云南大学学报（自然科学版），2010，32（1）：36～42　　CN 53－1045/N　ISSN 0258－7971
Journal of Yunnan University

一种基于博弈论的交通系统最优调度策略学习方法*[1]

韩　格，岳　昆，刘惟一
（云南大学 信息学院计算机科学与工程系，云南 昆明　650091）

摘要：交通网中，最大化车流量和最小化平均等待时间是每一个路口调度的目标. 交通调度中，各路口与其它路口发生博弈关系. 博弈过程中，相邻路口之间为使其自身利益最大化而存在一种策略间相互协调的约束. 针对复杂的交通调度控制问题，基于多智能体多阶段博弈论对交通系统进行建模. 考虑动态博弈交通环境的实际特征，进一步基于博弈的增强学习算法，提出一种以惩机制为约束条件的交通系统博弈策略的学习方法，最终使参与交通博弈的多个路口达到 Nash 均衡，从而得到交通系统的最优配时调度策略组合. 实验验证了所提出方法的可行性和有效性.

关键词：交通调度；博弈论；增强学习；协调约束；Nash 均衡
中图分类号：TP 311　　**文献标识码**：A　　**文章编号**：0258－7971(2010)01－0036－07

近年来，随着城市现代化交通的发展，社会迫切需求建立一套智能化交通系统网络. 在交通网络[1]中，每一个路口为使得自身有最高的车流量和平均最短的车辆等待时间，与相邻路口共享和竞争有限路段资源. 因此，某路口与相邻路口之间的调度会引发利益冲突.

博弈论[2－4]是研究若干利益冲突者在同一环境中如何进行决策以满足自身利益的学科. 博弈论中，给定其他人策略的条件下，每个利益冲突者选择自身最优的策略，从而使自身达到最大利益的过程称为 Nash 均衡[2－3]. 交通网中的每一个交通调度路口均以最大化车流量和最小化平均等待时间为调度利益目标. 调度路口获得的利益不仅取决于该路口自身的策略，而且还取决于其它路口的策略，因此各路口之间存在一种博弈关系. 博弈过程中，相邻路口之间为使其自身利益最大化而存在一种策略间相互协调的约束. 本文的研究旨在解决如何在动态博弈的交通系统中学习得到满足这种协调约束关系的策略，使得博弈的每一个调度路口在其所控制路段上有较高的车流量和较短的平均等待时间，最终达到 Nash 均衡，进而得到交通系统的最优调度策略.

在具有博弈关系的交通系统中，每一个调度路口的策略都受到相邻路口策略的影响，并且每一个路口在该影响下的收益（博弈论中用“效用”描述）都是未知的，因此使用经典的方法求解最优调度策略（即博弈交通系统的 Nash 均衡）是困难的. 增强学习（Reinforcement Learning，简称 RL）方法[5]是在不知道效用的情况下求解 Nash 均衡的有效方法，RL 通过“试错”的方式进行学习，当一个行为策略带来正确或错误的结果时，这种行为就被加强或减弱. 以最大化累积回报为目的，学习得到状态到行为策略的最佳映射. Q 学习[6]是一种最常用的 RL 算法，已经成功应用到智能体的控制及决策等领域[7－11]. 文献[8]基于适用于连续时间和状态的 Q 学习算法解决复杂的电梯调度问题. 文献[9]假定在格子世界中移动的 2 个机器人为到达各自的目标进行博弈，使用带有博弈的 Q 学习算法达到策略均衡.

近年来，增强学习算法被用于交通调度及决策

* **收稿日期**：2009－06－15
基金项目：国家自然科学基金资助项目(60763007)；云南省应用基础研究资助项目(2008CD0803)；云南省教育厅科研基金资助项目(08Y0023)；云南大学中青年骨干教师培养计划.
作者简介：韩　格(1983－)，男，河北人，硕士生，主要从事智能数据分析方面的研究.
通讯作者：岳　昆(1979－)，男，云南人，博士，副教授，主要从事数据与知识工程方面的研究.

