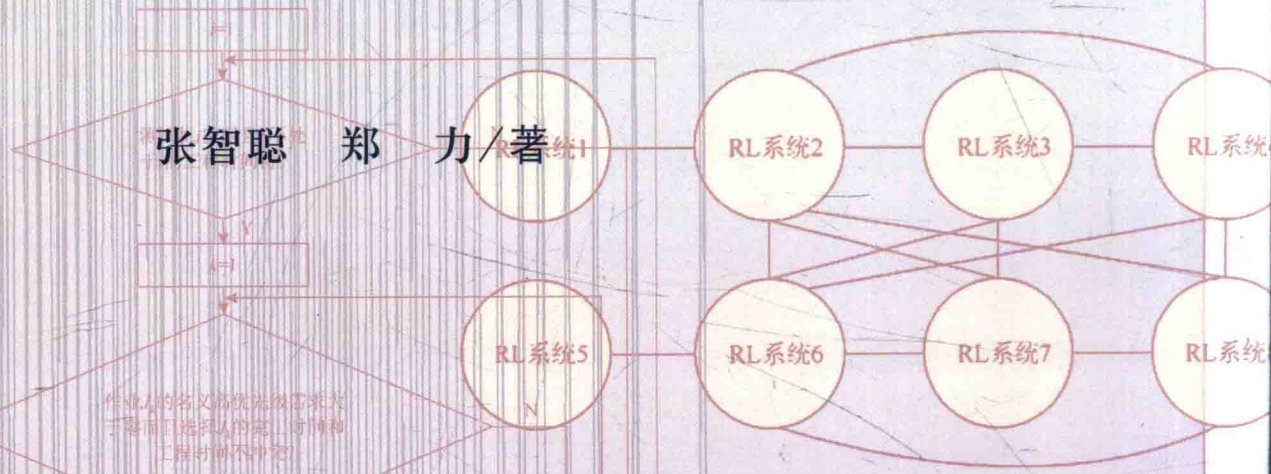
基于增强学习的制造系统调度

张智聪　郑　力/著



科学出版社

广东科学技术学术专著项目资金资助出版

基于增强学习的制造系统调度

张智聪　郑　力/著

科学出版社
北京

内 容 简 介

增强学习是人工智能领域一种应用越来越广泛的机器学习算法。本书对增强学习的基本原理、主要经典算法及其在制造系统调度领域若干问题的应用进行阐述。主要内容包括：Sarsa（λ, k）增强学习算法等增强学习算法的介绍及相关理论证明；增强学习架构及面向生产调度问题的增强学习模型构建方式；流水车间调度问题、平行机调度问题、半导体测试调度问题等制造系统调度问题与自组织型排队网络调度问题的增强学习模型及解决方案；增强学习在以上调度问题应用的实验结果及相关分析等。

本书适合管理科学与工程、工业工程等专业的研究生和本科生使用，也可供从事制造系统分析与优化、智能调度等领域工作的研究人员和工程技术人员参考。

图书在版编目（CIP）数据

基于增强学习的制造系统调度/张智聪，郑力著. —北京：科学出版社，2016.6

ISBN 978-7-03-049289-0

Ⅰ. ①基… Ⅱ. ①张… ②郑… Ⅲ. ①机器学习–应用–柔性制造系统–研究 Ⅳ. ①TH165-39

中国版本图书馆 CIP 数据核字（2016）第 146841 号

责任编辑：郭勇斌 肖 雷 邓新平 / 责任校对：王晓茜
责任印制：张 伟 / 封面设计：黄华斌

科 学 出 版 社 出版
北京东黄城根北街 16 号
邮政编码：100717
http：//www.sciencep.com

北京教图印刷有限公司印刷
科学出版社发行 各地新华书店经销
*
2016 年 6 月第 一 版 开本：787×1092 1/16
2016 年 6 月第一次印刷 印张：14 1/2
字数：344 000

定价：88.00 元

（如有印装质量问题，我社负责调换）

本书主要研究内容得到国家自然科学基金项目（71201026）、广东省自然科学基金项目（2015A030313649）、广东省科技计划项目公益研究与能力建设专项（2015A010103021）、广东省高等学校优秀青年教师培养计划（Yq2013156）、广东省教育厅 2015 年重点平台及科研项目特色创新类项目（自然科学类）（2015KTSCX137）资助。

前　言

增强学习（又称为强化学习、激励学习、再励学习）是一种可以求解大规模马尔可夫决策过程等序贯决策问题的机器学习算法，是目前机器学习领域研究的一个热点。该类算法刚提出时主要被计算机、自动化等领域的研究人员用于解决自动控制、机器人、人工智能等领域的问题。近年来，该类算法逐渐引起管理科学与工程等领域专家和企业界人士的重视，广泛应用于生产调度、库存控制、运输调度等制造与物流管理问题。

