文章编号：1000－8152(2002)06－0865－07

Markov 控制过程基于单个样本轨道的在线优化算法*

唐 昊，奚宏生，殷保群
（中国科学技术大学 自动化系，合肥 230026）

摘要：在 Markov 性能势理论基础上，研究了 Markov 控制过程的性能优化算法．不同于传统的基于计算的方法，文中的算法是根据单个样本轨道的仿真来估计性能指标关于策略参数的梯度，以寻找最优（或次优）随机平稳策略．由于可根据不同实际系统的特征来选择适当的算法参数，因此它能满足不同实际工程系统在线优化的需要．最后简要分析了这些算法在一个无限长的样本轨道上以概率 1 的收敛性，并给出了一个三-状态受控 Markov 过程的数值实例．

关键词：Markov 控制过程；Markov 性能势；随机平稳策略；在线优化
中图分类号：TP202 **文献标识码**：A

On-line optimization algorithm for Markov control processes based on a single sample path

TANG Hao, XI Hong-sheng, YIN Bao-qun
(Department of Automation, China University of Science and Technology, Hefei 230026, China)

Abstract: Based on the theory of Markov performance potentials, this paper studies a performance optimization algorithm for Markov control processes. Different from the traditional computation-based approaches, this algorithm could estimate the gradients of performance with respect to the policy parameters by simulating a single sample path, and look for an optimal (or suboptimal) randomized stationary policy. The algorithm provided here could satisfy the needs of on-line optimization of many different real-world engineering systems, because we can select suitable parameters in the algorithm according to the properties of a real system. Finally, the convergence of the algorithm with probability one on an infinite sample path is considered, and a numerical example for a three-state controlled Markov chain is provided.

Key words: Markov control processes; Markov performance potentials; randomized stationary policies; on-line optimization

1 引言(Introduction)

实际中的很多随机决策问题都可模型化为 Markov 控制过程（Markov control process, MCP）或 Markov 决策过程(MDP)，其性能优化问题和灵敏度分析已逐渐成为离散事件动态系统(DEDS)领域的一个重要研究方向．但是对于具有大规模状态空间或状态转移矩阵元素不完全可知的实际系统，一般很难通过直接计算来获得最优策略和灵敏度公式．曹希仁教授和陈翰馥教授等人在文献[1,2]中通过引进 Markov 势理论和实现因子概念而提出的性能灵敏度分析方法则较好地解决了这类系统的灵敏度分析问题，并在我们的最近工作中得到了进一步推广和应用[3~5]．这种方法也可被用来解决一些特殊 Markov 系统的性能优化问题[6]，本文就是针对一类 Markov 遍历链，采用 Markov 性能势理论，并基于单个样本轨道和神经元网络表示的参数化随机平稳策略，导出了适合于 MCP 性能优化的一般算法，同时简要分析了其收敛性．文中给出的算法也可被看作文献[7]中有关 Markov 报酬过程（MRP）的结果在 MCP 性能优化中的推广应用，其优点在于，可针对不同的系统选择不同的算法参数，定时更新网络权系数，以寻找最优随机平稳策略对应的权系数，而且在样本轨道的仿真过程中需要记录的信息量较少，因而能适应具体的实际工程系统在线优化的需要．

* 基金项目：国家自然科学基金(69974037)和国家高性能计算基金(00208)资助项目．
收稿日期：2001－05－14；收修改稿日期：2001－11－13．

2 问题描述和 Markov 性能势(Problem description and Markov performance potentials)

2.1 MCP 基于随机平稳策略的平均代价(Average-cost of MCPs based on randomized stationary policies)

考虑离散时间 Markov 遍历链 $\{X_n, n \geqslant 0\}$,具有有限状态空间 $\Phi = \{1,2,\cdots,M\}$ 和有限行动空间 A. 对 $\forall\, a \in A$,记$[p_{ij}(a)]_{i=1}^{M}|_{j=1}^{M}$是 X_n 在行动 a 驱动下的状态转移概率矩阵,$f(a) = (f(1,a),\cdots,f(M,a))^{\tau}$ 是定义在 A 上的性能函数向量.现假设任意策略参数向量 $\theta = (\theta_1,\theta_2,\cdots,\theta_K) \in \Theta \subseteq \mathbb{R}^K$ 都决定了唯一一个随机平稳策略 $q(\theta)$,则记全体参数化随机平稳策略集为 $Q = \{q(\theta) \mid \theta \in \Theta\}$.在给定的随机策略 $q(\theta)$ 作用下,记 $q_a(i,\theta) \in [0,1]$ 为在状态 i 选择行动 $a \in A$ 的概率,且 $\sum\limits_{a\in A} q_a(i,\theta) = 1$.即 Q 中的每一元素 $q(\theta)$ 都表示一概率分布 $q(\theta):\Phi \to P(A)$,其中 $P(A)$ 是行动空间上的概率分布集.由于人工神经元网络具有较强的函数逼近能力,一般可用一个多层感知器(MLP)或径向基函数网络(RBFN)来构造 $q(\theta)$,如构造 Softmax 函数

$$q_a(i,\theta) = \frac{\exp(r_a(i,\theta))}{\sum\limits_{b\in A}\exp(r_b(i,\theta))},\ \forall\, i \in \Phi, a \in A.$$

这里 $r_a(i,\theta)$ 是网络的输出,状态 - 行动对 (i,a) 为网络的输入,策略参数 θ 即为相应的网络权系数向量.显然,$q_a(i,\theta)$ 服从序列$\{r_a(i,\theta), a \in A\}$ 的 Gibbs 分布,且给定一个网络权值就给定一个随机策略.

