第 5 章　无穷阶段强化学习

本章将采用近似动态规划/强化学习方法来近似求解前一章中介绍的无穷阶段随机最短路径问题和折扣问题。具体来说，我们将考虑近似版本的值迭代和策略迭代算法。在此过程中，我们将频繁采用动态规划算子 T 和 T_μ（也称为贝尔曼算子）来进行讲解。这两个算子将 n 维向量 J 分别映射到 n 维向量 TJ 和 $T_\mu J$，并且简化了算法与分析的表述。为了便于参考，我们将符号定义列举如下：

对于随机最短路径问题：针对所有 i，引入

$$(TJ)(i) = \min_{u\in U(i)} \left[p_{it}(u)g(i,u,t) + \sum_{j=1}^{n} p_{ij}(u)\big(g(i,u,j) + J(j)\big) \right] \tag{5.1}$$

并针对所有的 μ 和 i，引入

$$(T_\mu J)(i) = p_{it}\big(\mu(i)\big)g\big(i,\mu(i),t\big) + \sum_{j=1}^{n} p_{ij}\big(\mu(i)\big)\Big(g\big(i,\mu(i),j\big) + J(j)\Big) \tag{5.2}$$

对于折扣问题：针对所有 i，引入

$$(TJ)(i) = \min_{u\in U(i)} \sum_{j=1}^{n} p_{ij}(u)\big(g(i,u,j) + \alpha J(j)\big) \tag{5.3}$$

并针对所有的 μ 和 i，引入

$$(T_\mu J)(i) = \sum_{j=1}^{n} p_{ij}\big(\mu(i)\big)\Big(g\big(i,\mu(i),j\big) + \alpha J_\mu(j)\Big) \tag{5.4}$$

5.1　值空间近似——性能界

本节以折扣问题为出发点，介绍适用于无穷阶段动态规划问题的值空间近似方法的一般框架。与第 2 章中介绍的有限阶段问题的相应方法一致，这些方法的基本思路是首先计算最优费用函数 J^* 的近似 $\tilde{J}$，然后通过执行一步或多步前瞻得到策略 $\tilde{\mu}$。因此，一步前瞻策略（one-step lookahead policy）在状态 i 所选取的控制 $\tilde{\mu}(i)$ 会取得表达式

$$\min_{u\in U(i)} \sum_{j=1}^{n} p_{ij}(u)\big(g(i,u,j) + \alpha\tilde{J}(j)\big) \tag{5.5}$$

的最小值，参见图 5.1.1。

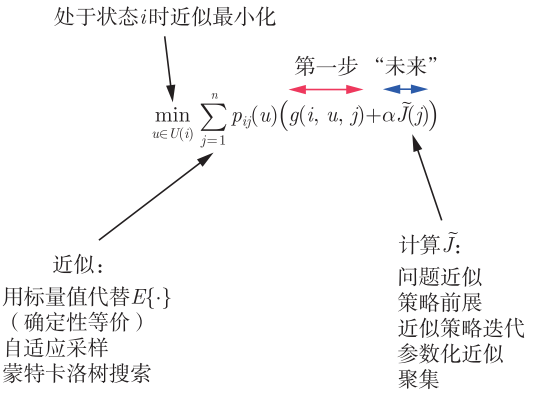

图 5.1.1 针对无穷阶段问题，采用一步前瞻的值空间近似方法有众多选择。此处前瞻函数的值 $\tilde{J}(j)$ 用于近似最优展望费用值 $J^*(j)$，并且可以通过多种方式计算得到。在此基础上，我们还可以引入针对 u_k 的最小化计算的近似以及期望值计算的近似。

类似地，两步前瞻策略（two-step lookahead policy）在状态 i 所选取的控制 $\tilde{\mu}(i)$ 也会最小化前面表达式的值，只是此时的函数 $\tilde{J}$ 本身就是在一步前瞻近似的基础上得到的。换句话说，对于所有可以从 i 抵达的状态 j，有

$$\tilde{J}(j) = \min_{u\in U(j)} \sum_{m=1}^{n} p_{jm}(u)\big(g(j,u,m) + \alpha\hat{J}(m)\big)$$

而此处的 $\hat{J}$ 则是另一个对于 J^* 的近似。因此，$\tilde{J}$ 是从 $\hat{J}$ 出发，通过一步值迭代得到的函数。其他通过多于两步的前瞻得到的策略也可以用类似方法定义。在 ℓ 步前瞻中，“有效的一步”费用近似 $\tilde{J}$ 是从某个初始值 $\hat{J}$ 出发、经过 $\ell-1$ 步值迭代运算后得到的结果。如果采用贝尔曼算子 T 来表述的话 [参见式 (5.3)]，以 $\hat{J}$ 终止的 ℓ 步前瞻等价于以 $T^{\ell-1}\hat{J}$ 终止的一步前瞻算法。