目前关于增强学习的专著多聚焦于增强学习算法理论本身或该类算法在自动控制等领域的应用，而本书主要阐述增强学习算法在制造系统调度领域的应用研究。本书的主题顺应智能制造等制造领域前沿研究的潮流，重点不在于介绍分类增强学习算法及其相关理论性质，因此并没有全面系统地介绍增强学习算法。本书的主要价值在于结合作者的研究经历介绍增强学习算法在制造业这一特定领域的应用，针对制造系统调度问题的特点，系统阐述应用增强学习算法解决制造系统调度问题的整体架构和应用流程，为解决制造系统调度相关问题提供一类可借鉴的方法和思维模式，希望能起到抛砖引玉的作用，启发相关研究人员的思路。如读者希望对增强学习算法及其相关理论进行系统了解，请查阅相关参考书。

本书是作者多年来从事国家自然科学基金项目（71201026）、广东省自然科学基金项目（2015A030313649）、广东省科技计划项目公益研究与能力建设专项（2015A010103021）、广东省高等学校优秀青年教师培养计划（Yq2013156）、广东省教育厅 2015 年重点平台及科研项目特色创新类项目（自然科学类）（2015KTSCX137）的研究成果的总结，在此特向国家自然科学基金委员会、广东省自然科学基金委员会、广东省科技厅、广东省教育厅表示衷心感谢！

由于作者水平和精力有限，对增强学习算法在制造系统调度领域的研究尚不够深入。书中的研究结果只是作者多年来在研究过程中把增强学习算法应用于若干调度问题获得的初步结果，离增强学习算法解决这些调度问题所能获得的最优结果尚有不小距离，由于各种原因没能进一步深入研究这些问题。书中难免有欠妥之处，敬请各位专家和读者批评指正。

作　者

2016 年 2 月于松山湖、清华园

目　录

第1章 绪 论

1.1 增强学习基本原理

增强学习（Reinforcement Learning，RL）又称为强化学习、激励学习、再励学习，是一种机器学习方法。增强学习可以解决很多类问题，序贯决策问题是其中一类应用很广泛的典型问题。动态规划方法是求解序贯决策问题的传统方法，但动态规划方法需要知道状态转移概率矩阵和报酬函数，而且每次迭代更新状态值时通常要对每个状态进行扫描或求解与状态个数相当的方程，因此，动态规划难以处理以下两类问题：状态转移概率矩阵、报酬未知或难以显式表示的问题；超大规模状态空间、无限状态空间或连续状态空间的问题，即具备“维数灾难”（Curse of Dimensionality）特点的问题。增强学习算法并不需要知道状态转移概率矩阵，不需要在迭代时对很多状态进行扫描，是解决这两类问题的有效方法。

由于马尔可夫决策过程（Markov Decison Processes，MDP）和半马尔可夫决策过程（Semi-Markov Decison Processes，SMDP）是典型的序贯决策问题，所以常用增强学习算法解决大规模的或转移概率未知的马尔可夫决策过程和半马尔可夫决策过程。需要说明的是，增强学习算法可以解决的问题不局限于这些问题。事实上，很多多阶段决策问题虽然不属于马尔可夫决策问题，其状态转移不严格具备马尔可夫属性，但仍可以用增强学习算法解决，并获得不错的效果。下面以有限状态和行为空间的马尔可夫决策过程为背景介绍增强学习算法的基本原理。

1.1.1 马尔可夫决策过程

马尔可夫决策过程可用如下五重组表示：

$$\{Z^+, S, A(s), p(s,a,s'), r(s,a,s')\}$$

式中，$Z^+ = \{0,1,2,\cdots\}$ 表示决策阶段的集合；S 表示状态空间；$A(s)$ 表示状态为 s（$s \in S$）时可以采取的行为的集合；A 表示行为空间；$p(s,a,s')$ 表示状态为 s 时采取行为 $a[a \in A(s)]$ 后状态转移到 s' 的概率；$r(s,a,s')$ 表示状态为 s 时采取行为 a 而下一状态为 s' 时获得的报酬，它是有界的报酬函数。

设 r_{t+k+1} 表示第 $t+k+1$ 个决策时刻获得的报酬，折扣型马尔可夫决策的目标是寻找最优策略 π^*，使任何状态 s 开始遵循该策略得到的期望总报酬 $E[R_t \mid s_t = s]$ 达到最大化。期望总报酬的定义如式（1.1）所示，其中 $\gamma(0 < \gamma \leqslant 1)$ 为折扣率。

$$E[R_t \mid s_t = s] = E\left[\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \mid s_t = s\right] \tag{1.1}$$

