基于状态 - 动作图测地高斯基的策略迭代强化学习

程玉虎 [1] 冯涣婷 [1] 王雪松 [1]

摘 要 在策略迭代强化学习中, 基函数构造是影响动作值函数逼近精度的一个重要因素. 为了给动作值函数逼近提供合适的基函数, 提出一种基于状态 - 动作图测地高斯基的策略迭代强化学习方法. 首先, 根据离策略方法建立马尔可夫决策过程的状态 - 动作图论描述; 然后, 在状态 - 动作图上定义测地高斯核函数, 利用基于近似线性相关的核稀疏方法自动选择测地高斯核的中心; 最后, 在策略评估阶段利用基于状态 - 动作图的测地高斯核逼近动作值函数, 并基于估计的值函数进行策略改进. 10×10 格子世界的仿真结果表明, 与基于状态图普通高斯基和测地高斯基的策略迭代强化学习方法相比, 本文所提方法能以较少的基函数、高精度地逼近具有光滑且不连续特性的动作值函数, 从而有效地获得最优策略.

关键词 状态 - 动作图, 测地高斯核, 基函数, 策略迭代, 强化学习

DOI 10.3724/SP.J.1004.2011.00044

Policy Iteration Reinforcement Learning Based on Geodesic Gaussian Basis Defined on State-action Graph

CHENG Yu-Hu[1] FENG Huan-Ting[1] WANG Xue-Song[1]

Abstract For policy iteration reinforcement learning methods, the construction of basis functions is an important factor of influencing the accuracy of action-value function approximation. In order to construct appropriate basis functions for the action-value function approximation, a policy iteration reinforcement learning method based on geodesic Gaussian basis defined on state-action graph is proposed. At first, a state-action graph for a Markov decision process is constructed according to an off-policy method. Secondly, geodesic Gaussian kernel functions are defined on the state-action graph and a kernel sparsification approach based on approximate linear dependency is used to automatically select centers of the geodesic Gaussian kernels. At last, the geodesic Gaussian kernels based on the state-action graph is used to approximate the action-value function during the process of policy evaluation, and then the policy is improved based on the estimated action-value function. Simulation results concerning a 10×10 grid-world illustrate that the proposed method can accurately approximate the action-value function having smoothness and discontinuity properties with less basis functions as compared with the policy iteration reinforcement learning methods based on either ordinary Gaussian basis or geodesic Gaussian basis defined on a state graph, which is helpful for obtaining an optimal policy effectively.

Key words State-action graph, geodesic Gaussian kernel, basis function, policy iteration, reinforcement learning

强化学习作为一类求解序贯优化决策问题的有效方法, 已在自动控制、运筹学和计算科学等领域得到了广泛应用[1]. 强化学习以马尔可夫决策过程 (Markov decision process, MDP) 为基础, 强调在与未知环境的交互中学习, 以极大化未来的累积回报为学习目标来获得最优或者次优行为策略. 很多强化学习方法都是通过计算值函数来寻找最优策略的[2]. 对于较大规模的离散或者连续状态空间问题, 值函数的准确表示需要存储每个状态或者状态 - 动作对的值, 从而导致维数灾难. 为了克服维数灾难, 函数逼近成为一种表示值函数的有效方法[3−4]. 根据函数逼近器的特性, 强化学习方法可以分为两类. 一类是基于非线性结构的值函数逼近, 例如基于多层感知机的强化学习, 这种方法虽然已在电梯控制、车间调度、城市交通信号优化控制等领域有了成功的应用[5], 但是, 由于其经验结果缺乏严格的理论分析, 且多层感知机的结构需要人工选择, 从而制约了强化学习的进一步应用[6]. 另一类是基于线性结构的值函数逼近, 例如最小二乘策略迭代方法[7]. 虽然这种方法的泛化能力没有神经网络强, 但是, 从理论分析、程序调试和工程特点来看, 这种方法简单且易于实施.

当使用线性函数逼近结构来学习值函数时, 基函数的选择对智能体学习策略的能力有很大的影响. 目前, 常用的基函数主要有傅里叶级数[8]、扩散小波[9] 和高斯核函数[10] 等. 傅里叶级数和高斯核

收稿日期 2010-07-05 录用日期 2010-10-13
Manuscript received July 5, 2010; accepted October 13, 2010
国家自然科学基金 (60804022, 60974050, 61072094), 教育部新世纪优秀人才支持计划 (NCET-08-0836), 霍英东教育基金会青年教师基金 (121066), 江苏省自然科学基金 (BK2008126) 资助
Supported by National Natural Science Foundation of China (60804022, 60974050, 61072094), Program for New Century Excellent Talents in University (NCET-08-0836), Fok Ying-Tung Education Foundation for Young Teachers (121066), and Natural Science Foundation of Jiangsu Province (BK2008126)
1. 中国矿业大学信息与电气工程学院 徐州 221116
1. School of Information and Electrical Engineering, China University of Mining and Technology, Xuzhou 221116

函数具有光滑特性, 适用于逼近光滑且连续的函数. 扩散小波是小波在图论中的推广, 可以构造多尺度基函数, 适用于表示具有不同光滑程度且不连续的函数. 在强化学习任务中, 很多值函数都具有光滑且不连续特性, 例如: 在迷宫问题中, 一方面由于智能体不能穿越障碍物, 因此, 在障碍物附近的值函数是高度不连续的; 另一方面, 两个相邻可达状态对应的值相似, 因此, 值函数应是光滑的. 如果使用傅里叶级数和高斯核函数作为基函数来逼近值函数, 那么, 在不连续附近很可能会产生不理想的结果, 从而影响算法的整体性能. 扩散小波可以解决上述问题, 但是, 扩散小波是一个过完备基, 要从过完备基中选择一个合适的子集作为基函数比较困难. 因此, 为了能够更好地逼近光滑且不连续的值函数, Sugiyama 等提出了一种新的基函数构造方法 — 测地高斯核 (Geodesic Gaussian kernel, GGK)[10], 通过在 MDP 状态图上定义测地高斯核来构造基函数.