值空间近似方法的类型

在第 2 章介绍了多种有限前瞻的方案，其中涉及的 $\tilde{J}$ 可以通过多种方法获得，例如问题近似和策略前展等。其中的一些方法经过适当修改就可以用在无穷阶段问题中，见图 5.1.1。例如，2.3 节介绍的问题近似方法就能够直接拓展到无穷阶段问题中，此时式 (5.5)中的函数 $\tilde{J}(j)$ 可以通过精确求解一个与原问题相关的无穷阶段（甚至是有限阶段）问题得到。聚集是另一类可行的近似方法，第 6 章将介绍此方法。

当考虑无穷阶段问题时，不完备状态信息会带来特殊的挑战。我们可以将此类问题重新表述为涉及置信状态的完整状态信息问题，但是此时状态空间就变成无限维（参见 1.3.6 节）。[①]当求解此类问题时，基于某种形式的确定性等价的问题近似尤为相关，这是因为通常都很容易得到一个近似问题。例如，前瞻函数 $\tilde{J}$ 可以通过求解对应于原问题的完备状态信息问题得到，此时系统状态的估

① 由于置信空间是无限维的，我们需要将第 4 章的理论进行扩展才能适用于此类问题。这是因为第 4 章的理论针对的是有限状态空间的问题。通常对于折扣问题来说，这种扩展是相当简单的，但对于随机最短路径问题则非如此。本书不会对相关内容作进一步讲解。

计值就被当作真实值加以使用。相应的完整状态信息问题有可能是确定性的，或者只涉及了数量不算太多的状态，故而可以求解。

本章的 5.1∼5.5 节所关注的值空间近似方法主要是基于近似策略迭代的思想得到的。具体来说，给定某初始策略 μ^0，这些算法通常有如下步骤：

（a）生成一系列策略 $\mu^0, \mu^1, \cdots, \mu^m$。

（b）对于每个策略 μ^k 执行近似策略评价，从而得到费用函数 $\tilde{J}_{\mu^k}$。评价过程可能涉及采用参数化近似架构/神经网络等方法，也可能用到截短策略前展。

（c）基于 $\tilde{J}_{\mu^k}$ 通过一步或多步策略改进得到下一个策略 μ^{k+1}。

（d）在所生成的一系列策略中，最后一个策略的近似评价函数 $\tilde{J}_{\mu^m}$ 就可以作为前瞻近似 $\tilde{J}$ 用于一步前瞻最小化式 (5.5)，或者相对应的多步前瞻最小化中。

我们会将策略前展视为近似策略迭代的一种简单变形，即在仿真的辅助下涉及一步策略迭代的方法。该方法中的前展仿真轨迹可以截短并辅以（可能非常复杂的）终止费用函数近似（参见 2.4 节）。5.1.1 节和 5.2.1 节将分别给出有限前瞻方案和策略前展的性能界。5.1.3 节则将讨论一般的近似策略迭代方法的性能界。

5.1.1 有限前瞻

现在我们考虑 ℓ 步前瞻方法的性能界。具体来说，对于给定状态 i_0，ℓ 步前瞻最小化问题

$$\min_{\mu_0,\cdots,\mu_{\ell-1}} E\left\{\sum_{k=0}^{\ell-1}\alpha^k g\big(i_k,\mu_k(i_k),i_{k+1}\big)+\alpha^\ell \tilde{J}(i_\ell)\right\}$$

的最优策略记为 $\hat{\mu}_0,\cdots,\hat{\mu}_{\ell-1}$。我们着重考虑定义为 $\tilde{\mu}(i_0)=\hat{\mu}_0(i_0)$ 的次优策略，并将 $\tilde{\mu}$ 称为**对应于 $\tilde{J}$ 的 ℓ 步前瞻策略**。如果采用式 (5.3)和式(5.4)中介绍的贝尔曼算子 T 和 T_μ，该策略可以用更为简短的表达式等效表述为

$$T_{\tilde{\mu}}(T^{\ell-1}\tilde{J})=T^\ell\tilde{J}$$

我们将在下列命题（a）中给出策略 $\tilde{\mu}$ 的性能界，它的证明则在本章附录中给出。

此外，我们还会推导出一个拓展的一步前瞻方法的性能界 [即以下命题（b）的部分]。通过针对子集 $\overline{U}(i)\subset U(i)$ 执行前瞻最小化，该方法旨在减小求取 $\tilde{\mu}(i)$ 时的运算量。因此在该拓展方法中，$\tilde{\mu}(i)$ 是取得表达式

$$\min_{u\in\overline{U}(i)} p_{ij}(u)+\big(g(i,u,j)+\alpha\tilde{J}(j)\big)$$

最小值的控制。如果通过某些启发式方法，我们能够识别出有希望包含原一步前瞻的最优解的控制子集 $\overline{U}(i)$，那么为了减少运算量，在相应的一步前瞻最小化中，就可以把需要考虑的控制局限在这个子集中。