用$V^{\pi}(s)$表示从状态s开始遵循策略π获得的期望总报酬。$V^{\pi}(s)$（$s\in S$）为策略π下的状态值函数。用$Q^{\pi}(s,a)$表示从状态s开始，先采取行为a，然后遵循策略π获得的期望总报酬。$Q^{\pi}(s,a)$是策略π下的行为值函数。$Q^{\pi}(s,a)$和$V^{\pi}(s)$的关系如式（1.2）。

$$Q^{\pi}(s,a)=\sum_{s'\in S}p(s,a,s')[r(s,a,s')+\gamma V^{\pi}(s')] \tag{1.2}$$

式（1.3）是最优状态值的 Bellman 方程[1]，其中$V^{*}(s)$是最优状态值函数，即在最优策略π^{*}下的状态值函数。该方程描述了最优状态值及其后续状态值的关系。

$$V^{*}(s)=\max_{a}\sum_{s'}p(s,a,s')[r(s,a,s')+\gamma V^{*}(s')] \tag{1.3}$$

根据式（1.2）和式（1.3）可得式（1.4）对任意$s\in S$成立，其中，$Q^{*}(s,a)$是最优行为值函数，即在最优策略π^{*}下的行为值函数。

$$Q^{*}(s,a)=\sum_{s'\in S}p(s,a,s')[r(s,a,s')+\gamma\max_{a'\in A(s')}Q^{*}(s',a')] \tag{1.4}$$

对于平均报酬型马尔可夫决策过程，用$\rho^{\pi}(s)$（其定义见式（1.5））表示平均报酬型马尔可夫决策过程在策略π下状态s的平均报酬函数，其中，s_t表示第t个决策阶段的状态，$\pi(s_t)$表示策略π在状态s_t采取的行为。对于单链的马尔可夫决策过程，$\rho^{\pi}(s)=\rho^{\pi}(s')$对任意两个状态$s$和$s'$都成立，因此所有状态的平均报酬函数可统一用$\rho^{\pi}$表示。决策目标是寻找最优策略$\pi^{*}$使平均报酬最大化。

$$\rho^{\pi}(s)=\lim_{K\to\infty}\inf\frac{1}{K+1}E\left[\sum_{u=0}^{K}r(s_t,\pi(s_t),s_{t+1})\,|\,s_0=s\right] \tag{1.5}$$

对于无限阶段平均报酬型单链马尔可夫决策过程，如果状态和行为空间都是有限的，那么存在状态值函数$V^{*}(s)$和实数ρ^{*}使 Bellman 最优方程组（式（1.6））对任意状态s成立。可以证明，$V^{*}(s)$为最优状态值函数；ρ^{*}为可获得的最大平均报酬，即在最优策略π^{*}下获得的报酬$\rho^{\pi^{*}}$。如果求出$V^{*}(s)(s\in S)$，那么最优策略π^{*}就可以根据式（1.6）得到。

$$V^{*}(s)=\max_{a\in A(s)}\left\{\sum_{s'\in S}p(s,a,s')[r(s,a,s')+V^{*}(s')]-\rho^{*}\right\} \tag{1.6}$$

动态规划算法是求解马尔可夫决策过程的经典算法，包括策略迭代和值迭代两类基本方法。策略迭代包括策略评估和策略改善两个交替的过程；而值迭代通过迭代求最优状态或行为值函数，从而得到最优策略。传统的动态规划方法难以解决大规模状态空间的马尔可夫决策过程或状态转移概率信息不全的马尔可夫决策过程，而增强学习算法的提出为解决这些问题提供了新的途径。

1.1.2　增强学习系统

在增强学习系统中，智能体或代理（Agent）通过真实体验或仿真实验与环境进行交互，通过反复试错（Trial-and-Error）的方法感知、学习环境的特性，在不同的系统状态下尝试各种可行的行为，得到即时的报酬作为行为短期效果的评价，根据报酬信息调整策略，从而找到最优或较优的控制策略，使长期的累积报酬（Return）或平均报酬最大化。

对于折扣型或有限阶段决策问题，增强学习的目标是最大化累积报酬。为了防止累积报酬趋向无限大，通常用折扣型累积报酬形式。如不特别说明，本书中的增强学习算法通常是指累积报酬形式的增强学习算法，用于解决有终止状态的序贯决策问题。除了累积报酬形式的增强学习算法，另一类常用的增强学习算法是平均报酬增强学习算法。下面用累积报酬形式的增强学习说明增强学习系统的组成及其运作机制。