Markov 过程在 $q(\theta)$ 作用下的状态转移概率和期望性能函数为

$$\begin{aligned} p_{ij}(\theta) &= \sum_{a\in A} q_a(i,\theta)p_{ij}(a),\\ f(i,\theta) &= \sum_{a\in A} q_a(i,\theta)f(i,a),\ \forall\, i,j \in \Phi. \end{aligned} \tag{2.1}$$

记 $f(\theta) = (f(1,\theta),f(2,\theta),\cdots,f(M,\theta))^{\tau}$,$P(\theta) = [p_{ij}(\theta)]_{i=1}^{M}|_{j=1}^{M}$,我们称 $X = (X_n,\Phi,A,P(\theta),f(\theta))$ 为约束在 Θ 上的离散时间 Markov 控制过程.设 X 的稳态概率向量为 $\pi(\theta) = (\pi(1,\theta),\cdots,\pi(M,\theta))$,则有

$$\pi(\theta)e = 1,\ P(\theta)e = e,\ \pi(\theta)P(\theta) = \pi(\theta). \tag{2.2}$$

其中,$e = (1,1,\cdots,1)^{\tau}$ 是 M 维列向量.可定义 X 关于参数向量 θ 的平均代价性能指标为

$$\eta(\theta) = \lim_{N\to\infty} E\left\{\frac{1}{N}\sum_{n=0}^{N-1} f(X_n,\theta)\right\} = \pi(\theta)f(\theta). \tag{2.3}$$

在 Markov 控制过程 X 中,状态转移规律与控制决策选用的行动方案相互作用决定了状态过程的演化,我们的目的是要选择一控制决策方案,使过程在性能代价准则下达到最优的运行效果.首先有下列假设.

假设 2.1

1) 对 $\forall\, i \in \Phi, a \in A$,函数 $q_a(i,\theta)$ 关于 θ 两次可微,且一阶和二阶导数有界.

2) 对 $\forall\, i \in \Phi, a \in A, \theta \in \Theta$,都存在一个有界的函数向量 $L_a(i,\theta)$,使得

$$\nabla\, q_a(i,\theta) = q_a(i,\theta)L_a(i,\theta). \tag{2.4}$$

文中,符号"∇"表示关于参数向量 θ 的梯度.

3) 记 $\bar{P}$ 为集合$\{P(\theta) \mid \theta \in \Theta\}$ 的闭包,相应于 $\bar{P}$ 中的每一个元素的 Markov 链均是遍历的.

2.2 Markov 性能势(Markov performance potentials)

对任意给定的 $\theta \in \Theta$,我们定义 Markov 过程 X 的 Poisson 方程为

$$(I - P(\theta) + e\pi(\theta))g(\theta) = f(\theta). \tag{2.5}$$

它的一个解向量为

$$g(\theta) = (I - P(\theta) + e\pi(\theta))^{-1}f(\theta). \tag{2.6}$$

定义 2.1 对任意参数向量 $\theta \in \Theta$,称 $g(\theta) + ec$ 为 Markov 控制过程 X 的平均代价性能势向量,它的第 i 个分量为 $g_i(\theta) + c$,其中 c 是任意常数.

Markov 控制过程 X 在状态 i 的性能势可由下式给出[1,2]

$$g(i,\theta) = \lim_{T\to\infty}\{E[\sum_{n=0}^{T-1} f(X_n,\theta) \mid X_0 = i] - T\eta(\theta)\}. \tag{2.7}$$

定义实现因子 d_{ij} 为[1,2]

$$d_{ij}(\theta) = g(j,\theta) - g(i,\theta),\ i,j = 1,2,\cdots,M. \tag{2.8}$$

在一个 Markov 链 $\{X_n, n \geqslant 0, X_0 = i\}$ 上,定义 $N_{ij}(\theta) = \min\{n : n > 0, X_n = j\}$,即 $N_{ij}(\theta)$ 为 Markov 链从状态 i 到状态 j 的首达时间,我们有 $E[N_{ij}(\theta) \mid X_0 = i] < \infty$.根据式(2.7)和(2.8),可得

$$d_{ji}(\theta) = E\{\sum_{n=0}^{N_{ij}(\theta)-1}[f(X_n,\theta) - \eta(\theta)] \mid X_0 = i\}, \tag{2.9}$$

并且

$$d_{ii}(\theta) = 0,\ d_{ij}(\theta) = -d_{ji}(\theta),\ \forall\, i,j. \tag{2.10}$$

同物理学系统中势的意义一样,在 Markov 控制过程

中,状态的性能势是相对的,其表达式也不唯一,但任意两状态的性能势之差是绝对的,因此上面定义的实现因子是完全确定的.

根据公式(2.2),(2.3)和(2.5),很容易证明有下列定理,其证明相似于文献[7,8].

定理 2.1　在假设 2.1 下,平均代价 $\eta(\theta)$ 关于 θ 的梯度为

$$\nabla \eta(\theta) = \pi(\theta)(\nabla f(\theta) + \nabla P(\theta) g(\theta)), \tag{2.11}$$

下面,我们将从式(2.11)出发,导出 Markov 控制过程 X 在参数化随机策略 $q(\theta)$ 作用下,基于单个样本轨道的仿真优化算法,即试图寻求参数向量 θ^* 使得 $\theta^* = \arg\min_{\theta\in\Theta}\eta(\theta)$. 最优参数 θ^* 确定了一个最优随机平稳策略,但在多数情况下,我们得到的是次最优策略.