命题 5.1.1（有限前瞻的性能界） （a）令 $\tilde{\mu}$ 为对应于 $\tilde{J}$ 的 ℓ 步前瞻策略。那么，

$$\|J_{\tilde{\mu}}-J^*\|\leqslant\frac{2\alpha^\ell}{1-\alpha}\|\tilde{J}-J^*\| \tag{5.6}$$

其中，$\|\cdot\|$ 表示最大范数 $\|J\|=\max\limits_{i=1,\cdots,n}|J(i)|$。

（b）定义

$$\hat{J}(i)=\min_{u\in\overline{U}(i)}\sum_{i=1}^{n}p_{ij}(u)\big(g(i,u,j)+\alpha\tilde{J}(j)\big),\quad i=1,\cdots,n \tag{5.7}$$

其中，$\overline{U}(i)\subset U(i)$ 对所有 $i=1,\cdots,n$ 都成立。令 $\tilde{\mu}$ 为通过最小化该式右侧所得的一步前瞻最小化策略，那么，

$$J_{\tilde{\mu}}(i)\leqslant\tilde{J}(i)+\frac{c}{1-\alpha},\quad i=1,\cdots,n \tag{5.8}$$

成立，其中，

$$c=\max_{i=1,\cdots,n}\big(\hat{J}(i)-\tilde{J}(i)\big)$$

关于性能界 [式(5.6)]，值得关注的一点是如果对 $\tilde{J}$ 作常数平移 [即在所有 $\tilde{J}(j)$ 的基础上添加常数 β]，那么相应的 $\tilde{\mu}$ 并不会受到影响。因此，式(5.6)中的 $\|\tilde{J}-J^*\|$ 可以用值

$$\min_{\beta\in\Re}\max_{i=1,\cdots,n}\big|\tilde{J}(i)+\beta-J^*(i)\big| \tag{5.9}$$

且后者的取值更小。另外一个有趣的点是上述的表达式中，只需要针对经过 ℓ 步前瞻后有可能达到的状态 i 进行最大化运算，从而有可能进一步改善性能界 [式(5.6)]。此方法也可用于得到对应于命题 5.1.1（a），以及后续相关的命题 5.1.3（a）的更好的性能界，不过本书不会对此作更进一步的分析。

性能界 [式(5.6)] 似乎表明性能会随着长度 ℓ 的增加而改善。类似的情况似乎会在前瞻费用近似 $\tilde{J}$ 靠近 J^*（当经过最优的常数平移 β 调整后）时发生。这两个结论都很直观，并且与我们的实践经验相符。值得注意的是，我们并没有证明多步前瞻最小化所得策略的性能一定优于通过一步前瞻得到的策略；前面章节中已经提及这并不一定成立（参见例 2.2.1）。此处所证明的采用多步前瞻会带来性能界（bound）的提高。

性能界 [式(5.8)] 表明，当 $c\leqslant 0$ 时，一步前瞻策略的费用 $J_{\tilde{\mu}}$ 不会大于 $\tilde{J}$。当 $c\leqslant 0$ 时，这就等价于 $\hat{J}\leqslant\tilde{J}$ 成立，而这与确定性策略前展方法中所提及的顺序提升属性类似（参见 2.4.1 节）。当对于某策略 μ，等式 $\tilde{J}=J_\mu$ 成立，且对于所有状态 i，满足 $\mu(i)\in\overline{U}(i)$（5.1.2 节中介绍的策略前展的精确形式将会假设这些条件成立），那么 $c\leqslant 0$，而且由式 (5.8)可知费用改进在此时成立，即 $J_{\tilde{\mu}}\leqslant J_\mu$。