增强学习的基本要素包括状态、行为、策略、报酬函数和值函数等。状态描述系统环境的特征，包括整体特征和局部特征。状态空间是所有可能的状态所组成的集合。把智能体需要做出决策的状态称为决策状态。行为是在某个决策状态可以采取的决策（操作）。行为空间是在每个决策状态可以选择的行为所组成的集合。策略是各阶段的决策组成的序列，它确定了状态空间到行为空间的映射，指定了智能体在各个决策状态采取的行为。报酬又称为强化信号，是环境对智能体在决策状态所采取行为的反馈信息。报酬函数是决策状态和所采取行为的函数。值函数表征了从某状态起遵循一定的策略所获得的总报酬。分别用状态值 $V(s)$ 和行为值 $Q(s,a)$ 表示增强学习系统从状态 s 和状态-行为对 (s,a) 开始的累积报酬，它们表征状态 s 和状态-行为对 (s,a) 的“优劣”程度。系统的动态特性决定了状态的转移规律，智能体需要经历一段学习历程才能逐渐感知它。

智能体和环境的具体交互过程可用图 1.1 表示。在与环境交互的过程中，智能体根据控制策略 π（如 ε-贪婪策略）选择行为。智能体在某决策时刻（假设为第 t 个时刻）感知环境的状态 s_t，根据策略 π 选择并执行行为 a_t。环境评价行为 a_t 的效果，在下一时刻（假设为第 $t+1$ 个时刻）赋予智能体一笔时间延迟的报酬 r_{t+1}[①]，而环境状态也转移到 s_{t+1}。对于马尔可夫决策型增强学习问题，智能体与环境交互的过程看作马尔可夫决策过程，状态转移具备马尔可夫属性，状态值和行为值是当前状态的函数，由当前的状态和采取的行为可预测下一时刻的状态和报酬。智能体的目标是在整个决策过程中的累积报酬最大化。需要说明的是，为统一起见，本书所涉及的增强学习算法的表达形式均以报酬最大化作为目标，因此，如用这些增强学习算法解决最小化目标函数形式的问题，报酬函数可设计为负值。

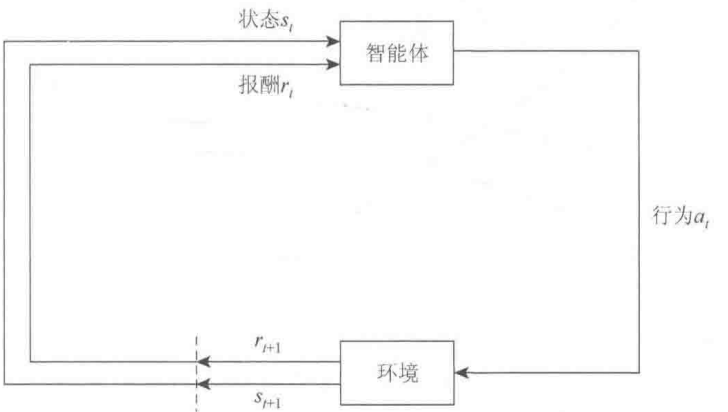


图 1.1 增强学习原理示意图

① 本节以时间延迟的报酬说明增强学习的原理，增强学习也可以处理报酬不延迟的问题。

增强学习算法的目的是随着智能体与环境交互过程的进行，使状态值函数$V(s)$逐渐逼近最优的状态值函数$V^*(s)$，或使行为值函数$Q(s,a)$逐渐逼近最优的行为值函数$Q^*(s,a)$，以期获得最优策略π^*。具体的增强学习算法介绍请看第 2 章及相关参考文献。增强学习算法在更新状态或行为值时，不一定穷举式的遍历每个状态或状态-行为对，而是沿着实际发生的或仿真产生的样本轨迹更新状态或行为值函数，这样，经历次数少的状态或状态-行为对的值的更新次数也少，从而大幅缩短求解问题的时间。在有限的运行时间内，增强学习算法优先保证学习到的策略对于频繁经历的状态的优化程度。换言之，在学习过程中，对于频繁经历的状态，增强学习算法学习到的行为的优化程度一般比较少经历状态的行为的优化程度高；对于频繁经历的状态，增强学习算法学习到的行为是较优行为甚至是最优行为，而对于较少经历的状态，增强学习算法学习到的行为却可能并非是很好的行为。对于表格式（Tabular）增强学习算法，如果所有的状态都经历足够多的次数，那么增强学习算法就能学习到所有状态的最优策略，即面对每一个状态都能选择最优行为。

