}_t(\vartheta) = \mathbf{F}_t(\varphi, \vartheta)\mathbf{X}_t = -\varphi_t(\vartheta)(\mathbf{x}_t - \tilde{\mathbf{p}}) \quad (17)$$

$\varphi_t(\vartheta)$ 中包含随机变量 ϑ .

定义系统的性能指标为

$$J_{I_{\max}}(\mathbf{X}_0) = \min \left\{ Obj(\mathbf{X}_N) + \sum_{t=0}^{I_{\max}-1} [Obj(\mathbf{X}_t) + \frac{1}{2} \mathbf{u}_t^T(\vartheta)\mathbf{u}_t(\vartheta)] \right\} \quad (18)$$

其中, $Obj(\cdot)$ 为系统目标函数(最小值), I_{\max} 表示最大的迭代周期(动态规划阶段数).

随机系统依照概率分布进行状态转移. 为了简化分析, 对系统状态观测过程进行状态估计(预测), 已知随机变量的概率分布, 可以通过积分求其状态估计. 因此, 通过状态估计可以将未知状态转化为确定状态, 该过程称为“滤波”. “滤波”之后, 随机最优控制问题被转化为一个确定的最优控制问题.

定义 6 (状态估计“滤波”). 通过状态估计, 将随机最优控制问题转化为确定系统的最优控制, 状态滤波 $\bar{\mathbf{X}}_{t+1}$ 可以根据下式进行计算:

$$\bar{\mathbf{X}}_{t+1} = E\{\mathbf{X}_{t+1}\} = \int T_t(\mathbf{X}_t, \mathbf{u}_t)p[d\vartheta_t | (\mathbf{X}_t, \mathbf{u}_t)] \quad (19)$$

定义 7 (代价估计“滤波”). 通过对代价函数的估计, 将随机最优控制转化为确定系统的最优控制, 即:

$$\bar{J}_t(\mathbf{X}_{t+1}) = E\{J_t(\mathbf{X}_{t+1})\} = \int J_t[T_t(\mathbf{X}_t, \mathbf{u}_t)]p[d\vartheta_t | (\mathbf{X}_t, \mathbf{u}_t)] \quad (20)$$

基于定义 6 或定义 7 进行“滤波”之后, 随机 PSO 系统转化为一个不含随机变量的确定系统, 然后再利用相应的最优控制方法进行优化求解.

基于 PSO 确定系统收敛条件, 利用统计分析 $\bar{\varphi} = E\varphi_t$, 计算得到参数的约束条件为

$$0 < \bar{\varphi} < 2(1 + \omega), \quad 0 < \omega < 1 \quad (21)$$

基于群体智能近似动态规划的随机 PSO 系统参数优化过程描述如下:

步骤 1. 定义 n 维系统状态 \mathbf{X} , 并对状态向量 \mathbf{X}_0 进行随机初始化; 设置初始的代价函数为 $J(\mathbf{X}_0) = 0^n$.

步骤 2. 根据下述的 PSO 动态方程 (状态方程) 进行状态转移操作 $\mathbf{X}_t \rightarrow \mathbf{X}_{t+1}$:

$$\mathbf{X}_{t+1} = T[\mathbf{X}_t, \mathbf{u}_t(\vartheta)] = A\mathbf{X}_t + B\mathbf{u}_t(\vartheta) \quad (22)$$

每个带有随机量的控制 $\mathbf{u}_t(\omega)$ 将会产生一个不确定的状态转移. 利用状态估计“滤波”法来处理状态转移, 计算得到相应的状态转移期望值:

$$\bar{\mathbf{X}}_{t+1} = E\{\mathbf{X}_{t+1}\} = \int T(\mathbf{X}_t, \mathbf{u}_t(\vartheta))d\vartheta \quad (23)$$

步骤 3. 对 PSO 动态系统进行群体启发式优化.

步骤 3.1. 微粒的编码和初始化. 定义状态变量 $\mathbf{X}_t(s)$ 对应的微粒群体 PS , 微粒 i ($i = 1, 2, \dots, N$) 的位置 $x_i(s)$ 表示为状态 $\mathbf{X}_t(s)$ 时施加于系统的控制量. 对于一维的控制系统, $x_i(s) = u_{t,i}(s)$. 微粒的速度变量 $v_i(s)$ 定义为对控制量的调整, 即 $v_i(s) = \Delta u_{t,i}(s)$.