不幸的是，当 α 接近 1 时，性能界 [式(5.6)] 并不令人放心。尽管如此，接下来的例子将表明即使在只涉及两个状态的简单问题中，上述的性能界也可以是紧的。该例中，单一阶段的两个控制所对应的阶段费用之差为 $O(\epsilon)$。该差值所带来的相应的策略费用之差为 $O\big(\epsilon/(1-\alpha)\big)$（通过累积无穷多阶段的折扣费用）。然而，在贝尔曼方程中，这些差值可能会被 J^* 和 $\tilde{J}$ 之间幅度为 $O(\epsilon)$ 的差值而“抹平”。

例 5.1.1 考虑如图 5.1.2所示的涉及两个状态的折扣问题，其中 ϵ 为一个正的常数，$\alpha\in[0,1)$ 为折扣因子。当处于状态 1 时有两个控制选择：前往状态 2 并花费 0（策略 μ^*）或者留在状态 1 并花费 $2\alpha\epsilon$（策略 μ）。最优策略是 μ^*，且最优展望费用函数是 $J^*(1)=J^*(2)=0$。现考虑费用函数近似 $\tilde{J}$

$$\tilde{J}(1)=-\epsilon,\quad \tilde{J}(2)=\epsilon$$

从而有

$$\|\tilde{J}-J^*\|=\epsilon$$

选择留在状态 1 的策略 μ 正是基于 $\tilde{J}$ 的一步前瞻策略，这是因为

$$2\alpha\epsilon + \alpha\tilde{J}(1) = \alpha\epsilon = 0 + \alpha\tilde{J}(2)$$

而且等式

$$J_\mu(1) = \frac{2\alpha\epsilon}{1-\alpha} = \frac{2\alpha}{1-\alpha}\|\tilde{J} - J^*\|$$

成立，从而可见式 (5.6)中给出的性能界在 $\ell = 1$ 时等号成立。

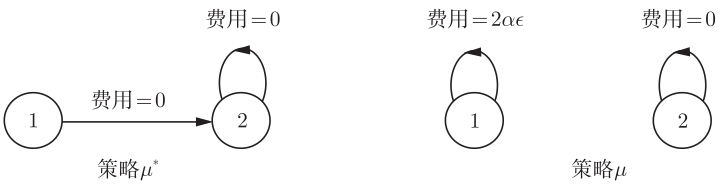

图 5.1.2　涉及两个状态的问题，用于说明命题 5.1.1（b）给出的性能界是紧的（参见例 5.1.1）。如图所示，所有转移都是确定性的。当处于状态 1 时有两个控制选择：前往状态 2 并花费 0（策略 μ^*）或者留在状态 1 并花费 $2\alpha\epsilon$（策略 μ）。

5.1.2　策略前展

首先考虑精确形式的策略前展，此时式 (5.5)中的函数 $\tilde{J}$ 即为某个稳态策略 μ [也称为基本策略（base policy）或基本启发式方法（base heuristics）] 的费用函数，即 $\tilde{J} = J_\mu$。那么，前展策略就是从 μ 出发、经过一步策略迭代得到的策略。执行策略改进需要一些费用函数 $J_\mu(j)$ 的值，求解这些值的策略评价可以通过任意适当方法来完成。蒙特卡洛仿真（对从 j 出发的许多轨迹的费用作平均）是其中的主要手段。当然如果问题是确定性的，那么获得从 j 出发的单一轨迹就足够了，此时策略前展所需的计算量就小了很多。此外，在求解折扣问题时，当仿真的步数足够大以至于折扣后剩余转移费用可以忽略不计时，可以将仿真轨迹截短。

另外一个重要的事实是，当采用策略前展的精确形式时，前展策略优于基本策略，这与有限阶段问题中的情况一致，参见 2.4 节。该结论由下面的命题给出 [如前面所提及的，此命题可以被视为命题 5.1.1（b）的特殊情况]，并且也符合我们的直觉，因为策略前展本身就是一步策略迭代，那么策略迭代算法中的策略改进这一一般属性在此情况下也成立。与此相关的一个结果是附录中的引理 5.9.1（5.9.3 节）。

命题 5.1.2（策略前展的费用改进）　令 $\tilde{\mu}$ 表示通过一步前瞻最小化

$$\min_{u\in\overline{U}(i)} \sum_{i=1}^{n} p_{ij}(u)\big(g(i,u,j) + \alpha J_\mu\big)$$

所得的前展策略，其中，μ 为基本策略 [参考式 5.7并令 $\tilde{J} = J_\mu$] 且假设 $\mu(i) \in \overline{U}(i) \subset U(i)$ 对所有 $i = 1, \cdots, n$ 都成立，那么 $J_{\tilde{\mu}} \leqslant J_\mu$。