策略迭代强化学习方法需要逼近动作值函数来学习策略, 而 Sugiyama 构造的 GGK 基函数是基于状态空间的, 因此, 需要把状态空间的 GGK 扩展到状态 - 动作空间. 一种最常用的方法就是把状态空间的基函数直接 "复制" 到整个动作空间, 这意味着在一个相同的状态上采取不同的动作其对应的基函数相同. 但是, 在同一状态上采取不同的动作通常会得到不同的回报, 那么, 在状态图上构造基函数就无法体现动作之间的差异性, 从而影响值函数的逼近性能. 为此, Osentoski 给出一种直接在状态 - 动作空间上构造原型基函数的方法[11]. 为了提高值函数的逼近质量, 根据状态 - 动作图的特点, 借鉴 Osentoski 的思想, 本文将基于状态图的 GGK 推广到状态 - 动作图上, 通过直接在状态 - 动作空间逼近值函数来学习最优策略, 提出一种基于状态 - 动作图测地高斯基的策略迭代强化学习方法. 首先, 构造 MDP 的状态 - 动作图, 在状态 - 动作图上定义 GGK, 并使用基于近似线性相关的核稀疏方法自动确定 GGK 的中心; 然后, 在策略评估中利用状态 - 动作图上的 GGK 逼近动作值函数, 并根据逼近的动作值函数进行策略改进.

1　策略迭代强化学习

1.1　马尔可夫决策问题

在强化学习中, 往往把问题建模为马尔可夫决策过程[2]. 考虑离散时间 MDP: $\{S, A, P, r, \gamma\}$, 其中, S 为有限状态空间, A 为有限动作空间, $P(\boldsymbol{s}, a, \boldsymbol{s}') \in [0,1]$ 为状态转移概率, 即在状态 $\boldsymbol{s}$ 下选择动作 a 后, 转移到状态 $\boldsymbol{s}'$ 的概率, $r(\boldsymbol{s}, a, \boldsymbol{s}') : S \times A \times S \to \mathbf{R}$ 是从状态 $\boldsymbol{s}$ 执行动作 a 转移到状态 $\boldsymbol{s}'$ 后得到的立即回报, $\gamma \in [0,1]$ 为未来长期回报的折扣因子. 状态 - 动作对 $(\boldsymbol{s}, a)$ 的期望回报 $R(\boldsymbol{s}, a)$ 为

$$R(\boldsymbol{s}, a) = \sum_{\boldsymbol{s}' \in S} P(\boldsymbol{s}, a, \boldsymbol{s}') r(\boldsymbol{s}, a, \boldsymbol{s}') \tag{1}$$

设 $\pi(\boldsymbol{s}) : S \to A$ 为从状态到动作的映射, 表示一个确定型策略. 在一个任意策略 π 下的动作值函数为 $Q^{\pi}(\boldsymbol{s}, a) : S \times A \to \mathbf{R}$, 表示智能体在状态 $\boldsymbol{s}$ 选择动作 a 后, 按照策略 π 执行获得的期望长期折扣总回报. 根据贝尔曼公式[2], $Q^{\pi}(\boldsymbol{s}, a)$ 可以表示为

$$Q^{\pi}(\boldsymbol{s}, a) = R(\boldsymbol{s}, a) + \gamma \sum_{\boldsymbol{s}' \in S} P(\boldsymbol{s}, a, \boldsymbol{s}') Q^{\pi}(\boldsymbol{s}', \pi(\boldsymbol{s}')) \tag{2}$$

强化学习的目标就是学习一个可以最大化长期累积折扣回报的最优行为策略 $\pi^* = \arg\max_{a \in A} Q^*(\boldsymbol{s}, a)$, $Q^*(\boldsymbol{s}, a)$ 为最优动作值函数:

$$Q^*(\boldsymbol{s}, a) = \max_{\pi} Q^{\pi}(\boldsymbol{s}, a) \tag{3}$$

1.2　最小二乘策略迭代

策略迭代 (Policy iteration, PI) 分为策略评估和策略改进两部分, 可表示为[2]

$$\pi_1 \xrightarrow{E} Q^{\pi_1} \xrightarrow{I} \pi_2 \xrightarrow{E} Q^{\pi_2} \xrightarrow{I} \pi_3 \xrightarrow{E} \cdots \xrightarrow{I} \pi \tag{4}$$

其中, π_1 为初始策略, E 表示策略评估, 即用于计算值函数 $Q^{\pi}(\boldsymbol{s}, a)$, I 表示策略改进, 即根据 $Q^{\pi}(\boldsymbol{s}, a)$ 更新策略. 虽然策略迭代可以保证找到最优策略, 但是, 由于强化学习中的 $R(\boldsymbol{s}, a)$ 和 $P(\boldsymbol{s}, a, \boldsymbol{s}')$ 模型未知, 且状态 - 动作空间很大, 因此不能直接获得最优策略 π^*.

为了解决这个问题, Lagoudakis 和 Parr 提出了最小二乘策略迭代 (Least squares policy iteration, LSPI) 方法[7], 使用线性结构模型逼近动作值函数 $Q^{\pi}(\boldsymbol{s}, a)$, 有

$$\hat{Q}^{\pi}(\boldsymbol{s}, a) = \sum_{i=1}^{d} w_i \varphi_i(\boldsymbol{s}, a) \tag{5}$$