支持中,文献[10 – 11]基于 Q 学习算法解决交通调度问题. 文献[10]以最小化车辆等待时间为目的,使用多智能体 Q 学习算法选择车辆到达目的地的最佳路径;在学习过程中,值函数表示等待时间,通信合作的各调度路口通过最小化共享的值函数的方式,使车辆能够预测并选择有最少等待时间的路径. 文献[11]在不考虑相邻路口存在博弈关系的情况下,结合神经网络、并使用 Q 学习算法学习多个调度路口对车辆的协调控制策略,假设车辆的到达情况满足泊松过程[12],使用神经网络存储 Q 值表,处理方法比较合理. 但是,由于行为策略不是指定的红绿灯配时方案,因此学到的结果可能出现红绿灯在较短的时段内有频繁的交替现象.

可见,上述基于增强学习方法的交通系统调度策略的研究中,相邻调度路口之间的博弈问题没有得到较好的解决. 此外,在相邻路口的博弈过程中,可能会出现如下现象:一个交通路口在某一个方向上长时间开绿灯,而相邻路口在这个方向上长时间开红灯,这就会导致拥塞,无法使自身得到最大利益. 本文将这种现象称为“不协调策略冲突”,上述增强学习方法仍然不能较好地处理这种现象.

基于上述讨论,我们首先对交通系统进行建模,然后基于博弈的增强学习算法研究约束条件下的交通调度策略学习方法. 具体而言,本文的主要工作可概括如下:

本文将观测到的车辆流量大小作为一项必要的参数进行最优策略的学习. 为了学习得到相互协调的策略,本文以调度路口的车辆数目作为收益,以车辆排队等待的时间作为惩罚,以最大化收益与惩罚之间的差值为目标,学习最优的配时策略.

采用能够求解 Nash 均衡的 Q 学习算法,实现了每一调度路口学习到最优协调的配时策略,最终达到 Nash 均衡. 实验表明使用惩罚约束学习配时策略的方法能够提高车流量和减少等待时间.

本文后续内容的组织结构如下:第 1 节描述基于博弈论对多智能体博弈交通模型进行建模的问题;第 2 节基于博弈交通模型解决如何在动态的交通环境中描述回报值函数,并学习到受协调约束的最优配时策略;第 3 节通过实验测试本文所提出方法的可行性和有效性;第 4 节总结全文并展望将来的研究工作.

1　问题描述

由“十字型”构成的交通网络中,每一个交通调度路口都被称为一个 Agent. Agent 能够在动态的交通环境中学习知识,并在下一次红绿灯交替时选择开灯持续时间,即选择配时策略. 有限多个交通调度 Agent 构成的博弈环境称为“有限博弈交通网”,如图 1 所示. 我们将 1 个 Agent 与其他 4 个相邻的 Agent 的共享路段的右行道称为该 Agent 的管辖区,而左行道称为该 Agent 的释放区. 从图 2 可以看出,管辖区和释放区的定义是相对,某 Agent 的一个管辖区也是这个路段上相邻 Agent 的一个释放区. 每个 Agent 通过采取决策使得在自己的管辖区上获得最大利益. 同时该策略也影响到释放区的车流量. 所以,相邻 Agent 之间共享路段的管辖区和释放区,构成了相邻 Agent 博弈的环境. 我们称这种博弈为“直接博弈”. 对于非直接相邻的 Agent 之间的博弈关系,我们称为“间接博弈”.

鉴于讨论的方便,我们首先给出如下假设,作为本文研究的前提:

本文考虑在不会导致交通完全恶化的低车辆流量情况下,如何解决局部拥塞以及提高每一个调度路口的车辆通行总量和降低平均车辆等待时间. 所有管辖区和释放区均只有一排车辆行驶,假设所有车辆以车速 v 行驶.

Agent 控制的交通灯色只有绿色和红色,分别表示通行和等待规则. 在开绿灯的路段上行驶的车辆经过 Agent 时,可以直行、左转或右转;在开红灯的路段上行驶的车辆经过 Agent 时,只能等待或右转. 因此,可能存在从红灯方向上驶入管辖区的车辆.