接下来介绍策略前展的另外一个变体，该方法涉及多个基本启发式方法，并且所得策略优于所有已有的启发式方法。鉴于该变体具有显而易见的并行执行的潜质，它也被称为并行策略前展 (parallel rollout)。它与适用于有限阶段问题的相应方法类似，参见 2.4.1 节。

例 5.1.2（涉及多个启发式方法的策略前展） 令 $\mu_1,\cdots,\mu_M$ 表示多个稳态策略，并记

$$\tilde{J}(i)=\min\big\{J_{\mu_i}(i),\cdots,J_{\mu_M}(i)\big\},\quad i=1,\cdots,n$$

以及 $\overline{U}(i)\subset U(i)$，并假设

$$\mu_1(i),\cdots,\mu_M(i)\in\overline{U}(i),\quad i=1,\cdots,n$$

那么对于所有的 i 以及 $m=1,\cdots,M$，有

$$\begin{aligned}\hat{J}(i)&=\min_{u\in\overline{U}(i)}\sum_{i=1}^{n}p_{ij}(u)\big(g(i,u,j)+\alpha\tilde{J}(j)\big)\\&\leqslant\min_{u\in\overline{U}(i)}\sum_{i=1}^{n}p_{ij}(u)\big(g(i,u,j)+\alpha J_{\mu_m}(j)\big)\\&\leqslant\sum_{j=1}^{n}p_{ij}\big(\mu_m(i)\big)\Big(g\big(i,\mu_m(i),j\big)+\alpha J_{\mu_m}(j)\Big)\\&\leqslant J_{\mu_m}(i)\end{aligned}$$

对不等式右侧取关于 m 的最小值就得到

$$\hat{J}(i)\leqslant\tilde{J}(i),\quad i=1,\cdots,n$$

由命题 5.1.1（b）可知，采用 $\tilde{J}$ 作为一步前瞻近似得到的前展策略 $\tilde{\mu}$ 满足

$$J_{\tilde{\mu}}(i)\leqslant\tilde{J}(i)=\min\big\{J_{\mu_i}(i),\cdots,J_{\mu_M}(i)\big\},\quad i=1,\cdots,n$$

即前展策略优于所有给定策略 $\mu_1,\cdots,\mu_M$。

含多步前瞻和终止费用函数近似的截短策略前展

在接下来介绍策略前展方法的一类变体中，我们首先采用 ℓ 步前瞻，然后根据策略 μ 执行有限多步的仿真前展，并在仿真末端采用终止费用近似 $\tilde{J}$ 来代表其余阶段的费用，参见图 5.1.3中给出的 $\ell=2$ 时的架构。我们可以将此方法视为与多步前瞻相结合的乐观策略迭代的一步迭代（因为此处的策略评估是以 $\tilde{J}$ 为出发点，通过 m 步值迭代实现的，因此相应的策略迭代是乐观的）。该类算法也用于 Tesauro 的基于策略前展的双陆棋程序中 [TG96]（AlphaGo 程序也采用了该算法的一种变体，其中采用了蒙特卡洛树搜索来代替普通的有限前瞻）。后续会给出更多细节。

需要注意的是，策略前展架构中的各个组分（多步前瞻、基于 μ 的策略前展以及费用函数近似 $\tilde{J}$）可以通过各自独立的设计得到。此外，尽管多步前展是通过在线执行实现的，μ 和 $\tilde{J}$ 则需要提前通过先前的离线计算得到。

接下来的命题扩展了前面给出的适用于有限前瞻的性能界（参见命题 5.1.1）。具体来说，下列命题的（a）部分可以通过运用命题 5.1.1（a）得到，这是因为截短策略前展可以被视为涉及 ℓ 步前瞻的值空间近似方法，其在前瞻末端所用的终止费用近似是 $T_\mu^n\tilde{J}$，而 T_μ 则是对应于 μ 的贝尔曼算子。

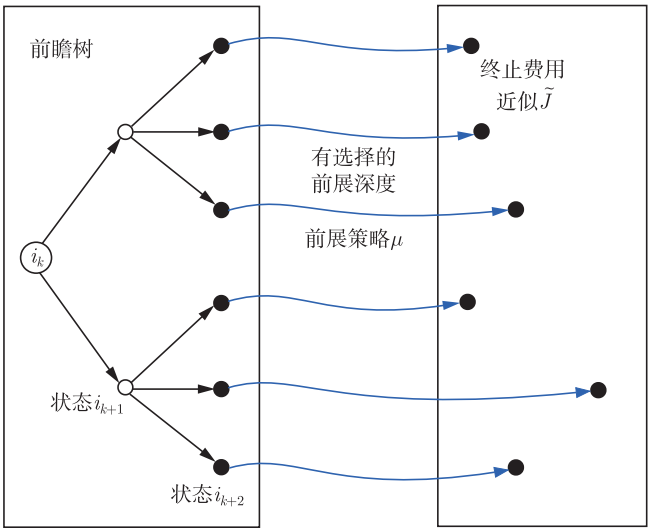

图 5.1.3 图示架构中采用了两步前瞻，然后根据策略 μ 执行随状态变化的有限多步的前展仿真，并在其后采用费用函数近似 $\tilde{J}$。蒙特卡洛树搜索方法也可以用于代替此处的多步前瞻，参见 2.4.2 节。