其中, $\boldsymbol{w} = (w_1, w_2, \cdots, w_d)^{\mathrm{T}}$ 为模型参数, $\boldsymbol{\varphi}(\boldsymbol{s}, a) = (\varphi_1(\boldsymbol{s}, a), \varphi_2(\boldsymbol{s}, a), \cdots, \varphi_d(\boldsymbol{s}, a))^{\mathrm{T}}$ 为预先选定的基向量, d 为基向量的长度 (基函数的个数), 且 $d << |S| \times |A|$. LSPI 是一种逼近动作值函数的最小二乘时间差分学习方法, 一般有两种: 不动点法和贝尔曼残差最小化法. Lagoudakis 和 Parr 证明了最小二乘不动点逼近法优于贝尔曼残差最小化逼近法[7], 因此, 此处采用最小二乘不动点逼近法来学习参数 $\boldsymbol{w}$. 假设在任意策略 π 下收集到 N 个样本 $\{(\boldsymbol{s}_n, a_n, \boldsymbol{s}'_n, r_n)\}_{n=1}^{N}$, 则有

$$\boldsymbol{w} = \tilde{A}^{-1} \boldsymbol{b} \tag{6}$$

其中,

$$\begin{cases} \tilde{A} = \sum\limits_{n=1}^{N} \boldsymbol{\varphi}(\boldsymbol{s}_n, a_n)(\boldsymbol{\varphi}(\boldsymbol{s}_n, a_n) - \gamma\boldsymbol{\varphi}(\boldsymbol{s}'_n, \pi(\boldsymbol{s}'_n)))^{\mathrm{T}} \\ \boldsymbol{b} = \sum\limits_{n=1}^{N} \boldsymbol{\varphi}(\boldsymbol{s}_n, a_n) r_n \end{cases} \tag{7}$$

基于逼近的动作值函数 $\hat{Q}^{\pi}(\boldsymbol{s}, a)$, 对策略进行更新, 即:

$$\pi(\boldsymbol{s}, a) = \arg\max_{a \in A} \hat{Q}^{\pi}(\boldsymbol{s}, a) \tag{8}$$

2 基于状态 - 动作图测地高斯基的策略迭代

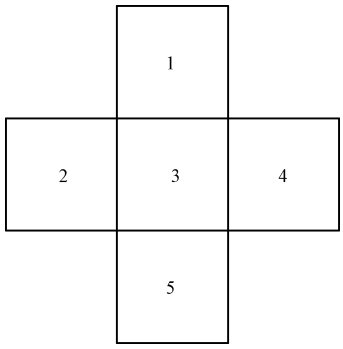
在 LSPI 方法中, 基函数的选择是一个开放性的设计问题. 高斯核函数具有模型选择简单、计算效率高以及算法易于实现的优点, 在值函数逼近中倾向于选择高斯核函数. 但是, 高斯核函数不能很好地逼近非连续函数. 因此, Sugiyama 等结合图结构, 提出了一种新的构造基函数方法 — 测地高斯核, 在 MDP 的状态图上定义高斯核, 然后将其扩展到状态 - 动作空间[10]. 当使用在状态空间定义的测地高斯核逼近动作值函数时, 存在一些不足: 1) 当将状态空间的基函数映射到状态 - 动作空间时, 这种映射不能够同时泛化状态和动作, 从而无法体现动作之间的相似性和差异性; 2) 在实际使用过程中, 通常由用户根据经验或反复实验试凑来确定测地高斯核的中心和宽度. 如果设置不当, 将极大影响算法的学习性能.

针对上述问题, 可以考虑通过直接在状态 - 动作图上定义测地高斯核来逼近动作值函数, 其中, 测地高斯核的中心采用基于近似线性相关的核稀疏方法自动确定.

2.1 MDP 的状态 - 动作图

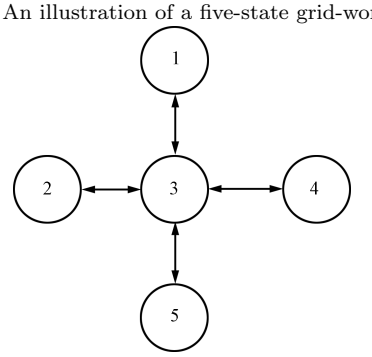
在强化学习中, 智能体使用任一随机策略对离散 MDP 环境进行采样, 基于获得的样本, 根据离策略方法[11] 来建立 MDP 环境的图论描述, 从而有效地表示状态 - 动作空间的拓扑结构.

设 $G_{\mathrm{sa}} = (V_{\mathrm{sa}}, E_{\mathrm{sa}}, W_{\mathrm{sa}})$ 为一个 MDP 的状态 - 动作图, 其中每一个顶点 $v \in V_{\mathrm{sa}}$ 对应一个状态 - 动作对 $(\boldsymbol{s}, a)$, 且 $\boldsymbol{s} \in S$ 和 $a \in A$. 若在一个状态 $\boldsymbol{s}$ 下采取动作 a, 转移到状态 $\boldsymbol{s}'$ 且有 $a' \in A(\boldsymbol{s}')$, 则这两个状态 - 动作对 $(\boldsymbol{s}, a)$ 和 $(\boldsymbol{s}', a')$ 之间的连接就构成了图的边 E_{sa}. W_{sa} 是一个 $|V_{\mathrm{sa}}| \times |V_{\mathrm{sa}}|$ 的权值矩阵, 根据转移概率或者两个顶点之间的距离给每条边赋予权值. 假设一个具有 5 个状态的简单格子世界如图 1 (a) 所示, 图 1 (b) 和图 1 (c) 分别给出了对应的状态和状态 - 动作拓扑结构图, 其中, U, D, L 和 R 分别表示智能体在格子世界中采取的 4 个动作, 即上、下、左、右. 对于一般的强化学习任务来说, 由环境生成的图是高度稀疏的, 即 $|E_{\mathrm{sa}}| << |V_{\mathrm{sa}}|\,(|V_{\mathrm{sa}}| - 1)/2$, 这一点对于求取任意两顶点间的最短路径非常有利, 能够极大降低计算复杂度.