为了使每一个 Agent 学习到协调策略,最终达到 Nash 均衡. 本节基于多 Agent 多阶段博弈理论对有限博弈交通网进行建模. 使用 GFM(Game Traffic Model)表示多 Agent 博弈交通模型,定义如下:

定义 1　多 Agent 多阶段博弈交通模型为一个六元组

$$\mathrm{GFM} = \langle \mathrm{N}, \boldsymbol{S}, \boldsymbol{A}, r, p \rangle.$$

其中:N 表示有限 N 个博弈的 Agent;$\boldsymbol{S} = S^1 \times S^2 \times \cdots \times S^i \cdots \times S^N$ 表示所有 Agent 可能所处的状态空间. S^i 是 Agent i 可能所处的局部状态构成的集合. 用向量 $\boldsymbol{s}$ 表示某时刻所有 Agent 所处全局状态;$\boldsymbol{A} = A^1 \times A^2 \times \cdots \times A^i \cdots \times A^N$ 表示配时策略向量空间. A^i 表示 Agent i 可采用的策略构成的集合. N 个 Agent 采取的联合配时策略记为向量 $\boldsymbol{a}$,$\boldsymbol{a} \in \boldsymbol{A}$;

图1 有限博弈交通网

Fig. 1 Traffic network of a finite game

图2 图1中 Agent *A* 和 Agent *O* 的博弈环境

Fig. 2 Game environment of agent *A* and *O* in Fig. 1

$r = \{r^1, r^2, \cdots, r^i, \cdots, r^N\}$ 表示所有 Agent 的回报函数构成的集合. $r^i: S \times A^1 \times A^2 \times \cdots \times A^N \to R$ 表示 Agent i 得到的回报值,它是当前状态和联合配时策略到实数 R 的映射. 用 $r^i(\boldsymbol{s}, \boldsymbol{a})$ 表示在状态 $\boldsymbol{s}$ 下,N 个 Agent 采取联合策略 $\boldsymbol{a}$ 时,Agent i 所得回报值;$p: S \times A^1 \times A^2 \times \cdots \times A^N \to [0,1]$ 表示当前状态和 N 个 Agent 采取的联合配时策略 $\boldsymbol{a}$ 到下一个状态的概率分布映射.

本文将西、北、东、南4个管辖区上驶向 Agent i 而未通过 Agent i 的第1辆车所在位置称为 Agent i 的一个局部状态 $s_i, s_i \in S^i$. 所有 Agent 所处在的局部状态分量构成了全局状态 $\boldsymbol{s}$. 全局状态就是多 Agent 的博弈环境. 如果以车辆到 Agent 的精确距离表示局部状态,就会使得状态集合无穷大. 实际上,行驶的车辆在很短的时间上行驶了较长距离,因此可以将所有路段划分成若干等长度的小区间,以小区间的编号表示局部状态. 离 Agent 最近的区间编号为最小值1,依次递增. 例如:Agent i 某时刻所处状态为 $\langle 8,2,3,7\rangle_i$, $\langle 8,2,3,7\rangle_i \in S^i$. 对于有限 N 个 Agent 的博弈交通模型中,我们用含有 N 个四元组的向量表示一个全局状态. 例如:某时刻观测到 $\boldsymbol{s} = \langle \langle 3,4,15,6\rangle_1, \cdots, \langle 8,2,3,7\rangle_i, \cdots, \langle 20,1,7,12\rangle_N\rangle$. 在计算车辆行驶时间的情况下,依然使用车辆到 Agent 的精确距离.

Agent i 的策略集合 A_i 是以路段长度与车速的关系制定的. 例如 $A^i = \{a_1 = l_w/2v, a_2 = l_n/2v, a_3 = l_e/2v, a_4 = l_s/2v, a_1 = l_w/3v, a_2 = l_n/3v, a_3 = l_e/3v, a_4 = l_s/3v\}$. 其中 l_w, l_n, l_e, l_s 分别表示该 Agent 在西、北、东、南4个方向上的路段长度.

在状态数目较多的博弈交通系统中得到状态转移概率函数 p 是困难的. 增强学习通过采用逼近的方法进行值函数估计[13],有效避免了获得 p. 回报值函数 r^i 是环境对 Agent i 的行为策略好坏的打分评价机制. 学习的过程就是搜索能使回报值最大的配时策略. 回报函数的值通常是无法知道的,往往是通过观测得到的. 恰当地描述回报值表达形式是学习到最优协调配时策略的关键.

下面第2节将基于博弈交通模型的定义,讨论协调约束条件下各 Agent 最优配时策略的学习方法.