步骤 3.2. 对状态向量 \mathbf{X}_t 进行采样, 并对每个采样状态 $\mathbf{X}_t(s)$, $s = 1, 2, \dots, n$, 利用式 (24) 进行有界控制 $\mathbf{u}_t \in \Omega_{AP}$ 的优化, 得到最优控制策略.

$$\hat{J}_{t+1}(\mathbf{X}_t(s)) = \min_{\mathbf{u}_t \in \Omega_{AP}} \{ [Obj(\mathbf{X}_t(s)) + \frac{1}{2}|\bar{\mathbf{u}}_t|^2] + \alpha \cdot \hat{J}_t[\bar{\mathbf{X}}_{t+1}(s)] \}$$

或者

$$\min_{\mathbf{u}_t \in \Omega_{AP}} E\{ [Obj(\mathbf{X}_t(s)) + \frac{1}{2}\mathbf{u}_t^T \cdot \bar{\mathbf{u}}_t] + \alpha \cdot J_t(\mathbf{X}_{t+1}(s)) \} \quad (24)$$

步骤 3.3. 阶段 t , 分别对每个采样状态 $\mathbf{X}_t(s)$, $s = 1, 2, \dots, n$, 在可行空间 $\mathbf{u}_t(s) \in \Omega_{AP}$ 内进行有界控制 $\mathbf{u}_t(s)$ 的群体优化求解. 微粒 i ($i = 1, 2, \dots, N$) 的速度及其同步位置更新通过式 (11) 来实现. 对微粒 i , 适应度评价函数定义为

$$FI(u_{t,i}(s, k)) = \min_{u_{t,i}(s, k) \in \Omega_{AP}} \left\{ [Obj(\mathbf{X}_t(s)) + \frac{1}{2}|U_{t,i}(s, k)|^2] + \alpha \cdot J_t[T(\mathbf{X}_t(s), u_{t,i}(s, k))] \right\} \quad (25)$$

其中, J_t 利用线性插值近似得到.

步骤 3.4. 令 $k = k + 1$, 进行群体寻优计算. 当 $k = C_{\max}$ 时, 近似认为对所有采样状态 $\mathbf{X}_t(s)$, $s = 1, 2, \dots, n$ 得到了最优的控制策略 $\hat{\mathbf{u}}_t^*(s)$, 记录为 $\hat{\mathbf{u}}_t(s)$, 并更新最优控制策略网络 $\hat{\mathbf{u}}_t = [\hat{u}_t(1) \ \hat{u}_t(2) \ \dots \ \hat{u}_t(n)]$.

在阶段 t , 依次施加最优控制序列 $\hat{\mathbf{u}}_{t-1}^*(s) = \{\hat{u}_0^*(s), \hat{u}_1^*(s), \dots, \hat{u}_{t-1}^*(s)\}$ 来获得最优代价, 将其存储并更新最优控制策略网络 $\hat{\mathbf{u}}$. 然后, 对 $\hat{J}_t(\mathbf{X}_t(s))$, $s = 1, 2, \dots, n$ 利用线性插值近似得到所有状态 \mathbf{X} 的代价函数值, 从而实现对整个最优代价网络 $\tilde{J}_t(\mathbf{X})$ 的更新 (代价网络学习).

步骤 4. 令 $t = t + 1$, 重复步骤 2 和步骤 3, 直至 $|\hat{J}_{t+1}(\mathbf{X}) - \hat{J}_t(\mathbf{X})| < \varepsilon$, 算法终止, 其中 ε 为一个任意的正实数, 于是有 $\hat{J}_t(\mathbf{X}) \rightarrow \tilde{J}(\mathbf{X})$; 或者当满足条件 $t = T_{\max}$ 时, 算法终止. 输出最优控制序列 $\hat{\mathbf{u}}_t^* = [\hat{u}_t^*(1) \ \hat{u}_t^*(2) \ \dots \ \hat{u}_t^*(n)]^T$.

步骤 5. 对所得到的控制序列 $\hat{\mathbf{u}}_t^*$ 进行数据拟合与分析, 得到相应参数的最优动态结构, 并作为 PSO 算法相关参数优化设置的依据.

3 数值实验与分析

3.1 基于 SADP 的随机 PSO 系统参数动态优化仿真

考虑由式 (16) 和 (17) 描述的随机 PSO 系统, 其中的 ϑ 为一个在区间 (0, 1) 上服从均匀分布的随机变量. 令 $\omega = 0.5$, 系统具体描述为

$$\mathbf{X}_{t+1} = \begin{bmatrix} 1 & 0.5 \\ 0 & 0.5 \end{bmatrix} \mathbf{X}_t + \vartheta \cdot \begin{bmatrix} 1 \\ 1 \end{bmatrix} \mathbf{u}_t = T(\mathbf{X}_t, \mathbf{u}_t, \vartheta) \quad (26)$$

根据系统的收敛条件 $0 < \bar{\varphi} < 2(1 + \omega)$, $0 < \omega < 1$, 可以得到 $\bar{\varphi}(\omega = 0.5) < 3$; 相应地, 有效控制策略的范围及边界为: $\Omega_{AP} = \sup(|\mathbf{u}_t|) = 3|x_t|$.

在系统的性能指标定义中, 选用表 1 中的几种基准函数作为目标函数 Obj .