命题 5.1.3（含终止费用函数近似的截短策略前展的性能界） 令 ℓ 和 m 为某些正整数，μ 表示某一策略，并令 $\tilde{J}$ 表示关于状态的函数。现考虑某一截短策略前展方案，其中含有 ℓ 步前瞻，且在其后伴随 m 步针对策略 μ 的策略前展，并在 m 步仿真后采用费用函数近似 $\tilde{J}$。我们将通过此方案得到的策略记为 $\tilde{\mu}$。

（a）可知

$$\|J_{\tilde{\mu}} - J^*\| \leqslant \frac{2\alpha^{\ell}}{1-\alpha}\|T_{\mu}^{m}\tilde{J} - J^*\| \tag{5.10}$$

成立，其中 T_{μ} 是式 (5.4)定义的贝尔曼算子，$\|\cdot\|$ 表示最大范数 $\|J\| = \max\limits_{i=1,\cdots,n}|J(i)|$。

（b）可知

$$J_{\tilde{\mu}}(i) \leqslant \tilde{J}(i) + \frac{c}{1-\alpha}, \quad i = 1, \cdots, n$$

成立，其中，

$$c = \max_{i=1,\cdots,n}\left((T_{\mu}\tilde{J})(i) - \tilde{J}(i)\right)$$

（c）有

$$J_{\tilde{\mu}}(i) \leqslant J_{\mu}(i) + \frac{2}{1-\alpha}\|\tilde{J} - J_{\mu}\|, \quad i = 1, \cdots, n$$

成立。

根据上述命题，可以得出一些有用的结论：

（1）命题的（a）部分说明随着前瞻步数 ℓ 的增大，前展策略的性能的界也得到改善。此外，如果 μ 是近乎最优的（从而当 $m \to \infty$ 时，$T_{\mu}^{m}\tilde{J}$ 也会靠近 J^*），前展策略 $\tilde{\mu}$ 的性能界也随着 m 的增大而改善（在此基础上如果 $\tilde{J}$ 接近 J_{μ}，则会给性能界带来进一步的提高）。[①]

① 对 $\tilde{J}$ 进行常数平移并不会影响生成的策略 $\tilde{\mu}$。因此，在性能界式(5.10)中可以加入一个经过优化得到的常数 β [参见式 (5.9)]。

（2）命题的（c）部分表明如果 $\tilde{J}$ 接近 J_μ，那么相对基本策略 μ，前展策略 $\tilde{\mu}$ 的性能几乎会得到改进。这与策略前展方法中的费用改进属性一致，参见命题 5.1.2。对上述命题（b）部分的解读也是类似的。

总之，由上述结果总结得出的涉及截短策略前展的指南是选取实际允许的尽可能大的前瞻步数，并且使 $\tilde{J}$ 尽可能接近 J_μ 或 J^*。目前尚不清楚把对应于 μ 执行的前展的步数 m 增大到什么程度能够改善该方法的性能，而有些例子表明 m 的取值不宜过大。此外，较小的 m 值意味着相应的策略前展算法所需的计算量较小，并且会降低估计费用时的方差。在实践中，当求解无穷阶段问题时，人们通常经过某种经验方法选取 m。

至于截短策略前展中所涉及的终止费用近似 $\tilde{J}$，它可以是根据启发式方法得到的，也可能是基于问题近似或更加系统性的仿真方法算出的。例如，我们可以首先找出状态空间中一些具有代表性的状态 i，然后通过仿真算出这些状态的费用函数值 $J_\mu(i)$。在此基础上，采用最小二乘回归方法就能从给定的某一个参数向量组中选出合适的参数向量，从而得到 $\tilde{J}$。计算 $\tilde{J}$ 的过程可以通过离线方式实现，故不受在线执行任务的实时性要求的影响。由此得到的 $\tilde{J}$ 可以在在线选取控制时取代 J_μ 作为终止费用函数的近似。需要注意的是，在随机最短路径问题中，当绝大多数甚至全部的费用都产生于抵达终止状态的那一步（例如游戏是赢了还是输了），那么一个好的终止费用近似就极为关键。此外，一旦在策略前展的末端引入了终止费用近似，那么前展策略的费用改进属性就不能保证成立 [参见命题 5.1.3（c）]。