2 受协调约束的最优调度策略学习

任意2个相邻路口 Agent 直接博弈的过程中,可能会出现"不协调策略冲突". 以图2为例,Agent *A* 采用的策略使得释放区有较高强度的车辆流驶向 Agent *O*. 而此时 Agent *O* 为使得在垂直方向上有较大的通行量,在此方向上长时间开放绿灯,与此同时,在水平方向上必然是长时间开放红灯. 由

于 Agent O 的“不协调策略”致使自身所控制的管辖区内的车辆平均等待时间延长. 所以,我们使用通行量和等待时间描述回报值. 此外,间接博弈也会对回报值有所影响. 本节将详细讨论回报值的计算方法,以及求解多 Agent 博弈过程中的 Nash 均衡策略.

2.1　直接博弈的收益值和惩罚值的计算　有限博弈交通网中的每一个 Agent 有相同的收益和惩罚计算方式. 下文以 Agent i 为例,描述因直接博弈而产生的收益值和惩罚值. 我们将车辆通行总量作为收益值,车辆累积等待时间作为惩罚值,使用收益与惩罚的差值作为增强学习的回报值,约束学习“协调策略”.

我们将车辆流视为泊松流[13],关于时间 ω 的泊松流概率密度函数如下:

$$f(\omega) = \frac{\lambda_m \omega^{(n-1)!} \mathrm{e}^{-\lambda_m \omega}}{(n-1)!}, \tag{1}$$

其中,ω 为时间变量,n 为到达的车辆数,λ_m 为管辖区上的泊松强度. $m = 1,2,3,4$ 分别表示西、北、东、南4个方向. 强度 λ_m 值是由与 Agent i 直接博弈的4个 Agent 的状态和策略决定的. 我们给出函数 $\lambda(\ )$ 表示 λ_m 值,表达式如下:

$$\lambda_m = \lambda(s_m, a_m), \tag{2}$$

其中,s_m 和 $a_m(m = 1,2,3,4)$ 分别为西、北、东、南方向上直接博弈 Agent 所在状态和配时策略,λ_m 值是通过多次观测统计历史数据的平均值得到的.

假定在红绿灯发生交替时 Agent i 采用配时策略 a 执行(即开放红绿灯的时间为 a 秒). 将开绿灯的某一管辖区上,未通过 Agent i 的第1辆车位于该管辖区的某一区间,与 Agent i 的距离为 l',那么到达并通过 Agent i 用时 $t = l'/v$. 若 $t < a$,则将此车到达时刻记为0时刻,后继车辆按 λ 为参数的泊松流到达. 按公式(1)和(2)的描述,下面给出以通行总量作为收益的表达式:

$$\sum_{n=1}^{\infty} \int_0^{a-1} \frac{\lambda \omega^{(n-1)!} \mathrm{e}^{-\lambda \omega}}{(n-1)!} \mathrm{d}\omega = \lambda(a-t). \tag{3}$$

惩罚值的计算方法与收益值的计算方法相类似. 假定在红绿等交替的瞬间,Agent i 采用配时策略 a 执行. 即将开放红灯的某一管辖区上,未通过该 Agent 的第1辆车到达该 Agent i 用时为 t. 若 $t < a$,则将此车到达时刻记为 0 时刻,此后等待在 Agent i 前,直到再次 Agent i 调度时发生红绿灯交替,即等待了$(a-t)$在这个时段内,又有若干车辆在时刻 p $(0 \leqslant p \leqslant a-t)$ 按值为 λ 的泊松强度到达并开始等待. Agent i 因车辆等待而受到惩罚,我们以等待时间长短定义惩罚度:

$$\int_p^{a-t} k \mathrm{d}x, \tag{4}$$

其中:k 为正常数,表示惩罚量的大小;p 为车辆到达时刻,$0 \leqslant p \leqslant a-t$. 因此,根据公式(1)、(2)和(4),我们可以计算从0时刻到 $a-t$ 时刻,n 辆车到达该 Agent 后,因累积等待时间而受到的惩罚值:

$$\sum_{n=1}^{\infty} \int_0^{a-t} \frac{\lambda \omega^{(n-1)!} \mathrm{e}^{-\lambda \omega}}{(n-1)!} \int_p^{a-t} k \mathrm{d}x \mathrm{d}\omega =$$

$$\int_0^{a-t} k\lambda(a-t-\omega) \mathrm{d}\omega =$$

$$\frac{1}{2} k\lambda (a-t)^2. \tag{5}$$

对于公式(4)和(5),当 $t > a$ 时,说明未通过的第一辆车远离该 Agent i,在红绿灯持续的 a 策略时段内无法到达 Agent i. 因此没有车辆通过或等待,Agent i 不受收益或惩罚,即收益值或惩罚值为0. 此外,从求解车辆行驶到 Agent 所用时间 t 的过程可以看出,t 值依赖于红绿灯交替时 Agent 所处的状态;同时,由公式(2)可以看出 λ 值依赖于博弈过程中的观测.