1.1.3　增强学习算法的分类与发展概述

Sutton[2]提出瞬时差分（Temporal Difference，TD）算法 TD（λ）、Watkins[3]提出 Q 学习之后，增强学习算法的理论和应用研究逐渐发展为人工智能的一个重要分支，目前该领域的研究已成为机器学习领域的热点之一。如图 1.2 所示，从不同的角度可把增强学习算法分为不同的种类。

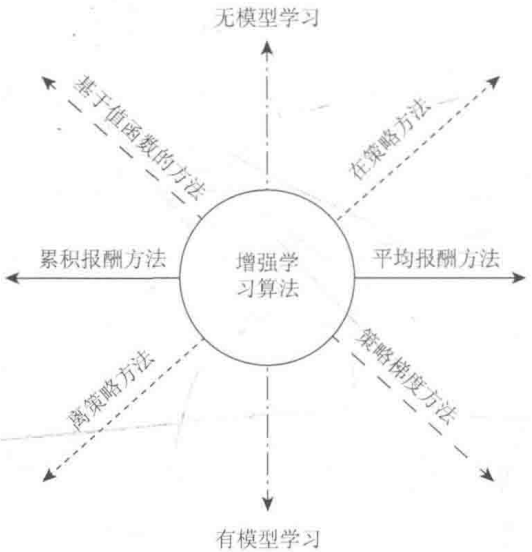


图 1.2　增强学习算法的分类

根据策略是否独立于状态或行为的表示值可把增强学习算法分为基于值函数的方法和基于策略的方法[4, 5]。基于值函数的方法是指在任意决策状态选择的行为基于当前增强学习系统表示的状态或行为值函数确定的方法。大多数算法是基于值函数的方法，如瞬时差分算法、Q 学习、Sarsa 算法[6]等。基于策略的方法是指采取的策略在增强学习系统内有专门的表示机制，与当前增强学习系统表示的状态或行为值函数无直接关系。Actor-critic

方法[7-11]是典型的基于策略的方法。

根据算法是否预测环境的模型，可把增强学习算法分为无模型（Model-free）学习和有模型（Model-based）学习。常用的增强学习算法多数属于无模型学习方法，如瞬时差分算法、Q学习、Sarsa算法等。有模型学习方法学习环境的模型并用函数泛化器表示，在每次迭代时不仅利用当前状态转移信息更新值函数，还利用预测的模型和以前经历过的状态转移信息更新值函数。有模型学习包括Dyna-Q[12]、Prioritized sweeping[13, 14]和H学习[15]等。

根据算法是否使用和遵循控制策略所产生的后续状态或状态-行为相同的分布更新（Backup）状态或行为值，可把增强学习算法分为在策略（On-policy）算法和离策略（Off-policy）算法。在策略算法学习的值函数是遵循控制策略所得的值函数，瞬时差分算法、Sarsa和Actor-critic等属于在策略算法。离策略算法学习的值函数并非遵循控制策略所得的值函数，Q学习是典型的离策略算法。文献[16]详细地介绍了增强学习算法原理和多种经典的增强学习算法。

根据解决的序贯决策问题属于累积报酬问题还是平均报酬问题，可把增强学习算法分为累积报酬型增强学习算法和平均报酬型增强学习算法。另外，增强学习算法常用于解决马尔可夫决策过程（MDP）和半马尔可夫决策过程（SMDP）这两类特殊的序贯决策问题，解决这两类问题的增强学习算法分别为MDP型增强学习算法和SMDP型增强学习算法。原始的瞬时差分、Q学习和Sarsa等算法都是针对累积报酬问题提出的。Sutton[2]提出了用于预测的TD（λ）算法，并证明了列表型TD（0）算法的收敛性。Dayan[17]证明了TD（λ）算法对任意的λ都以概率1收敛。Tsitsiklis和Van Roy[18]在P. Dayan的研究成果的基础上证明了与线性函数泛化器结合的在线TD（λ）算法在状态空间可数的马尔可夫链应用时的收敛性，并证明了其渐进误差的界。Tadić[19]进一步分析了与线性函数泛化器结合的TD（λ）算法在有限维的不可数状态空间的马尔可夫链应用时的收敛性质，并分析了渐进泛化误差的上界。Tsitsiklis和Van Roy[20]对比了结合线性函数泛化器的折扣报酬型TD（λ）算法和平均报酬型TD（λ）算法。

Watkins[3]提出另一种常用的增强学习算法——Q学习。Watkins[21]、Tsitsiklis[22]、Bertsekas和Tsitsiklis[23]对列表型Q学习的收敛性作了分析，证明$Q(s,a)$以概率1收敛到$Q^*(s,a)$。Mitchell[24]用简明的方法证明Q学习算法在确定性马尔可夫决策过程的收敛性。Potapov和Ali[25]研究了参数的取值范围对Q学习算法在确定性、随机性马尔可夫决策过程中的收敛性的影响。Peng和Williams[26]参考TD（λ）算法提出Q（λ）学习算法并给出其后视（Backward View）形式。Bradtke和Duff[27]给出解决SMDP问题的Q学习算法的具体步骤。另外，一些学者提出由Q学习衍生出来的算法[28, 29]。