实验基于表 1 中的 6 种基准函数优化, 对一维 PSO 系统进行最优控制求解. 设定最优目标 $\tilde{p} = 0$, 且 $y_t = x_t - \tilde{p} = x_t$. SADP 的参数设置如下: 用于离散动态规划近似求解的群体规模为 $N = 50$; 微粒群搜索的迭代周期为 C_{\max} ; 惯性权重 ω 的范围设置为 $[\omega_{\min}, \omega_{\max}] = [0.2, 0.9]$, 并且在 100 次迭代过程中采用线性减少的策略; 加速因子 $c_1 =$

$c_2 = 2.0$. 系统动态规划运行的总迭代次数 $TI = 1000$; 每一次动态规划的终止条件为达到最大阶段数 $t = T_{\max} = 50$. 动态规划的每一次迭代中采样状态个数为 $n = 1$. 贝尔曼方程中的折扣因子 α 设置为 $\alpha = 0.99$.

表 1 仿真采用的基准函数
Table 1 Benchmarks for simulations

函数	函数的数学表达	搜索区间
Sphere	$f_1 = \sum_{i=1}^n x_i^2$	$[-2, 2]$
Griewank	$f_2 = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos(\frac{x_i}{\sqrt{i}} x_i) + 1$	$[-2, 2]$
Rastrigin	$f_3 = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$	$[-2, 2]$
Ackley	$f_4 = -20 \exp(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}) - \exp(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)) + 20 + e$	$[-2, 2]$
Michalewicz	$f_5 = -\sum_{i=1}^n \sin(x_i) \sin^{2k}(\frac{ix_i^2}{\pi}), k = 10$	$[-\pi, \pi]$
Weierstrass	$f_6 = \sum_{i=1}^n (\sum_{k=0}^k \max[a^k \cos(2\pi b^k \times (x_i + 0.5))]) - n \sum_{k=0}^k \max[a^k \cos(2\pi b^k)]$ $a = 0.5, b = 0.3, k_{\max} = 20$	$[-2, 2]$

实验基于 Matlab 7 仿真平台. 采用以上参数设置, 分别得到基于 Sphere 函数 (f_1), Griewank 函数 (f_2), Rastrigin 函数 (f_3), Ackley 函数 (f_4), Michalewicz 函数 (f_5) 和 Weierstrass 函数 (f_6) 的 PSO 系统最优控制结果, 典型的仿真结果如图 1~6 所示. 图示结果包括代价值曲线、最优控制策略轨迹、最优控制策略拟合曲线和系统状态轨迹等.

根据系统反馈控制模式 $u_t = -\varphi_t x_t$, 应用近似线性拟合策略, 分别得到针对各函数优化的 PSO 系统最优控制策略的拟合曲线, 近似函数表达为

$$\begin{cases} u^*(f_1) \approx -1.71x, & u^*(f_2) \approx -1.78x \\ u^*(f_3) \approx -1.45x, & u^*(f_4) \approx -1.38x \\ u^*(f_5) \approx -1.61x, & u^*(f_6) \approx -1.56x \end{cases} \quad (27)$$

方程组 (27) 给出的近似拟合结果是静态的, 可用于 PSO 算法相应参数的初始化设置, 并且该参数在整个系统进化与寻优过程中保持不变. 从图 1~6 中所示的曲线可以看到, 线性拟合结果与实际得到的离散控制结果之间存在着较大的误差. 为了能够得到较为准确的最优控制轨迹, 则需要使用更高阶的非线性关系进行近似拟合. 这里, 以三次拟合为例来说明非线性拟合方法, 所得最优控制拟合结果为:

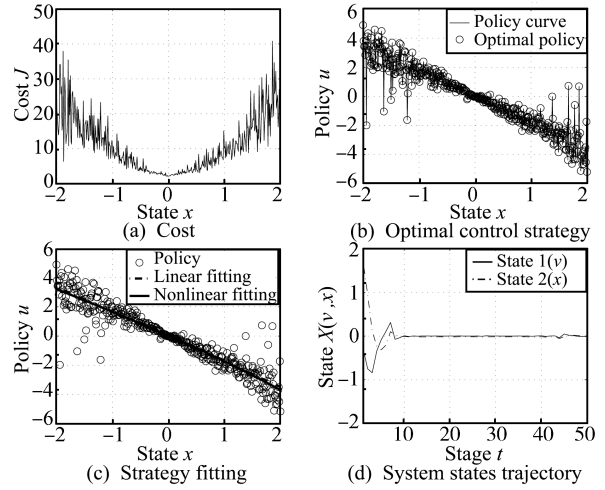


图 1 基于 Sphere 函数优化的 PSO 系统仿真结果
Fig. 1 Simulation results of Sphere function

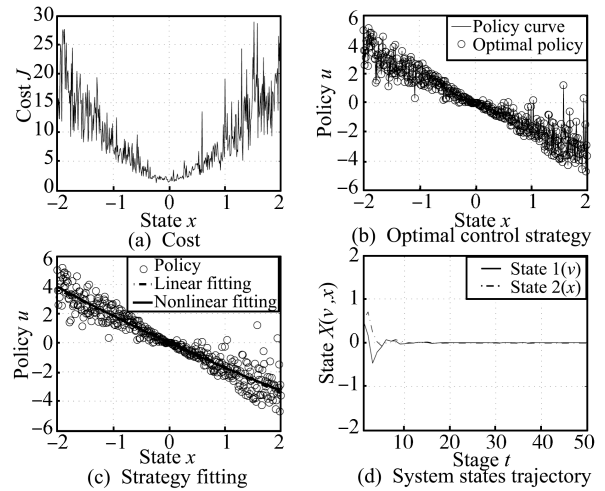


图 2 基于 Griewank 函数优化的 PSO 系统仿真结果
Fig. 2 Simulation results of Griewank function

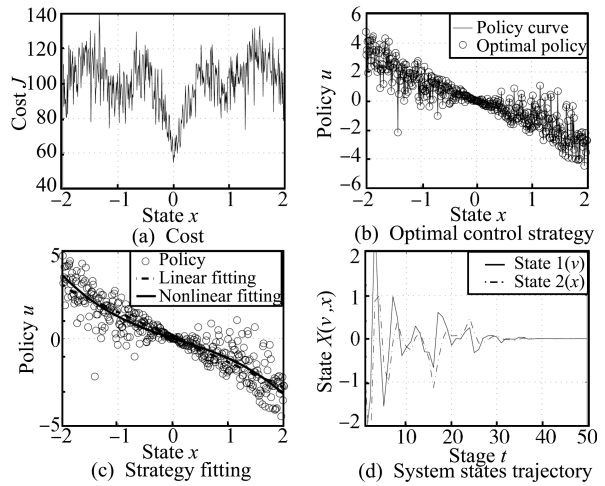


图 3 基于 Rastrigin 函数优化的 PSO 系统仿真结果
Fig. 3 Simulation results of Rastrigin function

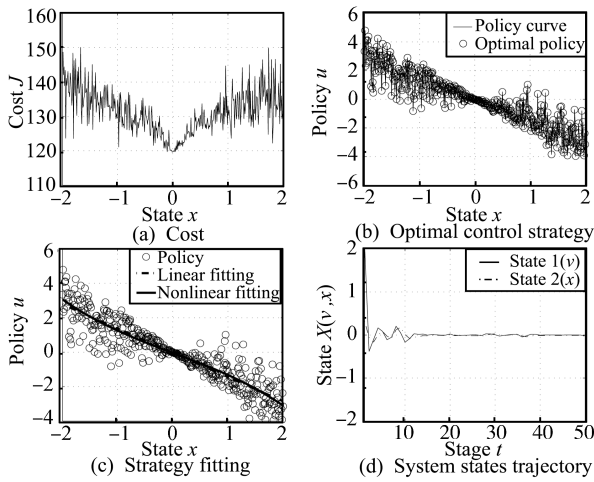


图 4 基于 Ackley 函数优化的 PSO 系统仿真结果
Fig. 4 Simulation results of Ackley function

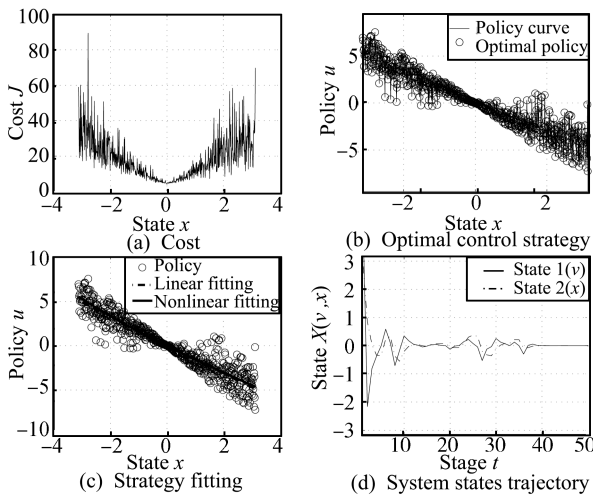


图 5 基于 Michalewicz 函数优化的 PSO 系统仿真结果

Fig. 5 Simulation results of Michalewicz function

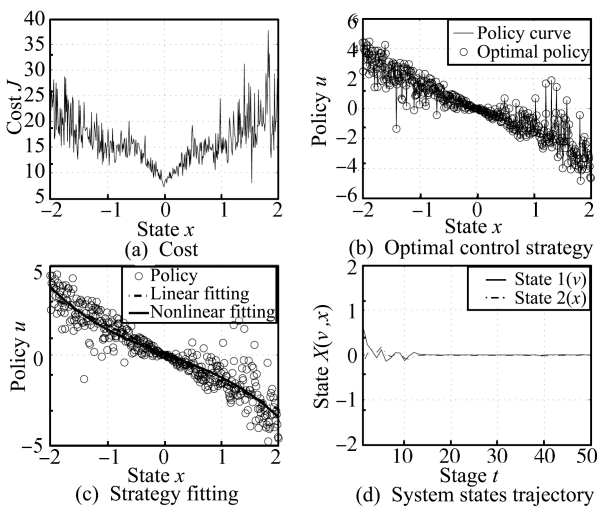


图 6 基于 Weierstrass 函数优化的 PSO 系统仿真结果
Fig. 6 Simulation results of Weierstrass function