2.2　回报值的计算　为了使 Agent i 学习到“协调策略”,使用收益和惩罚的差值作为增强学习回报值,约束学习最优配时策略. 学习的过程就是最大化这种差值的过程. 根据公式(3)和(5),可以计算 Agent i 在局部状态 s_i 下,采用配时策略 a_i 时,Agent i 以直接博弈得到的回报值. 其中 $s_i \in S^i$ 和 $a_i \in A^i$ 分别为向量 $\boldsymbol{s}$ 和 $\boldsymbol{a}$ 的分量.

$$r(s_i, a_i) = \lambda_n(a_i - t_n) + \lambda_s(a_i - t_s) - \frac{1}{2} k\lambda_w (a_i - t_w)^2 - \frac{1}{2} k\lambda_e (a_i - t_e)^2, \tag{6}$$

$$r(s_i, a_i) = \lambda_w(a_i - t_w) + \lambda_e(a_i - t_e) - \frac{1}{2} k\lambda_n (a_i - t_n)^2 - \frac{1}{2} k\lambda_s (a_i - t_s)^2, \tag{7}$$

其中,t_w,t_n,t_e 和 t_s 分别表示以红绿灯交替调度时刻为起始时刻,西、北、东、南4个管辖区上未通过该 Agent 的第1辆车到达该 Agent 所用的时间,λ_w,λ_n,λ_e 和 λ_s 分别表示4个区域上驶向 Agent i 的车辆泊松流强度. 公式(6)和(7)分别用以计算东西

方向上开绿灯和南北方向上开绿灯情况下的直接博弈回报值. 若 $a_i < t$(t 表示 t_w,t_n,t_e和 t_s),记为 $a - t = 0$.

以图2中的 Agent O 为例,下面我们给出计算直接博弈回报值的方法. 某时刻 Agent O 发生红绿灯调度,即将选择在南北方向上开绿灯(东西方向上开红灯)的某配时策略 a_i 执行. 瞬间观测到西、北、东、南4个管辖区上未通过该 Agent 的第1辆车与该 Agent 的距离分别为 73.3,50.6,112.1,0.0 m,若以 20 m 为一个划分区间,则对应的局部状态为〈4,3,6,1〉. 设车速为 2.5 m/s,那么4个方向上到达 Agent O 所用时间分别为 29.32,20.24,44.84,0.0 s. 假设观测并计算得到4个方向上的泊松强度值 $\lambda_w = 3$,$\lambda_n = 5$,$\lambda_e = 6$,$\lambda_s = 4$. 若此时 Agent O 选择配时策略 $a_i = 30$ s,设 $k = 0.25$,按照公式(6)即可计算直接博弈的回报值. 对于44.84 s > 30 s,表明在配时策略 a_i 执行时段内,东侧管辖区上没有车辆到达并通过,即在该方向上的收益值为0. $r(s_i,a_i) = 3 \times (30 - 29.32) + 6 \times 0 - (1/2) \times 0.25 \times 5 \times (30 - 20.24)^2 - (1/2) \times 0.25 \times (30 - 0)^2 = -454.06$.

回报值函数 $r^i(\boldsymbol{s},\boldsymbol{a})$ 不仅依赖于直接博弈产生的回报值,而且间接博弈对该值也有一定影响. 下面我们给出博弈相关度的定义,来描述相关博弈 Agent 对 Agent i 的回报值函数 $r^i(\boldsymbol{s},\boldsymbol{a})$ 的影响.

定义 2 有限博弈交通网中,设 Agent i 到 Agent j 的所有路径组成的集合 $L_{ij} = \{l_1,\cdots,l_m,\cdots,l_n\}$. 用 $|l_x|$ $(1 \leqslant x \leqslant n)$ 表示在路径 l_x 上 Agent i 到 Agent j 经过 Agent 的个数. 从 Agent i 到 Agent j 所经过的最少 Agent 的个数称为博弈相关度. 设 Agent i 与 Agent j 的博弈相关度为 $|l_m|$,则 $|l_m| = \min(|L_{ij}|),1 \leqslant m \leqslant n$.