3　MCP 在线优化算法(An on-line optimization algorithm for MCPs)

在参数寻优中,常用的方法是梯度算法,即最陡下降法. 在离散时间 Markov 过程中,若系统模型完全可知,梯度 $\nabla\eta(\theta)$ 可以根据式(2.6)和(2.11)精确计算,参数寻优按如下迭代公式进行

$$\theta := \theta - \gamma \nabla \eta(\theta),$$

但在大状态空间问题中,一般很难求解式(2.6)和(2.11),而且有些工程实际系统模型不完全可知,也不能按上式精确计算梯度 $\nabla\eta(\theta)$. 因此,不同于传统的基于计算的优化方法,我们需要考虑基于仿真和梯度估计的在线优化算法. 首先,根据式(2.1),(2.4)和(2.11),我们有

$$\begin{aligned}
&\nabla\eta(\theta) = \\
&\sum_{i\in\Phi}\pi_i(\theta)(\nabla f(i,\theta) + \sum_{j\in\Phi}\nabla P_{ij}(\theta) g(j,\theta)) = \\
&\sum_{i\in\Phi}\pi_i(\theta)(\nabla f(i,\theta) + \\
&\sum_{j\in\Phi}\sum_{a\in A} q_a(i,\theta) P_{ij}(a) L_a(i,\theta) g(j,\theta)) = \\
&\sum_{i\in\Phi} E[\chi_i(X_n)\nabla f(i,\theta)] + \\
&\sum_{i,j\in\Phi}\sum_{a\in A} E[\chi_i(X_n)\chi_j(X_{n+1})\chi_a(a_n) L_a(i,\theta) g(j,\theta)].
\end{aligned} \tag{3.1}$$

这里 a_n 是根据概率 $q_a(X_n,\theta)$ 随机选取的行动,$\chi_i(\cdot)$ 和 $\chi_a(\cdot)$ 分别是状态 i 和行动 a 的示性函数,例如

$$\chi_i(X_n) = \begin{cases} 1, & \text{if } X_n = i, \\ 0, & \text{otherwise.} \end{cases}$$

假设对任意给定的参数 θ, Markov 控制过程 X 在随机平稳策略 $q(\theta)$ 作用下,得到一条样本轨道 $(X_{u_0}, X_{u_0+1}, \cdots, X_{u_1}, X_{u_1+1}, \cdots, X_{u_N}, \cdots)$,并且定义

$$\begin{aligned}
&u_0 = 0,\ X_{u_0} = i_0; \\
&u_{k+1} = \min\{n : n > u_k,\ X_n = i_0\},\ 0 \leqslant k.
\end{aligned} \tag{3.2}$$

不失一般性,令这里的 i_0 为选定的初始状态. 因此 u_k 为第 k 次回到初始状态的时刻,是过程的再生点. 显然 $u_1 - u_0, u_2 - u_1, \cdots, u_N - u_{N-1}, \cdots$ 为独立同分布随机变量.

根据式(3.1),可用该样本轨道提供的一些信息来估计势和梯度,以构造算法,下面进行具体分析.

3.1　势的估计(Estimation of potentials)

现给定前面定义的一条样本轨道,令 $g(i_0,\theta) \equiv c$,这里 c 是任意常数. 则对任意整数 $k \geqslant 1$ 和任意整数 $u_{k-1} \leqslant n < u_k$,由式(2.7)~(2.10),我们有

$$g(X_n,\theta) = -d_{X_n X_{u_k}}(\theta) + g(X_{u_k},\theta) = E[\sum_{t=n}^{u_k-1}(f(X_t,\theta) - \eta(\theta))] + c.$$

为简单起见,令 $c = 0$,这时 $g(\theta)$ 和文献[7]中相对报酬 $v(\theta)$ 的定义一样. 于是把

$$\tilde{g}(X_n,\theta) = \begin{cases} 0, & \text{if } n = u_{k-1}, \\ \sum_{t=n}^{u_k-1}(f(X_t,\theta) - \tilde{\eta}), & \text{otherwise.} \end{cases} \tag{3.3}$$

当作势 $g(X_n,\theta)$, $u_{k-1} \leqslant n \leqslant u_k$ 的估计. 其中,假设 $\tilde{\eta}$ 为平均代价 $\eta(\theta)$ 的一个在 n 时刻已经得到的估计,它不依赖于当前时刻的状态 X_n,只与 n 时刻以前的历史有关.

3.2　梯度的估计(Estimation of gradients)

根据前面的样本轨道,对给定的整数 N, 我们记

$$F^N(\theta) = \sum_{n=u_0}^{u_N-1}[\nabla f(X_n,\theta) + L_{a_n}(X_n,\theta) g(X_{n+1},\theta)]. \tag{3.4}$$

其中, a_n 为系统在状态 X_n 时根据 $q_a(X_n,\theta), a \in A$ 随机选取的行动. 很显然,我们有

$$\begin{aligned}
&E_\theta[F^N(\theta)] = \\
&NE_\theta\{\sum_{n=u_0}^{u_1-1}[\nabla f(X_n,\theta) + L_{a_n}(X_n,\theta) g(X_{n+1},\theta)]\} = \\
&NE_\theta[u_1 - u_0]\nabla\eta(\theta) = NE_\theta[u_1]\nabla\eta(\theta).
\end{aligned}$$