$$\begin{cases} u^*(f_1) \approx -0.03x^3 + 0.04x^2 - 1.67x - 0.04 \\ u^*(f_2) \approx -0.04x^2 - 1.75x \\ u^*(f_3) \approx -0.01x^3 + 0.04x^2 - 1.22x \\ u^*(f_4) \approx -0.2x^3 + 0.02x^2 - 0.9x - 0.22 \\ u^*(f_5) \approx -0.03x^3 - 0.01x^2 - 1.44x - 0.07 \\ u^*(f_6) \approx -0.02x^3 - 0.06x^2 - 1.51x \end{cases} \quad (28)$$

根据 $\mathbf{u}_t = -\varphi_t \mathbf{x}_t$, 不能直接利用式 (28) 中的结果求参数 φ_t . 但可针对所选择的不同函数优化问题, 利用相应的关系式, 计算各离散时刻每个确定状态 x 对应的最优控制策略 \mathbf{u}_t^* , 然后根据所得数据求解 $\mathbf{u}_t = -\varphi_t \mathbf{x}_t$ 的系数, 以此来动态调整 φ_t 值. 同理, 可利用更高阶的非线性关系进行参数 φ_t 的动态优化规律拟合, 所得结果也更接近实验所得数据. 还可以直接利用实验得到的离散最优控制数据求得相应采样时刻的 φ_t 值.

3.2 比较分析

本实验目的是测试文中所提方法的有效性, 从算法最优解质量来检验采用不同加速因子策略的 PSO 优化性能. 对实验得到的最优控制结果, 分别采用线性拟合、非线性 (三次曲线) 拟合以及对离散最优控制数据线性插值的方式得到参数的近似动态规律, 分别基于这些结果设置并运行 PSO 算法.

在函数优化仿真实验中, 选用表 1 中的测试函数, 考察基于不同加速因子策略的 PSO 算法在一定的迭代周期之后找到的最优解的平均值. 前面的讨论基于恒定的惯性权重 ω , 为了便于比较, 本实验中 PSO 算法也均采用常数惯性权重, 即 $\omega = 0.5$. 采用非对称的初始化方式, 群体规模统一设置为 $N = 50$. 分别对每种 PSO 计算模式进行 50 次实验, 统计平均最优值, 并与 PSO-TVAC 在不同的加速因子设置下 (PSO-TVAC¹: $c_1 = c_2 = 2.0$ 和 PSO-TVAC²: $c_1 = 2.0 \rightarrow 0.25, c_2 = 0.25 \rightarrow 2.0$)^[4] 得到的平均最优结果进行比较. 各算法在达到最大迭代周期时, 得到如表 2 中的仿真结果.

从表 2 的结果可以看出, 基于表 1 的 6 种测试函数优化问题, 采用本文方法得到的加速因子动态规律的 PSO 算法总体上具有与一种动态加速因子微粒群算法 (PSO-TVAC²) 相当的性能, 该结果是基于对离散最优控制策略进行线性插值所得优化参数而取得的. 本文方法除了对 Rastrigin 函数得到比 PSO-TVAC 较差的结果外, 对其他函数均取得了比 PSO-TVAC 更好或相当的优化效果. 不过, 基于通过线性拟合和三次拟合得到的加速因子动态规律而得到的优化结果并不太理想. 从最优控制策略拟合曲线可以看出, 利用线性拟合甚至非线性拟合得

表 2 仿真结果比较 (平均最优值)
Table 2 Comparison of average results

函数	维数 (G_{\max})	寻优范围	PSO-TVAC ¹	PSO-TVAC ²	线性拟合	三次拟合	离散最优插值
f_1	30 (2 000)	[-100, 100]	9.383E-24	1.755E-25	7.993E-22	2.524E-23	1.645E-25
f_2	30 (2 000)	[-600, 600]	0.142	2.651E-02	0.107	7.349E-02	3.196E-02
f_3	30 (2 000)	[-10, 10]	32.79	2.408	36.55	12.39	5.617
f_4	30 (2 000)	[-100, 100]	6.883E-09	2.147E-12	5.153E-07	3.452E-09	9.679E-15
f_5	30 (2 000)	$[-\pi, \pi]$	-17.06	-24.41	-19.28	-25.21	-25.74
f_6	30 (2 000)	[-2.0, 2.0]	3.481E-08	3.146E-10	7.526E-09	1.567E-10	4.184E-11