设有限博弈交通网中的博弈相关度最大值为 d. 例如图1中,Agent O 与 Agent C 之间的博弈相关度为0;Agent A 与 Agent C 之间的博弈相关度为1;该网的博弈相关度最大值 $d = 3$. 可以看出博弈相关度越大,Agent 之间博弈关系越弱. 根据定义2,我们给出下式计算回报值函数:

$$\begin{aligned} r^i(\boldsymbol{s},\boldsymbol{a}) = {} & r(s_i,a_i) + \beta[r(s_j,a_j) + \cdots] + \\ & \beta^2[r(s_k,a_k) + \cdots]\cdots + \\ & \beta^d[r(s_l,a_l) + \cdots], \end{aligned} \tag{8}$$

其中,β 为影响因子 $(0 \leqslant \beta < 1)$,$\beta^d[r(s_l,a_l) + \cdots]$ 表示与 Agent i 博弈相关度为 d 的 Agent 对 $r^i(\boldsymbol{s},\boldsymbol{a})$ 值的影响.

2.3 多 Agent 博弈交通模型的 Nash - Q 学习算法 在多个 Agent 多阶段博弈的交通环境中,每个 Agent 以协调的方式达到自身利益最大,最终使多 Agent 达到 Nash 均衡. 文献[14]给出了求解 Nash 均衡的 Q 学习算法,并证明 Q 值的收敛到 Nash 均衡. 本文使用该算法求解有限 n 个 Agent 在交通博弈过程中的 Nash 均衡配时策略,基本思想由算法1给出.

算法 1 求解 n 个 Agent 的 Nash 均衡配时策略

输入:设置折扣因子 $\eta(0 \leqslant \eta < 1)$,足够大循环次数 T.

输出:博弈交通网络中所有 Agent 在各种可能状态和配时策略下的 Q 值表.

步骤:

(1) 对所有可能状态 $\boldsymbol{s}$,联合策略 $\boldsymbol{s}$,初始化每一个 Agent 的 Q^i 值:$Q^i(\boldsymbol{s},\boldsymbol{a}) = 0$.

(2) 某时刻开始观测到多 Agent 所处状态 $\boldsymbol{s}$ 以及联合配时策略 $\boldsymbol{a}$.

(3) FOR $t = 0$ TO T DO:

每一个 Agent i 在状态 $\boldsymbol{s}$ 下选择一可能的策略 $a_i \in A^i$ 执行,$\langle a_1,\cdots,a_n\rangle = \boldsymbol{a}$;

每一个 Agent i 观测到由 $\boldsymbol{a}$ 决定的各管辖区中的泊松流强度以及第1辆车行驶到 Agnet i 所用时间,根据公式(8)计算 $r^i(\boldsymbol{s},\boldsymbol{a})$,并将 $\boldsymbol{s}$ 赋值为 $\boldsymbol{s}'$;

更新 Q^i 值:

$$\begin{aligned} & Q^1_{t+1}(\boldsymbol{s},\boldsymbol{a}) = \\ & (1-\alpha_t)\cdot Q^1_t(\boldsymbol{s},\boldsymbol{a}) + \alpha_t(r^1(\boldsymbol{s},\boldsymbol{a}) + \\ & \eta\cdot\pi^1(\boldsymbol{s}')\cdot\pi^2(\boldsymbol{s}')\cdots\pi^n(\boldsymbol{s}')\cdot Q^1_t(\boldsymbol{s}'))\ ; \\ & Q^2_{t+1}(\boldsymbol{s},\boldsymbol{a}) = \\ & (1-\alpha_t)\cdot Q^2_t(\boldsymbol{s},\boldsymbol{a}) + \alpha_t(r^2(\boldsymbol{s},\boldsymbol{a}) + \\ & \eta\cdot\pi^1(\boldsymbol{s}')\cdot\pi^2(\boldsymbol{s}')\cdots\pi^n(\boldsymbol{s}')\cdot Q^2_t(\boldsymbol{s}'))\ ; \\ & \cdots\cdots \\ & Q^n_{t+1}(\boldsymbol{s},\boldsymbol{a}) = \\ & (1-\alpha_t)\cdot Q^n_t(\boldsymbol{s},\boldsymbol{a}) + \alpha_t(r^n(\boldsymbol{s},\boldsymbol{a}) + \\ & \eta\cdot\pi^1(\boldsymbol{s}')\cdot\pi^2(\boldsymbol{s}')\cdots\pi^n(\boldsymbol{s}')\cdot Q^n_t(\boldsymbol{s}'))\ ; \end{aligned}$$