可见 $F^N(\theta)/NE_\theta[u_1]$ 为 $\nabla\eta(\theta)$ 的一个无偏估计，即 $F^N(\theta)$ 为梯度方向的一个无偏估计. 当式(3.4)中的势难以直接计算时，需用前面定义的估计值 $\tilde{g}(X_n,\theta)$ 来代替.我们定义

$$F^N(\theta,\tilde{\eta})=\sum_{n=u_0}^{u_N-1}[\nabla f(X_n,\theta)+L_{a_n}(X_n,\theta)\tilde{g}(X_{n+1},\theta)]=F^{(1)}(\theta,\tilde{\eta})+F^{(2)}(\theta,\tilde{\eta})+\cdots+F^{(N)}(\theta,\tilde{\eta}). \tag{3.5}$$

其中

$$F^{(k)}(\theta,\tilde{\eta})=\sum_{n=u_{k-1}}^{u_k-1}[\nabla f(X_n,\theta)+L_{a_n}(X_n,\theta)\tilde{g}(X_{n+1},\theta)],\quad 1\leqslant k\leqslant N \tag{3.6}$$

为独立同分布的随机向量.记

$$f^N(\theta,\tilde{\eta})=E_\theta[F^N(\theta,\tilde{\eta})],\quad f^{(k)}(\theta,\tilde{\eta})=E_\theta[F^{(k)}(\theta,\tilde{\eta})]. \tag{3.7}$$

定理 3.1 对给定整数 N，我们有

$$f^{(k)}(\theta,\tilde{\eta})=E_\theta[u_1-u_0]\nabla\eta(\theta)+G(\theta)(\eta(\theta)-\tilde{\eta}),\quad \forall\, 1\leqslant k\leqslant n \tag{3.8}$$

且 $f^N(\theta,\tilde{\eta})=Nf^{(1)}(\theta,\tilde{\eta})$. 其中

$$G(\theta)=E_\theta\Big[\sum_{n=u_0+1}^{u_1-1}(u_1-n)L_{a_{n-1}}(X_{n-1},\theta)\Big].$$

定理 3.2 若整数 N 是序列 $F^{(1)}(\theta,\tilde{\eta})$，$F^{(2)}(\theta,\tilde{\eta}),\cdots$ 的停时，且 $E[N]<\infty$，则我们有

$$f^N(\theta,\tilde{\eta})=E_\theta[N]f^{(1)}(\theta,\tilde{\eta}). \tag{3.9}$$

定理3.1和3.2的证明：因为 $F^{(k)}(\theta),1\leqslant k\leqslant N$ 和 $u_k-u_{k-1},k\geqslant 1$ 分别是独立同分布随机序列，根据文献[7]中的定理2,很容易证得定理3.1.由式(3.5)和著名的 Wald 等式[9]以及定理3.1,直接有定理3.2. 证毕.

3.3 在线优化算法(An on-line optimization algorithm)

由式(2.1),(2.4),(3.3)和(3.6),我们有

$$\begin{aligned}
&F^{(k)}(\theta,\tilde{\eta})=\\
&\sum_{n=u_{k-1}}^{u_k-1}[\nabla f(X_n,\theta)+L_{a_n}(X_n,\theta)\tilde{g}(X_{n+1},\theta)]=\\
&\sum_{n=u_{k-1}}^{u_k-1}\nabla f(X_n,\theta)+\sum_{n=u_{k-1}+1}^{u_k-1}L_{a_{n-1}}(X_{n-1},\theta)\sum_{t=n}^{u_k-1}(f(X_t,\theta)-\tilde{\eta})=\\
&\sum_{n=u_{k-1}}^{u_k-1}\nabla f(X_n,\theta)+\sum_{n=u_{k-1}+1}^{u_k-1}(f(X_n,\theta)-\tilde{\eta})\sum_{t=u_{k-1}+1}^{n}L_{a_{t-1}}(X_{t-1},\theta)=\\
&\sum_{n=u_{k-1}}^{u_k-1}\sum_{a\in A}\nabla q_a(X_n,\theta)(f(X_n,a)-\tilde{\eta})+\\
&\sum_{n=u_{k-1}}^{u_k-1}\sum_{a\in A}q_a(X_n,\theta)(f(X_n,a)-\tilde{\eta})\sum_{t=u_{k-1}+1}^{n}L_{a_{t-1}}(X_{t-1},\theta)=\\
&\sum_{n=u_{k-1}}^{u_k-1}\sum_{a\in A}q_a(X_n,\theta)L_a(X_n,\theta)(f(X_n,a)-\tilde{\eta})+\\
&\sum_{n=u_{k-1}}^{u_k-1}\sum_{a\in A}q_a(X_n,\theta)(f(X_n,a)-\tilde{\eta})\sum_{t=u_{k-1}+1}^{n}L_{a_{t-1}}(X_{t-1},\theta)=\\
&\sum_{n=u_{k-1}}^{u_k-1}\sum_{a\in A}q_a(X_n,\theta)(f(X_n,a)-\tilde{\eta})z_n^{(k)}(\theta).
\end{aligned} \tag{3.10}$$

其中

$$z_n^{(k)}(\theta)=\begin{cases}L_{a_n}(X_n,\theta), & \text{if } n=u_{k-1},\\ z_{n-1}^{(k)}(\theta)+L_{a_n}(X_n,\theta), & \text{otherwise.}\end{cases}$$

因此，根据式(3.5)和(3.10)，我们有

$$F^N(\theta,\tilde{\eta})=\sum_{n=u_0}^{u_N-1}\sum_{a\in A}q_a(X_n,\theta)(f(X_n,a)-\tilde{\eta})z_n(\theta). \tag{3.11}$$

其中

$$z_n(\theta)=\begin{cases}L_{a_n}(X_n,\theta), & \text{if } n=u_0,u_1,\cdots,u_{N-1},\\ z_{n-1}(\theta)+L_{a_n}(X_n,\theta), & \text{otherwise.}\end{cases} \tag{3.12}$$