由 Tesauro 和 Galperin 在 [TG96] 中提出的策略前展双陆棋程序就采用了图 5.1.3的截短策略前展的架构。其中，策略 μ 和终止费用函数近似 $\tilde{J}$ 由 Tesauro 在 [Tes94] 中介绍的 TD-Gammon 算法提供。根据该方法，策略 μ 和终止费用函数近似 $\tilde{J}$ 均是基于神经网络得到的，其相应的训练方法是某种形式的乐观策略迭代以及 TD(λ)。AlphaGo 程序（见 [SHM+16]）也采用了相似的算法（不过 ℓ 步前瞻被蒙特卡洛树搜索所代替），其中的 μ 和 $\tilde{J}$ 是通过使用近似策略迭代方法和深度神经网络得到的。

5.1.3 近似策略迭代

当问题中涉及的状态的数目过大时，策略迭代算法中的策略评价和/或策略改进步骤可能只能通过近似方法执行。在这类近似策略迭代方法中，对于每个策略 μ^k 的评价都是近似的，且相应的近似费用函数 $\tilde{J}_{\mu^k}$ 常常会用到基于特征的近似架构或者神经网络。后续策略 μ^{k+1} 则是在函数 $\tilde{J}_{\mu^k}$ 的基础上，通过（或许是近似的）策略改进得到的。

接下来我们给出此类方法的严格数学表述。假设策略评价的误差满足不等式

$$\max_{i=1,\cdots,n} \left|\tilde{J}_{\mu^k} - J_{\mu^k}\right| \leqslant \delta \tag{5.11}$$

并且每一步策略改进的误差也满足

$$\max_{i=1,\cdots,n} \left| \sum_{j=1}^{n} p_{ij}\big(\mu^{k+1}(i)\big)\Big(g\big(i,\mu^{k+1}(i),j\big) + \alpha\tilde{J}_{\mu^k}(j)\Big) - \min_{u\in U(i)} \sum_{i=1}^{n} p_{ij}(u)\big(g(i,u,j) + \alpha\tilde{J}_{\mu^k}(j)\big)\right| \leqslant \epsilon \tag{5.12}$$

其中，δ 和 ϵ 是某些非负标量。此处的 δ 既包含了仿真误差，也包含了采用函数近似带来的误差。常数 ϵ 表示在策略改进步骤中执行前瞻最小化的精确程度（在许多情况下 $\epsilon = 0$）。

下列命题给出了该方法用于折扣问题时的性能界，其证明在附录中给出（此结论最早的出处及证明是 [BT96] 的 6.2.2 节）。适用于随机最短路径的相似结论可查阅 [BT96] 的 6.2.2 节。

命题 5.1.4（近似策略迭代的性能界） *考虑折扣问题，并且用 $\{\mu^k\}$ 表示通过以近似策略评价式(5.11)和近似策略改进式(5.12)定义的近似策略迭代算法生成的策略序列。那么策略误差*

$$\max_{i=1,\cdots,n} \left|J_{\mu^k}(i) - J^*(i)\right|$$

随着 $k \to \infty$，渐近地小于或等于

$$\frac{\epsilon + 2\alpha\delta}{(1-\alpha)^2}$$

从定性角度看，上述关于性能界的结论与近似策略迭代的实践经验相符，因此非常重要。一般来说，在前期几步迭代中，该算法倾向于产生快速且相对单调的改进，但最终都会产生振荡现象。当函数 J_{μ^k} 进入宽度不大于

$$\frac{\epsilon + 2\alpha\delta}{(1-\alpha)^2}$$

的误差区域后，振荡现象就会出现，且振荡行为相当随机，参见图 5.1.4。与实际相比，命题 5.1.4给出的误差界较为悲观。通常振荡区域的宽度比该性能界给出的宽度要窄得多。然而可以证明，该性能界是紧的。[BT96] 的 6.2.3 节通过一个例子对此加以说明。此外还要注意到，命题 5.1.4中的界对于涉及无穷多状态和控制，因此有无穷多策略的折扣问题同样适用（见 [Ber18a] 中的命题 2.4.3）。