(4) 输出所有的 $Q^i(\boldsymbol{s},\boldsymbol{a})$ 值,即 Q^i 表 算法1中,α_t 为学习速率 $(0 < \alpha_t \leqslant 1)$,$\alpha_t = 1/(1 + N_t(\boldsymbol{s},\boldsymbol{a}))$,$N_t(\boldsymbol{s},\boldsymbol{a})$ 为全局状态和联合策略对 $(\boldsymbol{s},\boldsymbol{a})$ 在这 t 次训练中被访问的总次数. $\pi^i(\boldsymbol{s}')$ 表示在状态 $\boldsymbol{s}'$ 下 Agent i 采用的混合配时策略,$Q^i_t(\boldsymbol{s}')$ 为在状

态 s' 下 Agent i 的 Q 值表. 在得到 Q^i 表后,在相同状态 s 下查找最大 Q 值,此时的配时策略组合即为 Nash 均衡策略.

算法 1 的时间复杂度依赖于求解 Nash 均衡过程中 Q 值表的更新. 文献[9]指出 Nash－Q 学习算法在最坏情况下具有指数计算时间. 本文以路口 Agent 数量较少的情形为代表进行测试,以验证实际中算法 1 的计算时间是可接受的.

3　实验结果

本节以实验的方式验证算法 1 在路口 Agent 数量较少的情形下有可接受的执行时间,以及从历史数据中学习“协调约束”策略的有效性. 实验中我们以包含 3 个 Agent 的博弈环境为测试对象,观测到的状态以随机的方式生成,实验过程如下:

以图2 中的 Agent A,Agent O 和 Agent B 为例,假设每次观测到驶入该交通网的车辆泊松流强度在区间[4,10]上服从均匀分布. 3 个 Agent 在西、北、东、南 4 个方向上的长度分别为:60,60,80,40,80,60,40,100,40,60,80,60. 车速为 2.5. 设 Agent A,Agent O 和 Agent B 策略分别为 $A^A=\{20,35,40\}$,$A^o=\{25,30,45,60\}$,$A^B=\{35,45\}$. 以 20 为 1 个区间长度划分各个路段,构成 622 080 个状态. 每次观测到未通过这 3 个 Agent 的第 1 辆车也按均匀分布生成在这个状态空间上. 算法执行 11 min 后,3 张 Q 值表中的数据基本收敛. 如图 2 所示,我们给出在状态 $\boldsymbol{s}=\langle\langle 2,1,3,1\rangle,\langle 1,3,2,5\rangle,\langle 2,3,4,2\rangle\rangle$,联合配时策略 $\boldsymbol{a}=\langle 20,50,35\rangle$时,3 个 Q 值变化过程.

图 3 给出的收敛性实验结果表明我们的算法在参与博弈的 Agent 数量较少时,求解均衡所耗费的时间是可以接受的. 基于文献[14]的结论,Q 值收敛时即达到 Nash 均衡. 一般情况下,城市交通状况在几个月时间内都有稳定的流量,对于求解均衡策略组合所需的十几分钟来说是可以接受的[1].

在得到收敛的 Q 值表后,我们验证学习“协调约束”策略的有效性. 我们分别使用单配时策略、随机混合配时策略、查找 Q 值表选择策略,这 3 种选策略方式下得到的结果进行比较. 按前述设定的参数对 Agent O 进行测试,并给定多个由状态、泊松强度、Agent A 和 Agent B 配时策略组成的序列. 反复实验测试表明,10 次以上的策略调度能够反映 3 种调度方式的性能. 因此,为了使 Agent O 发生 10 次以上调度,在相同假定的参数下,我们比较了 3 种选策略方式下通过车辆的总数和平均等待时间. 我们分别用 N 和 W 表示通过车辆总数和平均等待时间,如表 1 所示. 可见,与单一策略或随机策略相比,以查找 Q 值表的方式选择配时策略会有较高的车辆通行量和较短的平均等待时间.