因此，根据式(3.11)，对给定的参数 θ 和某个确定性整数 T，在一条样本轨道上，可分别把 T 个相继的 $(f(X_n,a)-\tilde{\eta})z_n(\theta)$ 的累积值作为相应梯度方向的估计值，以便进行参数更新.于是得到下列更新算法.

$$\begin{cases}\theta_{k+1}=\theta_k-\gamma_k\sum_{n=kT}^{(k+1)T-1}(f(X_n,a_n)-\tilde{\eta}_k)z_n(\theta_k),\\ \tilde{\eta}_{k+1}=\tilde{\eta}_k+\beta\gamma_k\sum_{n=kT}^{(k+1)T-1}(f(X_n,a_n)-\tilde{\eta}_k).\end{cases} \tag{3.13}$$

这里 $\{\gamma_k\}$ 是步长序列；β 为正标量系数，以调整 θ 和 $\tilde{\eta}$ 的相对更新速率.其中，参数 $\tilde{\eta}$ 的更新相当于一种自适应调整.具体的算法过程如下：

步骤 1 根据实际工程系统，选定初始状态 i_0，令 $X_0=X_{u_0}=i_0$，且选择好整数 T 和正标量系数 β 以及步长序列 $\{\gamma_k\}$，令 $k=0$，并初始化参数向量 $\theta_0,\tilde{\eta}_0$.

步骤 2 在线观测随机平稳策略 $q(\theta_k)$ 作用下的系统，得到一条样本轨道 $(X_{kT},X_{kT+1},\cdots,X_{(k+1)T})$.

步骤 3 按式(3.13)进行参数更新.

步骤 4 判断算法是否满足终止条件(譬如判

断是否达到给定的更新步数),若不满足,令 $k := k+1$,回到步骤 2;若满足,则停止更新.

上面分析的算法,直接推广了文献[7]中的算法在 Markov 控制过程中的应用,为基于单样本轨道的在线优化提供了更灵活的选择.可以根据系统的实际情况和实时更新的需要选择 T 的大小,便于定时更新参数,不必如文献[7]中提供的两种算法,只是在系统访问到固定常返状态时或每步状态转移之后进行参数更新.在状态变化很快或返回初始状态频率较高的系统中,可选择 T 大一点,以利用更多的样本信息来更新参数,譬如在高速通讯网络系统中;反之,选择 T 小一点,甚至可令 $T=1$(这时算法相似于文献[7]中的在每步状态转移之后进行参数更新的算法),以保证参数的更新频率,譬如在柔性制造系统中.而且我们提供的这种算法,在每步状态转移之后,只需计算 $z_n(\theta_k)$ 和记录观测到的代价 $f(X_n, a_n)$,因此节省了信息的存储空间.

为减小更新方差,可引入遗忘因子 α. 把式(3.12)中 $z_n(\theta)$ 替代为

$$z_n(\theta)=\begin{cases} L_{a_n}(X_n,\theta), & \text{if } n=u_0,u_1,\cdots,u_{N-1},\cdots,\\ \alpha z_{n-1}(\theta)+L_{a_n}(X_n,\theta), & \text{otherwise.}\end{cases}$$

4 算法的收敛性(Convergence of the algorithm)

为证明算法的收敛性,首先引入下列假设.

假设 4.1　a) 步长序列 $\{\gamma_k\}$ 非负且满足

$$\sum_{k=0}^{\infty}\gamma_k=\infty,\quad \sum_{k=0}^{\infty}\gamma_k^2<\infty.$$

b) 步长序列 $\{\gamma_k\}$ 单调不增,且存在一个正整数 p 和一个正标量 A,使得

$$\sum_{k=n}^{n+t}(\gamma_n-\gamma_k)\leqslant At^p\gamma_n^2,\quad \forall\, n,t>0.$$

假设 4.2　对初始状态 i_0 和给定的正整数 T,存在正整数 N_0,使得对 $\bar{P}$ 中的每 N_0 个矩阵构成的集合 $\{P_1,\cdots,P_{N_0}\}$,其中的元素 $P_l, l=1,2,\cdots,N_0$ 满足 $\sum_{n=1}^{N_0}[\prod_{l=1}^{n}P_l^{\mathrm{T}}]_{ii_0}>0, \forall\, i\in\Phi$.

这样,有下面定理,其证明见附录.

定理 4.1　在假设 2.1 和 4.1 及 4.2 下,对任意正整数 T,记 $\{\theta_k\}$ 为算法(3.13)式产生的参数向量序列,则 $\eta(\theta_k)$ 以概率 1 收敛,且 $\nabla\eta(\theta_k)$ 以概率 1 收敛于零.

5 实例(An example)

考虑一个三-状态受控 Markov 链, $\Phi=\{1,2,3\}$, $A=\{1,2\}$,如图 1 所示.在行动 $a=1$ 下,系统的一步概率转移矩阵 $[p_{ij}(a)]_{i=1}^{3}|_{j=1}^{3}$ 和一步转移代价矩阵 $[f(i,a,j)]_{i=1}^{3}|_{j=1}^{3}$ 分别为

$$\begin{bmatrix}0.5&0.25&0.25\\0.25&0.25&0.5\\0.5&0.25&0.25\end{bmatrix},\quad\begin{bmatrix}1&5&1\\1&1&1\\0.75&0.75&0.75\end{bmatrix};$$

在行动 $a=2$ 下,两矩阵分别为

$$\begin{bmatrix}0.25&0.5&0.25\\0.25&0.5&0.25\\0.25&0.25&0.5\end{bmatrix},\quad\begin{bmatrix}0.5&0.5&0.5\\1&1&5\\5&1&1\end{bmatrix}.$$

在式(2.1)中,定义 $f(i,a)=\sum_{j\in\Phi}p_{ij}(a)f(i,a,j)$. 可直接由相应 Bellman 方程解得最优策略,即在状态 1 以概率 1 采用行动 2,在状态 2 和 3 以概率 1 采用行动 1,对应最优平均代价为 $\eta=0.75$.