Rummery和Niranjan[6]、Rummery[30]提出了Q学习的修正算法——Sarsa算法，并把TD（λ）算法的思想用于控制策略的学习，构造了Sarsa（λ）算法。Singh等[31]证明了在策略学习算法在GLIE（Greedy in the Limit with Infinite Exploration）和RRR（Restricted Rank-based Randomized）控制策略下，列表型单步Sarsa算法对于有限的状态和行为空间的随机马尔可夫决策过程是收敛的。Sarsa（λ）算法需要存储每个状态-行为对的适合迹（Eligibility Trace），每次迭代时对其进行更新，当状态空间较大时算法运行速度很慢。为了提高算法效率，文献[32]提出遗忘Sarsa（λ）算法，当某个状态-行为对的适合迹小于某

个阈值时，把该状态-行为对的适合迹和 Q 值都遗忘掉。文献[33]～[35]分别提出多步截断的 TD（λ）、Sarsa（λ）和 Q（λ）算法。TD（λ）、Sarsa（λ）和 Q（λ）的学习目标是通过无穷多个 n（$n \geq 1$）步更新目标的加权总和更新状态值函数和行为值函数，而截断算法只通过前 k 个 n（$1 \leqslant n \leqslant k$）步更新目标的加权总和更新值函数，将最后一项（$k$ 步更新目标）的权重由 $(1-\lambda)\lambda^{k-1}$ 增大到 λ^{k-1} 以弥补被截去的项的信息，λ 较大时第 k 项的权重也较大。当 λ 较大时截断 TD（λ）算法的效果较差[33]。

平均报酬型增强学习算法用于解决平均报酬型决策问题，主要包括 R 学习[36]、SMART[37]、Relaxed-SMART[38]及 Q-P 学习[39]等。

当状态空间的规模增大时，越来越难学习到较优的策略。递阶增强学习[40, 41]可将增强学习问题分解成多个层次的子问题，子问题的状态空间较小，通过求解各个子问题得到原问题的较优策略。增强学习算法也可以用在多智能体（Multi-agent）系统中以提高增强智能体的智能程度。基于多智能体的方法利用分布式人工智能的优点，通过各个分散的智能体之间的通信和协商解决问题。智能体有各自的目标和自治的行为，但没有一个智能体能够解决全局问题。智能体之间的协作效率越高，系统的敏捷性就越好，所以协商机制至关重要。多智能体系统中的增强学习算法要考虑利益分配、目标的协调和建议的采纳等问题[42-44]。另外，利用增强学习解决部分可观察马尔可夫决策过程（Partially Observable Markov Decision Processes，POMDPs）[45]，增强学习和遗传算法、模拟退火、蚁群算法等智能搜索算法的结合使用是近年的研究方向之一。表 1.1 总结了一部分增强学习算法。

1. “利用”和“探索”的平衡

权衡“利用”（Exploitation）和“探索”（Exploration）是应用增强学习算法要解决的一个重要问题。“利用”是指选择当前最优的行为，这样可充分利用已经学到的经验知识，短期内可得到较多的报酬；“探索”是指选择非当前最优的行为，探索新的行为（采取以前没有执行过的行为）。“探索”有可能提升该行为的值，从而发现更好的行为，从长远来看可能获得更多的回报，然而过多的探索又可能会削弱算法的效率和效果。另外，列表型增强学习算法收敛的技术条件要求每个状态或状态-行为对都经历足够多的次数，所以一般来说增强学习算法都持续的进行探索。平衡“利用”和“探索”的常用方法[16]有ε-贪婪方法、玻尔兹曼（Boltzmann）方法、增强对比（Reinforcement Comparison）和追赶法（Pursuit Method）等。

表 1.1　部分增强学习算法

研究者	增强学习算法	算法类型
Witten[7]，Barto 等[8]，Sutton[9]，Williams[10]，Borkar[11]	actor-critic	基于策略的方法
Sutton[2]	TD（λ）	用于预测的算法
Watkins[3]	Q 学习	离策略学习
Sutton[12]	Dyna-Q	有模型学习
More 和 Atkeson[13]，Peng 和 Williams[14]	Prioritized sweeping	有模型学习
Schwartz[36]	R 学习	平均报酬型增强学习