图 3　3 个 Agent 在 s 和 a 下 Q 值随学习时间的变化

Fig. 3　Values of Q with the increase of learning time under s and Q of 3 agents

表 1　Agent O 的 3 种策略选择方式性能比较

Tab. 1　Performance comparisons of 3 strategy choices of Agent O

选择策略方式	N	W
始终选 $a_1=70$	271	46
始终选 $a_3=25$	336	41
随机选 A^o 中策略	305	44
查找 Q 值选策略	367	36

4　总结与展望

本文基于多 Agent 多阶段博弈理论对动态复杂的博弈交通问题进行了建模,为研究现代交通控制问题提出了一种可供参考的思路. 本文以惩罚机制为约束,较为有效地学习到协调配时策略,使用 Nash－Q 学习算法基于交通调度的历史数据求解得到最优配时策略组合. 实验表明了本文的方法在一定程度上提高了车辆的通行量和降低了平均等待时间.

基于本文提出的方法,对于状态空间较大的多 Agent 博弈过程需要大量的计算时间,且策略选择结果的好坏依赖于所制定的策略集,因此可能导致选出的策略并不是真正最优的策略. 为了解决该问

题,我们将以学习的方式制定策略集合,并尽可能缩小状态空间以提高学习效率,这是我们将来的研究工作.

参考文献:

[1] NAGEL K. Traffic networks[M]. Handbook of Graphs and Networks,2002,248-272.

[2] 候定丕. 博弈论导论[M]. 合肥:中国科学技术大学出版社,2004.

[3] 刘惟一,李维华,岳昆. 智能数据分析[M]. 北京: 科学出版社,2007.

[4] LIU W Y,LI J,YUE K,et al. An approach for solving fuzzy games[J]. Int'l Journal of Uncertainty, Fuzziness,and Knowledge - Based Systems,2006,14 (3): 277-292.

[5] MINSKY J F. Theory of neural - analog reinforcement systems and its application to the brain - model problem [D]. America:Princeton University,1954.

[6] WATKINS C. Learning with delayed rewards[D]. England:Cambridge University,1989.

[7] LITTMAN M, BOYAN J. A distributed reinforcement learning scheme for network routing[C]//Proc. of the 1st Int'l workshop on Application of Neural Networks to Telecommunication,1993:45-51.

[8] CRITES R H,BARTO A G. Improving elevator performance using reinforcement learning[C]//Proc. of NIPS' 1996:1 017-1 023.

[9] HU J L,MICHAEL P M. Nash Q - learning for general - sum stochastic games[J]. Machine Learning Research,2003,4:1 039-1 069.

[10] WIERING M A. Multi - agent reinforcement learning for traffic light control[C]//Proc. of ICML' 2000: 1 151-1 158.

[11] 王建宇,彭维,王康平,等. 增强学习与神经网络在交通信号控制中的应用[J]. 计算机工程与应用,2007,43 (31):242-244.

[12] 刘次华. 随机过程[M]. 武汉: 华中科技大学出版社,2001.

[13] KAEBLING L P,LITTMAN M L,MOORE A W. Reinforcement learning: a survey[J]. Artificial Intelligence Research,1996,4:237-285.

[14] HU J L,MICHAEL P M. Multiagent reinforcement learning:theoretical framework and an algorithm[C]//Proc of ICML'1998:242-250.

A game - theory - based approach for learning the optimal scheduling strategies in traffic systems

HAN Ge, YUE Kun, LIU Wei-yi

(Department of Computer Science and Engineering,School of Information Science and Engineering, Yunnan University,Kunming 650091,China)

Abstract: In traffic networks,the target of intersection scheduling is to maximize the flow rates and minimize average waiting time of all concerned vehicles. Game relationships exist between each intersection and the other ones in the traffic scheduling. In the process of this traffic game,there is a constraint of mutual coordination among strategies so that the maximal profits of neighboring intersections can be achieved. In this paper,we focus on the complex traffic scheduling problem,and give the modeling approach for traffic systems based on the multi - agent multi - step game theory. Considering the practical characteristics of dynamic traffic game environments,in this paper we further propose an approach for learning traffic scheduling strategies from historical traffic scheduling data based on the reinforcement learning algorithm in game theory. Then the Nash equilibrium of multiple intersections that participate in the traffic game can be achieved ultimately. Therefore,the optimal scheduling strategies of traffic systems will be obtained. Experimental results show the feasibility and effectiveness of our method.

Key words: traffic scheduling; game theory; reinforcement learning; coordination constraint; Nash equilibrium