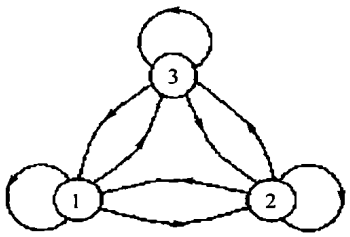

图 1　三-状态受控 Markov 链

Fig. 1　Three-state controlled Markov chain

仿真计算中,三个状态被编码为(0,0,1),(0,1,0),(1,0,0).按前面所述方法,引入随机策略,因为是双-行动问题,随机策略可用一个 $3\times1\times1$ 的前向神经元网络表示,其中网络输入为状态的编码,隐节点没有固定偏置输入,网络输出层的节点函数采用恒等函数,且输出表示选用行动 1 的概率.我们选择零初始参数向量,即初始策略以相等概率选用行动 1 和 2,仿真波形见图 2 所示.这里,(a)图是参数为 $T=3, \alpha=0.99, \beta=0.4$ 时的波形,横坐标为迭代步数,其中上图曲线表示平均代价的估计值 $\bar{\eta}$,下图三条曲线分别表示在状态 1,2,3 采用行动 1 的概率;(b)图是参数为 $T=4, \alpha=0.99, \beta=0.2$ 时的波形.

可见,算法在经过最初的更新调整后,在状态 1 以较小概率采用行动 1,而在状态 2 和 3 以较大概率采用行动 1,并逐渐趋近于最优值.算法的后期收敛很慢,一方面是因为所用的步长序列是递减的,另一方面是因为本例的最优策略为退化的确定性策略,对应的最优参数在网络隐节点的饱和区域内.

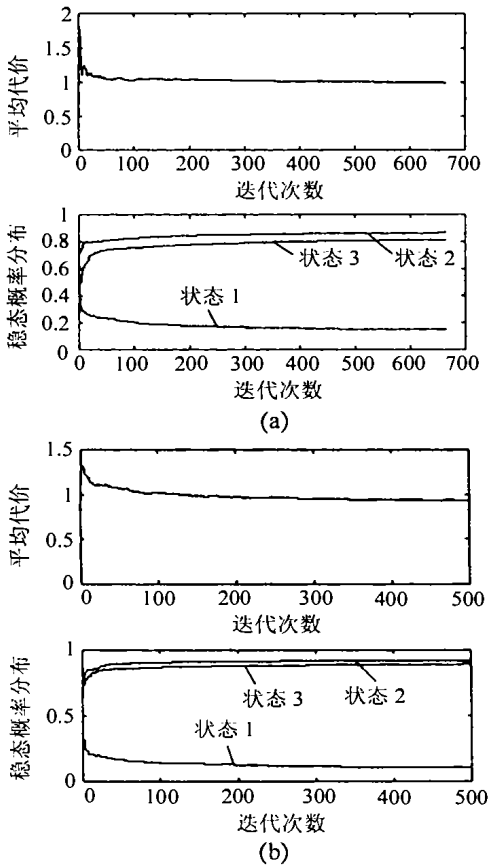

图2 三-状态受控Markov链对应2000次状态转移的仿真计算结果

Fig. 2 Computation results for three-state controlled Markov chain by 2000 state transitions

6 总结(Conclusions)

本文研究的离散时间、有限状态Markov控制过程基于单个样本轨道的在线优化算法,是近些年基于仿真学习思想及神经元网络理论而提出的神经元动态规划(NDP)或称强化学习(RL)方法[10,11]的一个具体应用.文中给出的算法也可以推广到具有一般状态空间和行动空间或连续时间的Markov控制过程,适合于解决模型不完全可知或存在状态空间"维数灾"的实际系统的优化问题,并可通过选择适当的算法参数,应用于不同的实际工程系统.

参考文献(References)

[1] Cao X R, Chen H F. Perturbation realization, potentials, and sensitivity analysis of Markov processes [J]. IEEE Trans. Automatic Control, 1997,42(10):1382 – 1393

[2] Cao X R, Wan Y W. Algorithms for sensitivity analysis of Markov systems through potentials and perturbation realization [J]. IEEE Trans. Control Systems Technology, 1998,6(4):482 – 494

[3] Yin B Q, Zhou Y P, Yang X X, et al. Sensitivity formulas of performance in state-dependent closed queueing networks [J]. Control Theory and Applications, 1999,16(2):255 – 257 (in Chinese)

[4] Yin B Q, Zhou Y P, Xi H S, et al. Sensitivity formulas of performance in two-server cyclic queueing networks with phase-type distributed service times [J]. International Transactions in Operation Research, 1999,6(6): 649 – 663

[5] Yin B Q, Zhou Y P, Xi H S. Sensitivity analysis of performance in queueing systems with phase-type distributed service times [J]. Operations Research Transactions, 2000,4(4):55 – 62 (in Chinese)

[6] Zhou Y P. Yin B Q, Xi H S, et al. Algorithms of decentralized optimization for a class of closed queueing network by using performance potentials [J]. Journal of University of Science and Technology of China, 2000,30(2):151 – 157 (in Chinese)