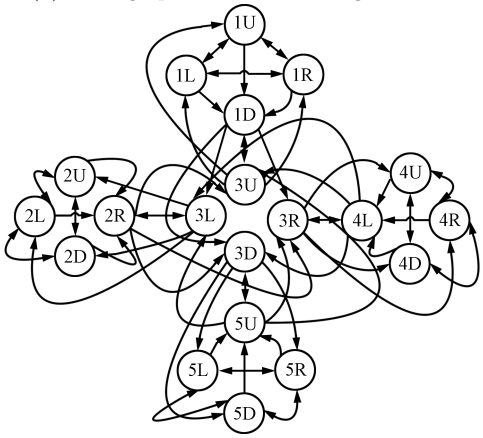
(a) 具有 5 个状态的格子世界

(a) An illustration of a five-state grid-world



(b) 格子世界 MDP 的状态空间图

(b) State graph of the five-state grid-world



(c) 格子世界 MDP 的状态 - 动作空间图

(c) State-action graph of the five-state grid-world

图 1 格子世界环境及其图论描述

Fig. 1 A grid-world and its graph description

2.2 状态 - 动作图上测地高斯核

定义在状态图上的测地高斯核函数为[10]

$$k(\boldsymbol{s}, \boldsymbol{s}_c) = \exp\left(-\frac{\mathrm{sp}(\boldsymbol{s}, \boldsymbol{s}_c)^2}{2\sigma^2}\right) \tag{9}$$

其中, σ 和 $\boldsymbol{s}_c$ 分别表示测地高斯核的宽度和中心,

$\mathrm{sp}(\boldsymbol{s},\boldsymbol{s}_c)$ 表示状态 $\boldsymbol{s}$ 和 $\boldsymbol{s}_c$ 之间的最短路径. 根据谱图理论可知, 图论中的最短路径是非线性流形上的测地线距离的离散逼近, 故式 (9) 称为测地高斯核.

在策略迭代强化学习中, 需要对式 (2) 定义的动作值函数进行逼近, 因此, 需要将在状态空间 S 上定义的测地高斯基扩展到状态 - 动作空间 $S\times A$. 常用的 "复制" 方法不能同时泛化状态和动作. 例如: 设图 1 (b) 和图 1 (c) 中相连两顶点间的距离为 1, 那么, 图 1 (b) 中状态 4 和 5 之间的测地距离为 2. 若按照 "复制" 方法将状态空间的基函数扩展到整个动作空间, 那么, 状态 - 动作对 (4,U), (4,D), (4,L) 和 (4,R) 与 (5,U) 之间的测地距离均为 2, 从而无法体现出相同状态下不同动作之间的差异性. 由图 1 (c) 表示的状态 - 动作图可以看出, 状态 - 动作对 (4,U), (4,D), (4,L) 和 (4,R) 与 (5,U) 之间的测地距离分别为 3, 3, 2 和 3. 这表明对于状态 4 来说, 选取动作 U, D 和 R 将产生相同的效果, 即撞墙, 而选取动作 L 将发生状态转移, 从而体现了在相同状态下不同动作之间的相似性和差异性. 为此, 状态 - 动作图上的测地高斯核定义为

$$k(\boldsymbol{x},\boldsymbol{x}_c)=\exp\left(-\frac{\mathrm{sp}(\boldsymbol{x},\boldsymbol{x}_c)^2}{2\sigma^2}\right) \tag{10}$$

其中, $\boldsymbol{x}$ 和 $\boldsymbol{x}_c$ 分别表示状态 - 动作对 $(\boldsymbol{s},a)$ 和 $(\boldsymbol{s}_c,a_c)$.

在求取测地高斯核时, 主要是计算图论中两顶点之间的最短路径. 计算最短路径的方法有很多, 如 Dijkstra 算法[12]、Bellman-Ford 算法、拓扑排序、最短路径快速算法 (Shortest path faster algorithm, SPFA)[13] 等. Dijkstra 算法是求最短路径的最基本和使用最广泛的算法, 它的优点是对图没有特殊要求、算法简单、计算直观, 不足之处是计算速度较慢, 算法的时间复杂度为 $\mathrm{O}(|V_{\mathrm{sa}}|^2)$. SPFA 是一种采用动态逼近优化的算法, 可以提高速度, 使其时间复杂性由 $\mathrm{O}(|V_{\mathrm{sa}}|^2)$ 降低为 $\mathrm{O}(|E_{\mathrm{sa}}|)$. 因此, 此处考虑采用 SPFA 方法计算最短路径.

2.3　基于状态 - 动作图测地高斯基的值函数逼近

使用基于状态 - 动作图的 GGK 逼近动作值函数 $Q^{\pi}(\boldsymbol{s},a)$, 则式 (5) 可以写为

$$\hat{Q}^{\pi}(\boldsymbol{x})=\sum_{i=1}^{d}w_i k(\boldsymbol{x},\boldsymbol{x}_i) \tag{11}$$

其中, $\boldsymbol{x}$ 和 $\boldsymbol{x}_i$ 分别表示状态 - 动作对 $(\boldsymbol{s},a)$ 和 $(\boldsymbol{s}_i,a_i)$. $(\boldsymbol{s}_i,a_i)$ 为选定的第 i 个测地高斯核中心. 相应地, 式 (7) 可以写为

$$\begin{cases}\tilde{A}=\sum\limits_{n=1}^{N}\boldsymbol{k}(\boldsymbol{x}_n)(\boldsymbol{k}(\boldsymbol{x}_n)-\gamma\boldsymbol{k}(\boldsymbol{x}'_n))^{\mathrm{T}}\\ \boldsymbol{b}=\sum\limits_{n=1}^{N}\boldsymbol{k}(\boldsymbol{x}_n)r_n\end{cases} \tag{12}$$

其中, $\boldsymbol{x}'_n$ 为状态 - 动作对 $(\boldsymbol{s}'_n,\pi(\boldsymbol{s}'_n))$, $\boldsymbol{k}(\boldsymbol{x}_n)=(k(\boldsymbol{x}_n,\boldsymbol{x}_1),k(\boldsymbol{x}_n,\boldsymbol{x}_2),\cdots,k(\boldsymbol{x}_n,\boldsymbol{x}_d))^{\mathrm{T}}$. 由式 (10) 可以看出, 在学习时需要设置测地高斯核的两个参数: 中心和宽度, 这两个参数对算法性能影响很大. 目前, 测地高斯核的参数都是用户根据经验或通过反复实验试凑事先给定的. 与此不同的是, 本文采用基于近似线性相关 (Approximate linear dependence, ALD) 的核稀疏方法[14] 来自动选择测地高斯核中心.