续表

研究者	增强学习算法	算法类型
Rummery 和 Niranjan[6]，Rummery[30]	Sarsa、Sarsa（λ）	在策略学习
Peng 和 Williams[26]	Q（λ）	离策略学习
Tadepalli 和 Ok[15]	H 学习	有模型学习
Dietterich[40]	MaxQ	递阶增强学习算法
Das 等[37]	SMART	平均报酬型增强学习
Gosavi[38]	Relaxed-SMART	平均报酬型增强学习
Gosavi[39]	Q-P 学习	平均报酬型增强学习

2. 处理大规模状态空间的方法

增强学习算法对值函数的评估是建立在对状态或行为的多次重复的基础之上，对于小规模状态空间问题，可以用列表显式表示每一个状态或行为的值，此时的增强学习算法称为列表型算法。但实际问题往往具有大规模或连续的状态空间，增强学习算法不可能遍历所有状态，很多状态不具有重复性，因此增强学习算法面临“维数灾难”问题。在这种情况下，状态或行为的值不能用列表显式表示。解决该问题的方法主要有两类：一类是基于离散化的方法，把连续状态空间离散化，或者把大规模空间通过压缩、分解等手段转化为小规模状态空间，增大状态空间划分的粒度，在减小问题规模、分解状态空间的同时尽可能保持原问题的结构与性质；另一类是值函数泛化（值函数逼近）方法。值函数泛化是一种降维处理的方法，它根据经历过的状态或行为值推测尚未经历过的状态或行为值，从而得到这些状态或行为值的近似表示。常用的值函数泛化器包括多元回归[15]、基于案例的方法[46, 47]、支持向量机[48]、基于核的（Kernel-based）方法[49]等。神经网络函数泛化器是一种较为常用的函数泛化器，它在学习过程中不断调整神经网络的权重，相当于不断改变状态或行为函数。一般采用梯度下降法调整神经网络的权重。Menache 等[50]提出使用交叉熵（Cross Entropy）的方法调整径向基函数神经网络的权重。支持向量机的学习性能和泛化能力与参数取值有关，有一些文献结合微粒群优化（PSO）机制对支持向量机的参数进行优化，寻找优化的参数取值组合。

1.2 增强学习算法应用引例——最短路问题

由于增强学习算法具有独特的特点和功能，因此获得越来越广泛的重视，近年来其应用领域不断扩大。本节以一类经典的最短路问题为例解释增强学习算法解决简单问题的应用过程。如图 1.3 所示，图中有 1～16 共 16 个节点（圆圈表示节点），两个节点之间如有连线则表示这两个节点之间存在通路（箭头表示路径的方向），连线旁边的数字表示这两个节点之间的路径长度。例如，节点 1 和节点 2 之间存在一条连线（箭头从节点 1 指向节

点 2），意味着从节点 1 可以直接走向节点 2，路径长度为 6。由于节点 1 和节点 4 之间不存在直接连线，因此不能从节点 1 直接走向节点 4。假设一辆车从节点 1 出发，要开往目的地节点 16，该问题的目的是求出从节点 1 到节点 16 的最短路径。该问题是一个典型的多阶段决策问题（共 6 个阶段），第一个阶段是车辆在节点 1，它要作出决策，选择下一步要走的路。车辆在节点 1 有两种选择：开往节点 2 或开往节点 3。第二个阶段是车辆在节点 2 或节点 3，无论车辆在节点 2，还是节点 3，它作出下一步决策时均有三种选择：如果车辆在节点 2，则可以选择开往节点 4、开往节点 5 或开往节点 6；如果车辆在节点 3，则可以选择开往节点 5、开往节点 6 或开往节点 7。依此类推，车辆无论在第三、第四、第五个阶段的哪个节点均有两种选择，而在第六个阶段，无论车辆在节点 14 还是节点 15 均只有一种选择：开往目的地——节点 16。由此可见，车辆从节点 1 开往节点 16 共有 2×3×2×2×2×1=48 条路径可以选择。求解最短路问题的目的是在这 48 条路径中寻找总长度最小的路径。