[7] Marbach P, Tsitsiklis J N. Simulation-based optimization of Markov reward processes [J]. IEEE Trans. Automatic Control, 2001, 46 (2):191 – 209

[8] Bartlett P L, Baxter J. Stochastic optimization of controlled partially observable Markov decision processes [A]. Proceedings of the 39th Conference on Decision and Control [C]. Sydney, Australia, 2000, 124 – 129

[9] Sheldon M Ross. Stochastic Process [M]. New York: Wiley, 1983

[10] Bertsekas D P, Tsitsiklis J N. Neuro-Dynamic Programming [M]. Belmont, MA: Athena Scientific, 1996

[11] Sutton R S, Barto A G. Reinforcement Learning: An Introduction [M]. Cambridge, MA:MIT Press, 1998

附录 A(Appendix A)

定理 4.1 的证明:给定参数向量 θ_m,根据随机平稳策略 $q(\theta_m)$ 产生一条样本轨道$(X_{u_{0,m}},\cdots,X_{u_{1,m}},\cdots,X_{u_{N,m}-1},X_{u_{N_m,m}})$.根据式(3.5)和(3.6)定义 $F_m^{N_m}(\theta,\tilde{\eta})$ 和 $F_m^{(k)}(\theta,\tilde{\eta}),1\leqslant k\leqslant N_m$,构造下列算法更新公式

$$\begin{cases}\theta_{m+1}=\theta_m-\gamma_m F_m^{N_m}(\theta_m,\tilde{\eta}_m),\\ \tilde{\eta}_{m+1}=\tilde{\eta}_m+\beta\gamma_m\sum\limits_{n=u_{0,m}}^{u_{N_m,m}-1}(f(X_n,\theta_m)-\tilde{\eta}_m).\end{cases}\tag{A1}$$

算法中,序列 $u_{0,0},u_{1,0},\cdots,u_{N_0,0},\cdots,u_{0,m},u_{1,m},\cdots,u_{N_m,m}\cdots$ 的定义同式(3.2)一样,其中 m 表示参数已经更新的次数.首先我们有下列引理.

引理 A1 如果 $N_m,m\geqslant 0$ 为一固定的正整数,或 N_m 是序列 $F_m^{(1)}(\theta,\tilde{\eta}),F_m^{(2)}(\theta,\tilde{\eta}),\cdots$ 的停时,且 $E[N_m]<\infty$,记 $\{\theta_m\}$ 为式(A1)产生的参数向量序列.若假设2.1和4.1a)成立,则 $\eta(\theta_m)$ 以概率1收敛,且 $\nabla\eta(\theta_m)$ 以概率1收敛于零.

证 先假设 $N_m,m\geqslant 0$ 为一固定的正整数,记 $N_m=N$.令 F_m 为算法(A1)式直到第 m 次更新且包括初始点在内的历史,$r_m=(\theta_m,\tilde{\eta}_m)^{\mathrm{T}}$ 为增广向量,则式(A1)可写为

$$r_{m+1}=r_m+\gamma_m H_m^N(r_m)=r_m+\gamma_m h^N(r_m)+\varepsilon_m.\tag{A2}$$

这里

$$H_m^N(r_m)=\begin{bmatrix}-F_m^N(\theta_m,\tilde{\eta}_m)\\ \beta\sum\limits_{n=u_{0,m}}^{u_{N,m}-1}(f(X_n,\theta_m)-\tilde{\eta}_m)\end{bmatrix},$$

$$h^N(r_m) = E[H_m^N(r_m) \mid F_m],$$
$$\varepsilon_m = \gamma_m(H_m^N(r_m) - h^N(r_m)).$$

其中

$$h^N(r) = \begin{bmatrix} -NE_\theta[u_1 - u_0]\nabla\eta(\theta) - NG(\theta)(\eta(\theta) - \tilde{\eta}) \\ N\beta E_\theta[u_1 - u_0](\eta(\theta) - \tilde{\eta}) \end{bmatrix}. \tag{A3}$$

并且，$E[\varepsilon_m \mid F_m] = 0$.

当 $N = 1$ 时，上式可直接看作文献[7]中提供的第一种算法在 Markov 控制过程中的应用，引理的证明可直接运用该文献中定理 3 的证明结果；当 N 为大于 1 的确定性正整数时，按同样的线索也可证明 $\lim_{m\to\infty}(r_{m+1} - r_m) \xrightarrow{\text{w.p.1}} 0$，$\lim_{m\to\infty}\nabla\eta(\theta_m) \xrightarrow{\text{w.p.1}} 0$，此处不再赘述.

若 N_m 是序列 $F_m^{(1)}(\theta,\tilde{\eta}), F_m^{(2)}(\theta,\tilde{\eta}),\cdots$ 的停时，根据定理 3.2 同样可写出形如(A2)的公式，需要改变的是把式(A3)中的 N 用 $E[N_m]$ 代替. 因为 $E[N_m] < \infty$，不影响算法的收敛性证明，仍有上面的结果.　　证毕.