设 $X=\{\boldsymbol{x}_n\}_{n=1}^{N}$ 表示观察到的状态 - 动作对样本, $\boldsymbol{\varphi}(\boldsymbol{x}_n)$ 为状态 - 动作对 $\boldsymbol{x}_n$ 的一个特征映射, 特征向量集记为 $\Phi=\{\boldsymbol{\varphi}(\boldsymbol{x}_n)\}_{n=1}^{N}$, 根据基于 ALD 的核稀疏方法选择的 d 个测地高斯核中心记为数据字典 Ω. Ω 定义为样本集 X 的一个子集, 且 Ω 中元素对应的特征向量均是线性独立的. 初始时, 字典 Ω 为空集. 然后, 基于 ALD 分析, 通过测试每一个数据样本的特征向量来确定 Ω 中的元素. 设已经测试了样本集 X 中的 $n-1$ $(1<n\le N)$ 个样本, 且收集到的字典是 $n-1$ 个样本集的一个子集 $\Omega_{n-1}=\{\boldsymbol{x}_j\}_{j=1}^{m_{n-1}}$. 对于第 n 个样本 $\boldsymbol{x}_n$, 测试特征向量 $\boldsymbol{\varphi}(\boldsymbol{x}_n)$ 与字典 Ω_{n-1} 对应的特征向量的线性相关性: 在指定的精度内, 如果 $\boldsymbol{\varphi}(\boldsymbol{x}_n)$ 不能用字典 Ω_{n-1} 中所有元素对应的特征向量的线性组合来近似表示, 则可将样本 $\boldsymbol{x}_n$ 添加到字典 Ω_{n-1} 中. 如果样本 $\boldsymbol{x}_n$ 不能被添加到字典 Ω_{n-1} 中, 则其特征向量 $\boldsymbol{\varphi}(\boldsymbol{x}_n)$ 满足 ALD 条件[14]:

$$\delta_n=\min_{\boldsymbol{\alpha}}\left\|\sum_{j}^{m_{n-1}}\alpha_j\boldsymbol{\varphi}(\boldsymbol{x}_j)-\boldsymbol{\varphi}(\boldsymbol{x}_n)\right\|^2\le\mu \tag{13}$$

其中, $\boldsymbol{\alpha}=(\alpha_1,\alpha_2,\cdots,\alpha_{m_{n-1}})^{\mathrm{T}}$ 为 ALD 系数, μ 是一个给定的阈值参数, 用来平衡特征向量逼近的准确度和数据字典的稀疏水平.

下面分两步具体阐述如何根据 ALD 来选择测地高斯核中心, 即确定字典中的元素[14]. 第一步, 计算 δ_n, 将式 (13) 展开, 有

$$\begin{aligned}\delta_n=\min_{\boldsymbol{\alpha}}\Bigg\{&\sum_{j,l}^{m_{n-1}}\alpha_j\alpha_l\left\langle\boldsymbol{\varphi}(\boldsymbol{x}_j),\boldsymbol{\varphi}(\boldsymbol{x}_l)\right\rangle-\\ &2\sum_{j=1}^{m_{n-1}}\alpha_j\left\langle\boldsymbol{\varphi}(\boldsymbol{x}_j),\boldsymbol{\varphi}(\boldsymbol{x}_n)\right\rangle+\left\langle\boldsymbol{\varphi}(\boldsymbol{x}_n),\boldsymbol{\varphi}(\boldsymbol{x}_n)\right\rangle\Bigg\}\end{aligned} \tag{14}$$

其中, $\langle\cdot\rangle$ 表示两个特征向量 $\boldsymbol{\varphi}(\cdot)$ 的内积. 从式 (14)

可知, δ_n 的求解需要计算样本特征向量的内积, 而本文是使用测地高斯核来逼近 $Q^{\pi}(\boldsymbol{s}, a)$ 的, 故可根据核技巧将两个特征向量的内积计算转化为核函数, 即 $\langle \boldsymbol{\varphi}(\boldsymbol{x}_j), \boldsymbol{\varphi}(\boldsymbol{x}_n) \rangle = k(\boldsymbol{x}_j, \boldsymbol{x}_n)$, 于是式 (14) 可以写为

$$\delta_n = \min_{\boldsymbol{\alpha}} \{ \boldsymbol{\alpha}^{\mathrm{T}} K_{n-1} \boldsymbol{\alpha} - 2\boldsymbol{\alpha}^{\mathrm{T}} \boldsymbol{k}_{n-1}(\boldsymbol{x}_n) + k_{nn} \} \quad (15)$$

其中, $[K_{n-1}]_{j,l} = k(\boldsymbol{x}_j, \boldsymbol{x}_l)$, $\boldsymbol{x}_j$ 和 $\boldsymbol{x}_l$ 均是字典 Ω_{n-1} 中的元素, $\boldsymbol{k}_{n-1}(\boldsymbol{x}_n) = (k(\boldsymbol{x}_n, \boldsymbol{x}_1),\ k(\boldsymbol{x}_n,\ \boldsymbol{x}_2),\ \cdots,\ k(\boldsymbol{x}_n,\ \boldsymbol{x}_{m_{n-1}}))^{\mathrm{T}}$, $k_{nn} = k(\boldsymbol{x}_n, \boldsymbol{x}_n)$. 式 (15) 的最优解为

$$\begin{cases} \boldsymbol{\alpha}_n = K_{n-1}^{-1} \boldsymbol{k}_{n-1}(\boldsymbol{x}_n) \\ \delta_n = k_{nn} - \boldsymbol{k}_{n-1}^{\mathrm{T}}(\boldsymbol{x}_n) \boldsymbol{\alpha}_n \end{cases} \quad (16)$$

第二步, 基于式 (16) 的最优结果, 更新数据字典 Ω_{n-1}. 如果 $\delta_n \leq \mu$, 则字典 Ω_{n-1} 不发生改变, 即 $\Omega_n = \Omega_{n-1}$; 如果 $\delta_n > \mu$, 则使用样本 $\boldsymbol{x}_n$ 扩大当前的字典, 即 $\Omega_n = \Omega_{n-1} \cup \boldsymbol{x}_n$.