到的近似规律与实际的最优动态数据之间存在着一定的误差; 拟合误差较大的函数, 用不同设置得到的优化效果的差距通常也较明显, 这与表 2 中的结果基本吻合。

从表 2 的结果还可看出, 采用恒定加速因子 (PSO-TVAC¹ 和线性拟合) 所得结果相对较差. 需要说明的是, 本实验采用的算法对除 Sphere, Ackley 和 Weierstrass 函数外的优化结果都不够理想, 特别是对 Rastrigin 函数, 得到的结果均较差, 这可能与采用固定惯性权重 $\omega = 0.5$ 有关. 当然, 本文工作主要是从系统最优控制的新视角对 PSO 参数的动态优化设置问题进行探讨, 作出近似假设处理, 对计算结果产生了一定程度的影响。

4 结语

本文将群体智能引入近似动态规划模式的设计, 提出一种新型的近似动态规划模式 SADP, 并从系统最优控制的角度出发, 研究了微粒群参数的动态优化问题. 在 SADP 设计以及 PSO 参数的动态优化计算过程中, 采取了适当的近似简化处理, 致使所得结果与其精确解产生了偏差, 实验结果也表明这会对函数优化问题求解造成一些不利影响. 不过, 本文以优化参数设置为目的, 微粒群参数本身对精确度要求并不太高, 计算复杂度更显重要, 而采取的近似处理有效降低了计算复杂度. 还需要强调的是, 本工作的初衷是从最优控制的视角对 PSO 参数的动态优化进行探讨, 是一种新的尝试与方法探索. 因此, 采取近似简化处理应当是合理的, 由此产生的计算误差不影响对方法的探讨与研究. 当然, 本文除需要进一步对所提方法进行更为细致完善的仿真验证工作之外, 还缺乏对算法的收敛性或逼近特性等方面进行严格的证明, 这也是今后需要努力的方向。

本文对微粒群近似动态优化的相关研究均基于单个微粒在一维空间飞行的假设前提. 这时面临的问题是, 绝大多数待求解问题在解空间中各个维度间并不是独立的, 这样很可能对求解性能造成很大

的影响, 这也可能是求解效果不太理想的潜在原因. 因此, 在对实际问题求解过程中, 需要首先设计相应的算法将问题进行解耦, 然后在每个独立的维度空间再应用本文提出的方法, 相应的解耦算法的设计与实现将是今后的研究方向. 此外, 本文所作假设在某种程度上偏离了微粒群启发式随机优化和群体协作等内在的优秀特性. 因此, 从微粒群优化算法的原始本意出发, 从最接近微粒群系统本质特性着眼进行多维、并行、分布式计算的近似动态优化研究, 对多目标优化的微粒群系统甚至对其他各类群体智能系统的动态优化研究等, 将是一个很有意义的课题, 也是下一步要做的工作. 尽管如此, 本文工作是对微粒群系统进行近似动态优化的有益尝试, 是对该课题深入研究的一种初步的探索性工作, 作者希望能对相关研究领域的研究者有所帮助与启示。

References

- 1 Kennedy J, Eberhart R. Particle swarm optimization. In: Proceedings of the International Conference on Neural Networks. Perth, Australia: IEEE, 1995. 1942-1948
- 2 Eberhart R, Shi Y. Particle swarm optimization: developments, applications and resources. In: Proceedings of the Congress on Evolutionary Computation. Seoul, Korea: IEEE, 2001. 81-86
- 3 Zhan Z H, Zhang J, Li Y, Chung H S. Adaptive particle swarm optimization. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 2009, **39**(6): 1362-1381
- 4 Asanga R, Saman K H, Harry C W. Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients. *IEEE Transactions on Evolutionary Computation*, 2004, **8**(3): 240-255
- 5 Rapaic M R, Kanovic Z. Time-varying PSO-convergence analysis, convergence-related parameterization and new parameter adjustment schemes. *Information Processing Letters*, 2009, **109**(11): 548-552
- 6 Janson S, Middendorf M. A hierarchical particle swarm optimizer and its adaptive variant. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 2005, **35**(6): 1272-1282
- 7 Hsieh S T, Sun T Y, Liu C C, Tsai S J. Efficient population utilization strategy for particle swarm optimizer. *IEEE*

- Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 2009, **39**(2): 444–456
- 8 Liu B, Wang L, Jin Y. An effective hybrid PSO-based algorithm for flow shop scheduling with limited buffers. *Computers and Operations Research*, 2008, **35**(9): 2791–2806
- 9 Lv Qiang, Liu Shi-Rong, Qiu Xue-Na. Design and realization of particle swarm optimization based on pheromone mechanism. *Acta Automatica Sinica*, 2009, **35**(11): 1410–1419
(吕强, 刘士荣, 邱雪娜. 基于信息素机制的粒子群优化算法的设计与实现. *自动化学报*, 2009, **35**(11): 1410–1419)
- 10 Zhang Yong, Gong Dun-Wei, Zhang Wan-Qiu. A simplex method based improved particle swarm optimization and analysis on its global convergence. *Acta Automatica Sinica*, 2009, **35**(3): 289–298
(张勇, 巩敦卫, 张婉秋. 一种基于单纯形法的改进微粒群优化算法及其收敛性分析. *自动化学报*, 2009, **35**(3): 289–298)
- 11 Coello C A, Pulido G T, Lechuga M S. Handling multiple objectives with particle swarm optimization. *IEEE Transactions on Evolutionary Computation*, 2004, **8**(3): 256–279
- 12 Vlachogiannis I J, Lee K Y. Economic load dispatch — a comparative study on heuristic optimization techniques with an improved coordinated aggregation-based PSO. *IEEE Transactions on Power Systems*, 2009, **24**(2): 991–1001
- 13 Ko C N, Chang Y P, Wu C J. A PSO method with nonlinear time-varying evolution for optimal design of harmonic filters. *IEEE Transactions on Power Systems*, 2009, **24**(1): 437–444
- 14 Zhang Z X, Lu C W. Application of the improved particle swarm optimizer to vehicle routing and scheduling problems. In: Proceedings of the IEEE International Conference on Grey Systems and Intelligent Services. Nanjing, China: IEEE, 2007. 1150–1152
- 15 Huang Xiao-Ling, Chai Tian-You. Particle swarm optimization for raw material purchasing plan in large scale ore dressing plant. *Acta Automatica Sinica*, 2009, **35**(5): 632–636
(黄肖玲, 柴天佑. 粒子群优化算法在大型选矿企业原料采购计划中的应用. *自动化学报*, 2009, **35**(5): 632–636)
- 16 Shi Y, Eberhart R. Empirical study of particle swarm optimization. In: Proceedings of the Congress on Evolutionary Computation. Washington D. C., USA: IEEE, 1999. 1945–1950
- 17 Shi Y, Eberhart R. Fuzzy adaptive particle swarm optimization. In: Proceedings of the Congress on Evolutionary Computation. Washington D. C., USA: IEEE, 2001. 101–106
- 18 Zhang Chang-Sheng, Sun Ji-Gui, Ouyang Dan-Tong. A self-adaptive discrete particle swarm optimization algorithm. *Acta Electronica Sinica*, 2009, **37**(2): 299–304
(张长胜, 孙吉贵, 欧阳彤彤. 一种自适应离散粒子群算法及其应用研究. *电子学报*, 2009, **37**(2): 299–304)
- 19 Han K, Zhao J, Xu Z, Qian J X. A closed-loop particle swarm optimizer for multivariable process controller design. *Journal of Zhejiang University, Science A*, 2008, **9**(8): 1050–1060
- 20 Suganthan P N. Particle swarm optimizer with neighborhood operator. In: Proceedings of the Congress on Evolutionary Computation. Washington D. C., USA: IEEE, 1999. 1958–1962
- 21 de Oca M A, Stutzle T, Birattari M, Dorigo M. Frankenstein's PSO: a composite particle swarm optimization algorithm. *IEEE Transactions on Evolutionary Computation*, 2009, **13**(5): 1120–1132
- 22 Leong W F, Yen G G. PSO-based multi-objective optimization with dynamic population size and adaptive local archives. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 2008, **38**(5): 1270–1293
- 23 Fan H. A modification to particle swarm optimization algorithm. *Engineering Computations*, 2002, **19**(8): 970–989
- 24 Gao Fang, Cui Gang, Wu Zhi-Bo, Yang Xiao-Zong. A novel multi-step position-selectable updating particle swarm optimization algorithm. *Acta Electronica Sinica*, 2009, **37**(3): 529–534
(高芳, 崔刚, 吴智博, 杨孝宗. 一种新型多步式位置可选择更新粒子群优化算法. *电子学报*, 2009, **37**(3): 529–534)
- 25 Kennedy J. The behavior of particle. In: Proceedings of the 7th International Conference on Evolutionary Programming. San Diego, USA: Springer, 1998. 581–589
- 26 Ozcan E, Mohan C K. Particle swarm optimization: surfing the waves. In: Proceedings of the Congress on Evolutionary Computation. Washington D. C., USA: IEEE, 1999. 1939–1944
- 27 Clerc M, Kennedy J. The particle swarm-explosion, stability, and convergence in a multidimensional complex space. *IEEE Transactions on Evolutionary Computation*, 2002, **6**(1): 58–73
- 28 Trelea I C. The particle swarm optimization algorithm: convergence analysis and parameter selection. *Information Processing Letters*, 2003, **85**(6): 317–325
- 29 van den Bergh F, Engelbrecht A P. A study of particle swarm optimization particle trajectories. *Information Sciences*, 2006, **176**(8): 937–971
- 30 Pan Feng, Chen Jie, Gan Ming-Gang, Cai Tao, Tu Xu-Yan. Model analysis of particle swarm optimizer. *Acta Automatica Sinica*, 2006, **32**(3): 368–377
(潘峰, 陈杰, 甘明刚, 蔡涛, 涂序彦. 粒子群优化算法模型分析. *自动化学报*, 2006, **32**(3): 368–377)
- 31 Pan Feng, Chen Jie, Xin Bin, Zhang Juan. Several characteristics analysis of particle swarm optimizer. *Acta Automatica Sinica*, 2009, **35**(7): 1010–1016
(潘峰, 陈杰, 辛斌, 张娟. 粒子群优化方法若干特性分析. *自动化学报*, 2009, **35**(7): 1010–1016)
- 32 Kadirkamanathan V, Selvarajah K, Fleming P J. Stability analysis of the particle dynamics in particle swarm optimizer. *IEEE Transactions on Evolutionary Computation*, 2006, **10**(3): 245–255
- 33 Bellman R E. *Dynamic Programming*. Princeton: Princeton University Press, 1957
- 34 Liu D, Jin N. ε -adaptive dynamic programming for discrete-time systems. In: Proceedings of the IEEE International Joint Conference on Neural Networks. Hong Kong, China: IEEE, 2008. 1417–1424
- 35 Powell W B. *Approximate Dynamic Programming: Solving the Curses of Dimensionality*. New York: Wiley-Interscience, 2007
- 36 Liu De-Rong. Approximate dynamic programming for self-learning control. *Acta Automatica Sinica*, 2005, **31**(1): 13–18
- 37 Zhao D B, Yi J Q, Liu D R. Particle swarm optimized adaptive dynamic programming. In: Proceedings of the IEEE International Symposium on Approximate Dynamic Programming and Reinforcement Learning. Honolulu, USA: IEEE, 2007. 32–37