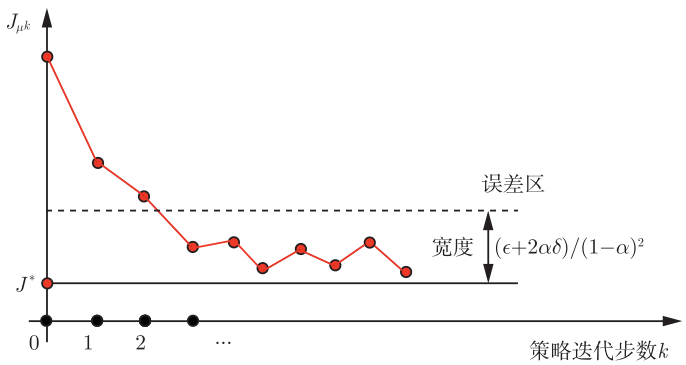

图 5.1.4 近似策略迭代算法的典型表现。在早期的几步迭代中，该方法倾向于给出较大的且相对单调的进展。当函数 J_{μ^k} 进入宽度小于或等于

$$\frac{\epsilon + 2\alpha\delta}{(1-\alpha)^2}$$

误差区域后，上述进展终止。此后函数 J_{μ^k} 将在误差区域内随机振荡。该图片是对实际情况的过度简化，因为在图中只给出了 $J_{\mu^k} - J^*$ 在单一状态的差值。对于不同的状态，其对应误差的振荡形式可能会不同。

策略收敛时的性能界

如前文所述，一般而言通过近似策略迭代生成的策略序列 $\{\mu^k\}$ 最终会在几个策略间振荡。但是在某些情况下，该序列会收敛到某策略 $\tilde{\mu}$，即

$$\mu^{\overline{k}+1} = \mu^{\overline{k}} = \tilde{\mu} \quad \text{对于某个} \overline{k} \tag{5.13}$$

成立。当采用聚集方法求解时，上述情况就会出现，详细内容见第 6 章。当出现策略收敛的现象时，我们可以得到比命题 5.1.4更好的性能界，且原有界与新界的比值为 $1/(1-\alpha)$，如图 5.1.5所示。下列命题给出了该性能界，其证明可见本章附录（或者该命题的原始出处 [BT96] 的 6.2.2 节）。

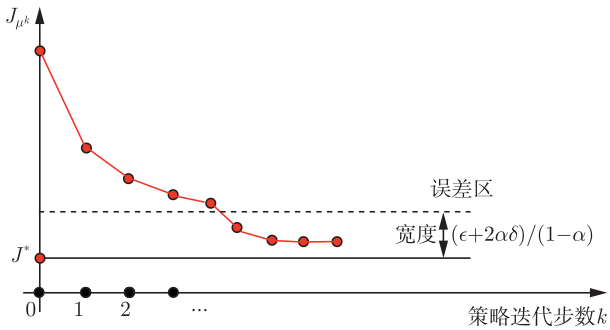

图 5.1.5 近似策略迭代算法在策略收敛时的典型表现。该方法倾向于给出相对单调的进展，且函数 J_{μ^k} 在一个误差宽度小于 $(\epsilon+2\alpha\delta)/(1-\alpha)$ 的区域内收敛。

命题 5.1.5（近似策略迭代在策略收敛时的性能界） 令 $\tilde{\mu}$ 表示采用近似策略迭代算法在式(5.11)、式(5.12)和式(5.13)条件下得到的策略。那么有

$$\max_{i=1,\cdots,n} \left|J_{\tilde{\mu}}(i) - J^*(i)\right| \leqslant \frac{\epsilon + 2\alpha\delta}{1-\alpha}$$

最后我们指出，针对乐观策略迭代，即策略评价仅通过几步值迭代来执行且策略改进是近似完成的方法（见 4.6.2 节），我们也可以得到与上述结论相关的性能界。这些性能界与适用于非乐观策略迭代的界类似，而且并不能说明某一类的策略迭代算法比其他的强。这些界的推导相当复杂，感兴趣的读者可查阅 [Ber12a] 的第 2 章或 [Ber18a] 的 2.5.2 节，以及本章末罗列的文献。

5.2 拟合值迭代

在第 4 章中，我们讲解了适用于随机最短路径问题的值迭代算法

$$J_{k+1}(i) = \min_{u\in U(i)} \left[p_{it}(u)g(i,u,t) + \sum_{j=1}^{n} p_{ij}(u)\big(g(i,u,j) + J_k(j)\big)\right] \tag{5.14}$$

以及相应的折扣问题的版本

$$J_{k+1}(i) = \min_{u\in U(i)} \sum_{j=1}^{n} p_{ij}(u)\big(g(i,u,j) + \alpha J_k(j)\big) \tag{5.15}$$