因此, 当测试完样本集 X 中的所有状态-动作对后, 得到的字典 Ω 即为待求的 d 个测地高斯核中心.

3 算法步骤

根据上述分析, 给出基于状态-动作图测地高斯基的策略迭代强化学习方法的计算步骤如下:

步骤 1. 根据强化学习任务, 建立环境的状态-动作图论描述;

步骤 2. 初始化参数, 包括折扣因子 γ、初始 ALD 系数 α_0、阈值参数 μ、测地高斯核宽度 σ 及权值学习精度 ε;

步骤 3. 根据基于 ALD 的核稀疏方法, 按照式 (16) 从环境的所有状态-动作对中确定测地高斯核的中心;

步骤 4. 根据任意策略 π 收集训练样本, 由式 (6) 计算权值 $\boldsymbol{w}$;

步骤 5. 若两个连续迭代步的权值变化小于指定的精度 ε, 即 $\|\boldsymbol{w}_{t+1} - \boldsymbol{w}_t\| < \varepsilon$, 则结束权值迭代学习过程, 转步骤 6, 否则, 转步骤 4;

步骤 6. 由式 (11) 计算估计的动作值函数 $\hat{Q}^{\pi}(\boldsymbol{s}, a)$, 并按照式 (8) 获得最优策略 π.

4 仿真研究

为验证本文所提算法的有效性, 选用如图 2 所示的 10×10 格子世界进行仿真研究. 图 2 中, 白色表示开阔地, 外部和内部的灰色阴影分别表示墙和障碍物, 格子中的数字表示状态, G 表示目标状态 100. 学习的目的是寻找一条从任一起始状态 (去除障碍物及目标状态, 共有 90 个起始状态) 到目标状态的最短路径. 强化学习智能体可选择的动作是 4 个 (上、下、左、右), 在碰到墙或障碍物时停在原地, 即后继状态仍为当前状态. 当智能体到达终点时, 得到的立即回报值为 +100, 否则都为 0.

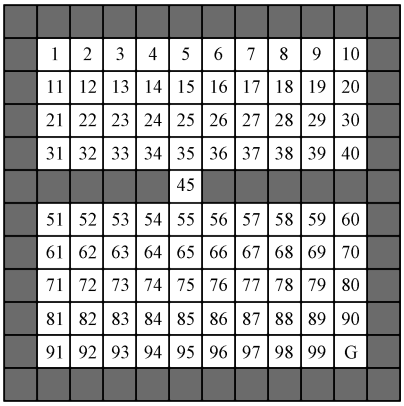


图 2 10×10 格子世界

Fig. 2 An illustration of a 10×10 grid-world

分别将以基于状态图的普通高斯核 (Ordinary Gaussian kernel, OGK)、基于状态图的测地高斯核和基于状态-动作图的测地高斯核作为基函数的 3 种类型策略迭代 (分别记为 S-OGK-PI、S-GGK-PI 和 SA-GGK-PI) 强化学习方法用于求解上述 10×10 格子世界问题. 仿真过程中, 随机收集 100 幕训练样本用于构造环境的状态图和状态-动作图, 其中每幕样本的长度均为 300, 且将状态图和状态-动作图中所有边的权值都设为 1. 仿真中, 定义在状态图上的 OGK 和 GGK 的宽度 σ 取为 10, 定义在状态-动作图上的 GGK 的宽度 σ 取为 0.1, $\gamma = 0.9$, $\varepsilon = 10^{-3}$, $\mu = 0.4$. S-OGK-PI 和 S-GGK-PI 强化学习方法均选取了 5 组不同的高斯核个数 (10, 30, 50, 70, 90), 高斯核的中心分别分布在目标点 G 和除墙以及障碍物以外的随机选取的顶点上. 值得指出的是, 由于使用 "复制" 方法进行基函数的扩展且此处动作的个数为 4 个, 因此, S-OGK-PI 和 S-GGK-PI 强化学习方法使用的基函数的个数分别为 (40, 120, 200, 280, 360).

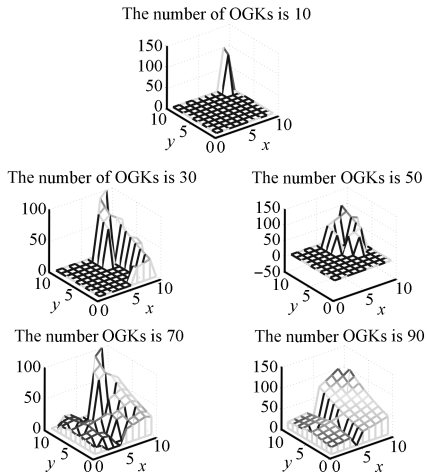
强化学习算法的性能由值函数的逼近和学到的最优策略来衡量, 图 3 和图 4 分别给出了相应的仿真结果. 图 3 为三种强化学习方法估计的值函数, 可以看出, 在核函数个数分别取 10、30、50 和 70 时, 利用状态图上的 OGK 和 GGK 估计的值函数很不光滑、波动大, 且在障碍物附近不具有非连续性, 而 SA-GGK-PI 仅使用 11 个状态-动作图 GGK 基函数, 不仅能够使其估计的值函数非常光滑、无波动, 而且在障碍物附近具有非连续性, 符合真实值函数的特性. 这是因为, 一方面, 根据 ALD 分析方法自动确定的 11 个状态-动作图 GGK 中心

((1, 上)、(7, 下)、(45, 下)、(60, 下)、(71, 下)、(90, 下)、(99, 右)、(100, 上)、(100, 下)、(100, 左)、(100, 右)) 对应的特征向量均是线性独立的, 从而可以有效地表示所有状态. 另一方面, 相比于通过复制状态基函数来构造状态 - 动作基函数的方法, 直接在状态 - 动作图上构造基函数可以呈现出状态 - 动作空间的光滑性, 从而能够更好地估计值函数. 由图 3 (a) 和图 3 (b) 可以看出, 随着核函数个数的增多, 得以有效表示的状态随之增多, 进而值函数的估计值也越来越准确.