- 38 Sitarz S. Ant algorithms and simulated annealing for multi-criteria dynamic programming. *Computers and Operations Research*, 2009, **36**(2): 433–441
- 39 Lu Y Z. An effective hybrid ADP-PSO strategy for optimization and its application to face recognition. In: Proceedings of the 3rd International Conference on Natural Computation. Washington D. C., USA: IEEE, 2007. 600–605
- 40 Jin Xin-Lei, Ma Long-Hua, Wu Tie-Jun, Qian Ji-Xin. Convergence analysis of the particle swarm optimization based on stochastic processes. *Acta Automatica Sinica*, 2007, **33**(12): 1263–1268
(金欣磊, 马龙华, 吴铁军, 钱积新. 基于随机过程的 PSO 收敛性分析. 自动化学报, 2007, **33**(12): 1263–1268)
- 41 Jiang M, Luo Y P, Yang S Y. Stochastic convergence analysis and parameter selection of the standard particle swarm optimization algorithm. *Information Processing Letters*, 2007, **102**(1): 8–16
- 42 Fernández M J, Garcia G E. The PSO family: deduction, stochastic analysis and comparison. *Swarm Intelligence*, 2009, **3**(4): 245–273
- 43 Chen G, Chen G R, Hsu S H. *Linear Stochastic Control Systems*. Boca Raton: CRC press, 1995



康琦 博士, 同济大学电子与信息工程学院控制科学与工程系讲师. 主要研究方向为计算智能, 智能控制和自适应动态规划. 本文通信作者.

E-mail: qkang@tongji.edu.cn

(**KANG Qi** Ph.D., lecturer in the Department of Control Science and Engineering, College of Electronics and Information Engineering, Tongji University. His research interest covers computational intelligence, intelligent control, and adaptive dynamic programming. Corresponding author of this paper.)

His research interest covers computational intelligence, intelligent control, and adaptive dynamic programming. Corresponding author of this paper.)



汪镭 同济大学电子与信息工程学院控制科学与工程系教授. 主要研究方向为智能自动化, 计算智能和系统工程.

E-mail: wanglei@tongji.edu.cn

(**WANG Lei** Professor in the Department of Control Science and Engineering, College of Electronics and Information Engineering, Tongji University. His research interest covers intelligent automation, computational intelligence, and system engineering.)



安静 同济大学电子与信息工程学院控制科学与工程系博士研究生. 主要研究方向为计算智能和智能信息处理.

E-mail: anjing_tj@163.com

(**AN Jing** Ph.D. candidate in the Department of Control Science and Engineering, College of Electronics and Information Engineering, Tongji University. Her research interest covers computational intelligence and intelligent information processing.)



吴启迪 同济大学电子与信息工程学院控制科学与工程系教授. 主要研究方向为控制理论与工程, 智能自动化, 计算智能, 复杂制造系统调度与优化, 系统工程和管理工程.

E-mail: wuqidi@moe.edu.cn

(**WU Qi-Di** Professor in the Department of Control Science and Engineering, College of Electronics and Information Engineering, Tongji University. Her research interest covers control theory and engineering, intelligent automation, computational intelligence, complex systems scheduling and optimization, system engineering, and management engineering.)