现在分析定理 4.1 的证明. 首先，对应算法(3.13)式的更新过程，相似于(3.2)式，重新定义序列 $u_{0,0} = 0$，

$$u_{1,0} = \min\{n : n > u_{0,0}, X_n = i_0\},\cdots,$$
$$u_{0,m} = \min\{nT : nT > u_{0,m-1}, X_{nT} = i_0\},$$
$$u_{1,m} = \min\{n : n > u_{0,m}, X_n = i_0\},\cdots.$$

令 $u_{N_m,m} = u_{0,m+1}$，$w_m = (u_{N_m,m} - u_{0,m})/T$，$m \geqslant 0$，这里 N_m 表示系统在部分样本轨道 $(X_{u_{0,m}}, X_{u_{0,m}+1},\cdots, X_{u_{N_m,m}})$ 上返回到状态 i_0 的次数，随机变量 w_m 为参数更新次数. 重新定义 $F_m = \{\theta_0,\tilde{\eta}_0, X_{u_{0,0}},\cdots, X_{u_{0,m}}\}$ 为算法直到且包括时间 $u_{0,m}$ 在内的历史，记 $u_{0,m}$ 时刻的参数为 $\{\theta_{l_m},\tilde{\eta}_{l_m},\gamma_{l_m}\}$，其中 $l_0 = 0, l_m = w_0 + \cdots + w_{m-1}$. 定义

$$\theta_n = \theta_{l_m},\ \tilde{\eta}_n = \tilde{\eta}_{l_m},\ \gamma_n = \gamma_{l_m},\ u_{0,m} \leqslant n < u_{0,m} + T,$$
$$\theta_n = \theta_{l_m+1},\ \tilde{\eta}_n = \tilde{\eta}_{l_m+1},\ \gamma_n = \gamma_{l_m+1},$$
$$u_{0,m} + T \leqslant n < u_{0,m} + 2T,$$
$$\vdots$$
$$\theta_n = \theta_{l_m+w_m-1},\ \tilde{\eta}_n = \tilde{\eta}_{l_m+w_m-1},\ \gamma_n = \gamma_{l_m+w_m-1},$$
$$u_{0,m} + (w_m - 1)T \leqslant n < u_{N_m,m},$$
$$r_n = (\theta_n,\tilde{\eta}_n)^\tau,\ R(r_n) = \begin{bmatrix} -f(X_n,a_n) - \tilde{\eta}_n)z_n(\theta_n) \\ \beta(f(X_n,a_n) - \tilde{\eta}_n) \end{bmatrix}.$$

再记 $r(l_m) = (\theta_{l_m},\tilde{\eta}_{l_m})^\tau$，则根据式(3.13)我们有

$$r(l_{m+1}) = r(l_m) + \sum_{n=u_{0,m}}^{u_{Nm,m}-1}\gamma_n R(r_n) = r(l_m) + \hat{\gamma}_m\hat{h}(r(l_m)) + \hat{\varepsilon}_m. \tag{A4}$$

这里 $\hat{\gamma}_m = \sum_{n=u_{0,m}}^{u_{Nm,m}-1}\gamma_n, \hat{\varepsilon}_m = \sum_{n=u_{0,m}}^{u_{Nm,m}-1}\gamma_n(R(r_n) - \hat{h}(r(l_m)))$，且

$$\hat{h}(r) = \begin{bmatrix} -\nabla\eta(\theta) - G(\theta)(\eta(\theta) - \tilde{\eta})/E_\theta[u_1 - u_0] \\ \beta(\eta(\theta) - \tilde{\eta}) \end{bmatrix}.$$

不难证明 $\hat{h}(r)$ 以概率 1 有界，并且根据假设 4.2，我们有下列引理.

引理 A2　对任意正整数 s，都存在一个常数 D_s，使得

$$E[(u_{N_m,m} - u_{0,m})^s \mid F_m] \leqslant D_s.$$

另外，根据假设 4.1 和上面引理，很容易验证有下列引理.

引理 A3　我们以概率 1 有

$$\sum_{m=0}^{\infty}\hat{\gamma}_m = \infty,\ \sum_{m=0}^{\infty}\hat{\gamma}_m^2 < \infty.$$

因为 N_m 为均值有界的随机变量，且式(A4)和式(A2)的形式一样，根据前面的三个引理，不难理解算法(A4)式的收敛性. 但是在部分样本轨道 $(X_{u_{0,m}}, X_{u_{0,m}+1},\cdots, X_{u_{N_m,m}})$ 上，θ_k 经过多次更新，引理 A1 的条件不完全满足，不能直接引用该引理的结果. 需要指出的是，引理 A1 证明的关键一步是证明序列 $\sum_m\varepsilon_m$ 以概率 1 收敛，这里我们仍然可以证明序列 $\sum_m\hat{\varepsilon}_m$ 以概率 1 收敛，并且 $\lim_{m\to\infty}(\theta_n - \theta_k) \xrightarrow{\text{w.p.1}} 0, n,k \in [u_{0,m}\quad u_{0,m+1}]$. 在引理 A2 和 A3 下，这部分的证明相似于文献[7]中定理 4 的证明过程，此处不再赘述. 有了这个结果，再引用引理 A1 的证明，就能证得 $\lim_{m\to\infty}(r(l_{m+1}) - r(l_m)) \xrightarrow{\text{w.p.1}} 0, \lim_{m\to\infty}\nabla\eta(\theta_{l_m}) \xrightarrow{\text{w.p.1}} 0, \lim_{k\to\infty}\nabla\eta(\theta_k) \xrightarrow{\text{w.p.1}} 0.$

证毕.

本文作者简介

唐　昊　1972 年生. 1995 年毕业于安徽工学院电气系，并免试进入中国科学院等离子体所学习且于 1998 年获得工学硕士学位. 现为中国科学技术大学自动化系博士生. 主要从事离散事件动态系统的性能优化和应用以及神经元动态规划方法等方面的研究.　E-mail: xihs@ustc.edu.cn

奚宏生　见本刊 2002 年第 2 期第 312 页.

殷保群　见本刊 2002 年第 2 期第 312 页.