图 4 中的箭头表示三种策略迭代强化学习方法通过学习得到的某个最优策略, 可以看出, 当利用 S-OGK-PI 和 S-GGK-PI 强化学习方法学习时, 在核函数个数取为 10、30、50 和 70 的情况下, 智能体虽然可能从某些状态出发到达目标点, 但是位于墙和障碍物附近状态处的智能体会试图采取穿越墙或者障碍物的动作, 这是由于状态图不能体现动作之间的差异性, 从而导致产生错误动作. 当核函数个数取为 90 时, 位于墙和障碍物附近状态处的智能体能够采取合适的动作, 表明 S-OGK-PI 和 S-GGK-PI 强化学习方法此次学到的策略能使智能体从任一起始状态找到一条最短路径到达目标状态.

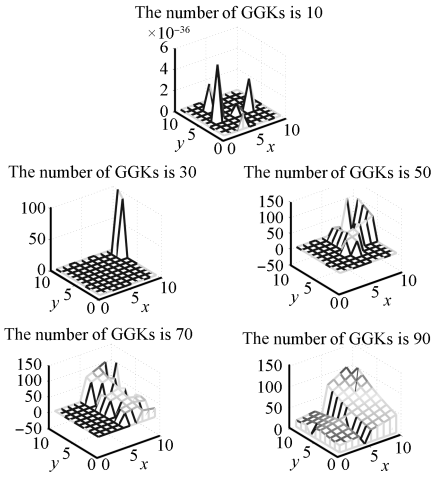
策略迭代强化学习方法得到的最优策略不唯一, 一般具有随机性, 图 4 只是给出了某一次运行中得到的最优策略. 为了评估策略的稳定性, 让这三种强化学习方法分别重复运行 20 次, 在每一次运行中智能体都随机收集 10 000 个训练样本来学习策略, 然后基于学到的策略, 测试智能体能否从任一起始状态在最小步数内到达目标点, 此处最小步数取为 30.

表 1 统计了在 20 次运行中, 智能体能够在最小步数内到达目标点的起始状态数的百分比. 以基于状态图普通高斯基的策略迭代强化学习方法为例,



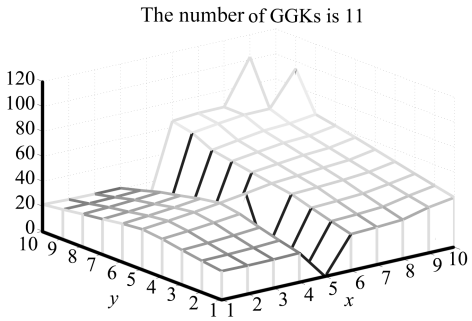
(a) 基于状态图普通高斯基的策略迭代强化学习

(a) Policy iteration reinforcement learning based on ordinary Gaussian basis defined on a state graph



(b) 基于状态图测地高斯基的策略迭代强化学习

(b) Policy iteration reinforcement learning based on geodesic Gaussian basis defined on a state graph

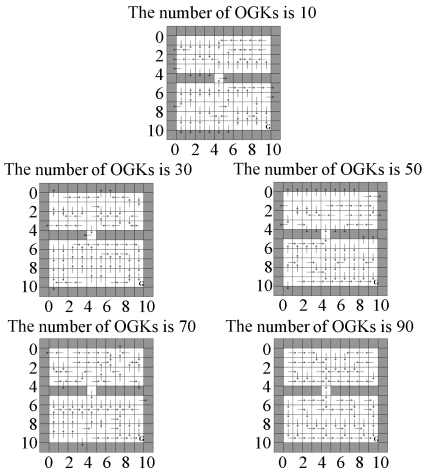


(c) 基于状态 - 动作图测地高斯基的策略迭代强化学习

(c) Policy iteration reinforcement learning based on geodesic Gaussian basis defined on a state-action graph

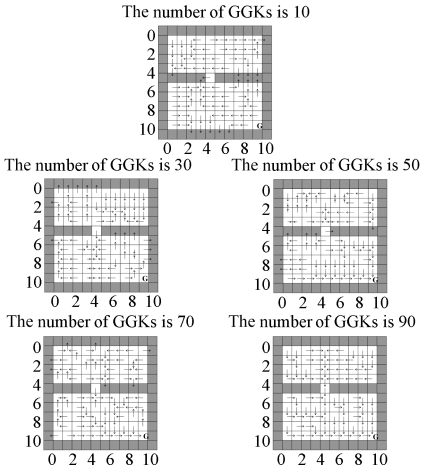
图 3　值函数的估计

Fig. 3　Estimation of value function



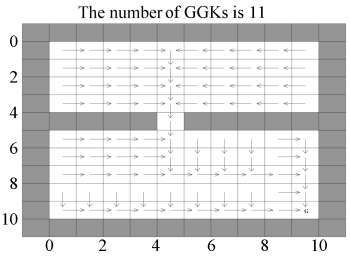
(a) 基于状态图普通高斯基的策略迭代强化学习

(a) Policy iteration reinforcement learning based on ordinary Gaussian basis defined on a state graph



(b) 基于状态图测地高斯基的策略迭代强化学习

(b) Policy iteration reinforcement learning based on geodesic Gaussian basis defined on a state graph



(c) 基于状态 - 动作图测地高斯基的策略迭代强化学习

(c) Policy iteration reinforcement learning based on geodesic Gaussian basis defined on a state-action graph

图 4 最优策略

Fig. 4 Optimal policy

表 1 在最小步数内到达目标点的起始状态数比例

Table 1 Proportions of initial states reaching the goal state within minimal steps

高斯核函数个数	基于状态图普通高斯基的策略迭代	基于状态图测地高斯基的策略迭代
10	8.17 %	4.56 %
30	11.83 %	10.50 %
50	27.61 %	32.39 %
70	57.10 %	59.94 %
90	97.50 %	97.94 %

当高斯核函数的个数取为 10 时, 仅有平均 7.353 个 $(90 \times 8.17\,\%)$ 起始状态能够在 30 步内到达目标点. 图 3 的结果表明, 对于基于状态图的 OGK 和 GGK 方法而言, 较多的核函数有利于值函数估计值精度的提高, 学到的最优策略性能也会相应地得以提高. 但是, 即使核函数的个数多达 90 个, S-OGK-PI 和 S-GGK-PI 强化学习方法也无法确保所有的起始状态均能在最小步数内到达目标点, 即无法保证每次均能学到最优策略, 而基于状态 - 动作图 GGK 的策略迭代强化学习方法能够以 100 % 的概率从任一起始状态在最小步数内到达目标点.