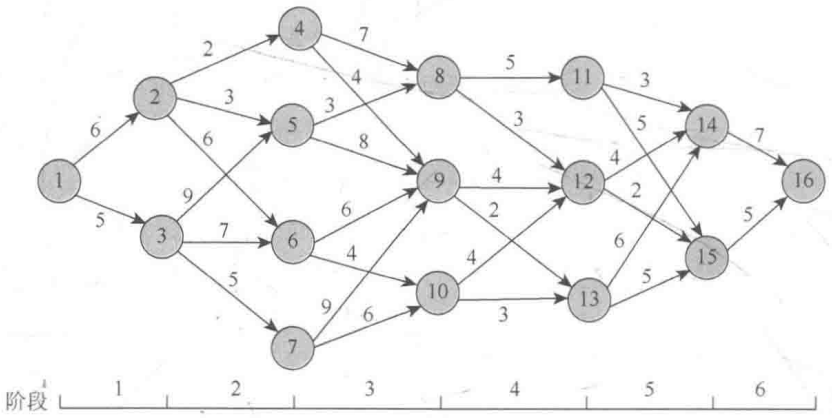


图 1.3　最短路问题

在运筹学教科书中，此类最短路问题通常用动态规划方法或图论的 Dijkstra（迪杰斯特拉或狄克斯特拉）标号算法求解。下面采用增强学习算法进行求解。在最短路问题中，状态变量 s 表示车辆当前所处的节点，状态空间 $S=\{1, 2, 3, \cdots, 16\}$，其中 $s=16$ 表示终止状态。行为表示车辆在某状态所选择的下一步的走向，例如，状态 $s=2$ 表示车辆当前在节点 2，在该节点车辆可以选择三个行为，分别是路线节点 2→节点 4、路线节点 2→节点 5、路线节点 2→节点 6。如果选择第 1 个行为，则状态转移后的下一个状态为 $s=4$（意味着车辆位于节点 4）。决策对应于车辆在一个阶段的行为选择，策略对应于车辆从节点 1 到节点 16 所经历六个阶段所选择行为组合成的整条路径，例如，“节点 1→节点 2→节点 4→节点 8→节点 11→节点 14→节点 16”对应于一个完整的策略。报酬的定义应和路线长度直接相关，由于最短路问题的目标函数是求最小化，而本书的增强学习算法的表达形式是针对最大化目标函数的形式，因此报酬定义为路线长度的相反数。例如，状态 $s=2$ 时如果车辆选择的行为是路线“节点 2→节点 4”，那么在此阶段获得的报酬是–2。由于值函数 $V(s)$ 反映的是从状态 s 开始到决策过程结束所获得的累积报酬，因此，在最短路问题中，值函数 $V(s)$ 反映的是从节点 s 开始到目

标节点（节点 16）的路径长度的信息。可采用瞬时差分（TD）增强学习算法求解最短路问题。由于最短路问题的决策过程有终止状态，因此，最短路问题是有限阶段的决策问题，TD 算法中的折扣因子 γ 可取 1。本例采用的 TD 算法步骤如下面的算法 1.1 所示，其中，num_Episode 表示 TD 算法的试验次数（重复解决该最短路问题的次数），一次“试验”表示 TD 算法解决一次最短路问题，产生一个解决方案（即找到一条从节点 1 到节点 16 的路径）。

算法 1.1 求解最短路问题的 TD 算法

步骤 1：设置学习率参数 α =0.1，初始化 num_Episode=0。对任意的 $s \in S$，初始化 $V(s)=1$。

步骤 2：设置当前状态 s_t 为初始状态 s_0 =1（表示开始时车辆位于节点 1）。

步骤 3：根据 s_t、状态值函数 $V(s)$ 和贪婪策略 π 选择行为 a_t。贪婪行为 a_t 定义如下：

$$a_t = \arg\max_{a \in A(s_t)} \{r_{t+1}^a + V(s_{t+1}^a)\} \tag{1.7}$$

其中，$A(s_t)$ 表示在状态 s_t 的可选行为集合，s_{t+1}^a 表示在状态 s_t 选择行为 a 后转移到的下一个状态，r_{t+1}^a 表示在状态 s_t 选择行为 a 后将获得的报酬。

步骤 4：执行行为 a_t，确定下一个决策状态 s_{t+1}，并获得报酬 r_{t+1}。

步骤 5：根据式（1.8）更新 s_t 的状态值 $V(s_t)$：

$$V(s_t) \leftarrow V(s_t) + \alpha[r_{t+1} + \gamma V(s_{t+1}) - V(s_t)] \tag{1.8}$$

步骤 6：如果 s_{t+1} 不是终止状态（即 $s_{t+1} \neq 16$），则令 $t=t+1$，跳转到步骤 3。如果 s_{t+1} 是终止状态且未满足算法的终止条件（num_Episode 等于预设值 MAX_Episode），则令 num_Episode=num_Episode+1，跳转到步骤 2。

步骤 7：根据最终的状态值函数 $V(s)$（$s \in S$）和贪婪策略的定义式（1.7）确定在每一阶段选择的行为，从而获得最终策略，并计算其对应的路径长度。

采用上面的 TD 算法进行 300 次试验（算法 1.1 中的 MAX_Episode 设为 300），得到各个状态值 $V(s)$ 随着试验次数的变化曲线如下：