5 结论

在策略迭代强化学习方法中, 用以构造状态 - 动作基函数的方法通常都是先构造状态空间基函数, 然后再将其映射到状态 - 动作空间, 但是这种方法不能够同时泛化状态和动作, 导致动作值函数的估计精度较低. 与状态图相比, 状态 - 动作图可以呈现出不同动作之间的相似性和差异性, 并可以同时泛化状态和动作. 因此, 为了构造合适的基函数以准确逼近动作值函数, 直接在状态 - 动作图上构造测地高斯基函数, 并利用基于近似线性相关的核稀疏方法自动确定基函数的中心, 提出一种基于状态 - 动作图测地高斯基的策略迭代强化学习方法. 与以基于状态图的普通高斯核和测地高斯核作为基函数的两种策略迭代强化学习方法相比, 10×10 格子世界的仿真结果表明, 该方法不但可以构造较少的基函数, 而且逼近的值函数更准确, 并可以为每个状态找到与之相对应的最优策略. 文中建立的状态 - 动作图仅是针对离散状态空间的, 将状态 - 动作图的构造方法扩展到连续状态空间有待于进一步研究.

References

1 Gao Yang, Chen Shi-Fu, Lu Xin. Research on reinforcement learning technology: a review. *Acta Automatica Sinica*, 2004, **30**(1): 86−100
(高阳, 陈世富, 陆鑫. 强化学习研究综述. 自动化学报, 2004, **30**(1): 86−100)

2 Sutton R S, Barto A G. *Reinforcement Learning: An Introduction*. Cambridge: The MIT Press, 1998

3 Wang X S, Cheng Y H, Yi J Q. A fuzzy ActorCritic reinforcement learning network. *Information Sciences*, 2007, **177**(18): 3764−3781

4 Wang Xue-Song, Tian Xi-Lan, Cheng Yu-Hu, Yi Jian-Qiang. Q learning system based on cooperative least squares support vector machine. *Acta Automatica Sinica*, 2009, **35**(2): 214−219
(王雪松, 田西兰, 程玉虎, 易建强. 基于协同最小二乘支持向量机的 Q 学习. 自动化学报, 2009, **35**(2): 214−219)

5 Zhao Dong-Bin, Liu De-Rong, Yi Jian-Qiang. An overview on the adaptive dynamic programming based urban city traffic signal optimal control. *Acta Automatica Sinica*, 2009, 35(6): 676−681
(赵冬斌, 刘德荣, 易建强. 基于自适应动态规划的城市交通信号优化控制方法综述. 自动化学报, 2009, **35**(6): 676−681)

6 Xu X, Hu D W, Lu X C. Kernel-based least squares policy iteration for reinforcement learning. *IEEE Transactions on Neural Networks*, 2007, **18**(4): 973−992

7 Lagoudakis M G, Parr R. Least-squares policy iteration. *Journal of Machine Learning Research*, 2003, **4**: 1107−1149

8 Konidaris G, Osentoski S. Value Function Approximation in Reinforcement Learning Using the Fourier Basis, Technical Report UM-CS-2008-19, Department of Computer Science, University of Massachusetts Amherst, USA, 2008

9 Mahadevan S, Maggioni M. Value function approximation with diffusion wavelets and Laplacian eigenfunctions. In: Proceedings of the Advances in Neural Information Processing Systems 18. Cambridge, USA: The MIT Press, 2006. 843−850

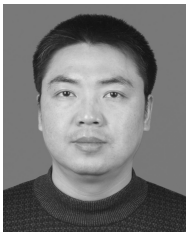
10 Sugiyama M, Hachiya H, Towell C, Vijayakumar S. Value function approximation on non-linear manifolds for robot motor control. In: Proceedings of the IEEE International Conference on Robotics and Automation. Rome, Italy: IEEE, 2007. 1733−1740

11 Osentoski S. Action-based Representation Discovery in Markov Decision Process [Ph. D. dissertation], University of Massachusetts Amherst, USA, 2009

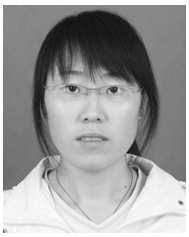
12 Dijkstra E W. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1959, **1**(1): 269−271

13 Duan Fan-Ding. A faster algorithm for shortestpath — SPFA. *Journal of Southwest Jiaotong University*, 1994, **29**(2): 207−212
(段凡丁. 关于最短路径的 SPFA 快速算法. 西南交通大学学报 1994, **29**(2): 207−212)

14 Engel Y, Mannor S, Meir R. The kernel recursive least squares algorithm. *IEEE Transactions on Signal Process* 2004, **52**(8): 2275−2285



程玉虎 中国矿业大学副教授. 主要研究方向为机器学习、智能优化与控制. 本文通信作者.
E-mail: chengyuhu@163.com
(**CHENG Yu-Hu** Associate professor at China University of Mining and Technology. His research interest covers machine learning, intelligent optimization and control. Corresponding author of this paper.)



冯涣婷 中国矿业大学硕士研究生. 主要研究方向为强化学习.
E-mail: fhtcumt@163.com
(**FENG Huan-Ting** Master student at China University of Mining and Technology. Her main research interest is reinforcement learning.)

王雪松 中国矿业大学教授. 主要研究方向为机器学习和生物信息学.
E-mail: wangxuesongcumt@163.com
(**WANG Xue-Song** Professor at China University of Mining and Technology. Her research interest covers machine learning, and bioinformatics.)