

基于马尔科夫理论的不确定 性规划和感知问题研究

姓名： 柏爱俊

导师： 陈小平

中国科学技术大学
计算机科学与技术学院

2014 年 10 月 27 日

主要内容

- ① 不确定环境下自主规划和感知问题
- ② 基于 MAXQ 分层分解的在线规划算法
- ③ 基于后验动作采样的蒙特卡洛在线规划算法
- ④ 基于集合粒子滤波的多对象跟踪算法
- ⑤ 总结和展望

主要内容

- ① 不确定环境下自主规划和感知问题
- ② 基于 MAXQ 分层分解的在线规划算法
- ③ 基于后验动作采样的蒙特卡洛在线规划算法
- ④ 基于集合粒子滤波的多对象跟踪算法
- ⑤ 总结和展望

自主智能体和多智能体系统



Figure 1 : 各种智能体系统

智能体感知和规划任务

- 感知：状态估计问题
 - 输入：观察和行动序列
 - 输出：状态或状态分布
- 规划：序列化决策问题
 - 输入：状态或状态分布
 - 输出：行动序列

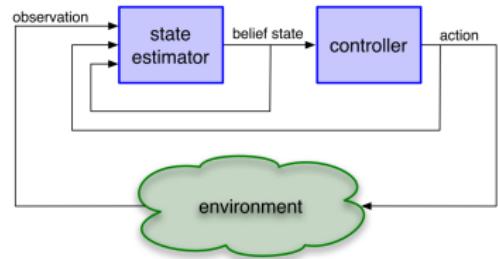


Figure 2 : 与环境进行交互

环境不确定性

- 观察不确定性
 - 无法准确估计当前状态
 - * 不准确的观测数据
 - * 不可观测的隐藏信息
- 行动不确定性
 - 无法准确预测未来状态
 - * 不可预知的行动结果

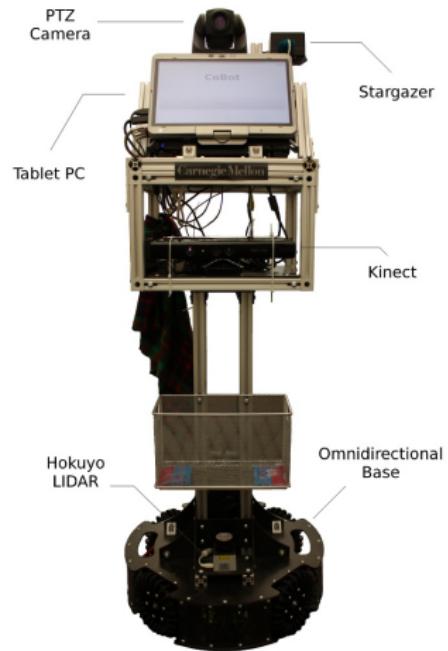


Figure 3 : CoBot 机器人

研究动机

- 理论基础
 - 马尔科夫决策过程 (MDP)
 - 部分可观察马尔科夫决策过程 (POMDP)
- 技术路线
 - 建模：根据马尔科夫理论建立环境的概率模型
 - 感知：通过贝叶斯方法推理出环境状态的概率分布
 - 规划：通过分层分解和蒙特卡洛方法进行启发式搜索

马尔科夫决策理论

- 为不确定性环境下的规划和感知问题提供了基本的理论框架 (Puterman, 1994; Kaelbling et al., 1998)
- 马尔科夫性
 - 系统的状态转移仅依赖于当前状态和行动，不依赖于过去的状态和行动，即：

$$\Pr(s_{t+1} | s_0, a_0, s_1, a_1, \dots, s_t, a_t) = \Pr(s_{t+1} | s_t, a_t) \quad (1)$$

马尔科夫决策过程 (MDP)

- MDP 仅考虑动作不确定性
 - 状态空间： $S = \{s_1, s_2, \dots, s_{|S|}\}$
 - 动作空间： $A = \{a_1, a_2, \dots, a_{|A|}\}$
 - 转移函数： $T(s' | s, a) : S \times A \times S \rightarrow [0, 1]$
 - 回报函数： $R(s, a) : S \times A \rightarrow \mathbb{R}$
- 策略： $\pi(s) : S \rightarrow A$
- 值函数： $V^\pi(s_0) = \mathbb{E} \left[\sum_{t \geq 0} \gamma^t R(s_i, \pi(s_i)) \right]$
- 贝尔曼最优性等式：

$$V^*(s) = \max_{a \in A} \left\{ R(s, a) + \gamma \sum_{s' \in S} T(s' | s, a) V^*(s') \right\} \quad (2)$$

部分可观察马尔科夫决策过程 (POMDP)

- POMDP 同时考虑观察不确定性

- 观察空间： $O = \{o_1, o_2, \dots, o_{|O|}\}$

- 观察函数： $\Omega(o | a, s') : S \times A \times O \rightarrow [0, 1]$

- 历史： $h = (a_0, o_1, a_1, o_2, \dots, a_{t-1}, o_t)$

- 信念状态： $b(s) : S \rightarrow [0, 1]$

- 信念更新： $b' = \zeta(b, a, o)$, 即：

$$b'(s') = \eta \Omega(o | s', a) \sum_{s \in S} T(s' | s, a) b(s) \quad (3)$$

- POMDP 可以转化成信念状态空间上的连续 MDP 问题

主要工作

- 基于 MAXQ 分层分解的 MDP 在线规划算法
 - MAXQ-OP (Bai et al., 2012a,b,c, 2013b)
- 基于后验动作采样的蒙特卡洛 (PO)MDP 在线规划算法
 - DNG-MCTS (Bai et al., 2013a)
 - D²NG-POMCP (Bai et al., 2014b)
- 基于集合粒子滤波的多对象跟踪 POMDP 信念更新算法
 - PFS (Bai et al., 2014a)

主要内容

- ① 不确定环境下自主规划和感知问题
- ② 基于 MAXQ 分层分解的在线规划算法
- ③ 基于后验动作采样的蒙特卡洛在线规划算法
- ④ 基于集合粒子滤波的多对象跟踪算法
- ⑤ 总结和展望

MAXQ 分层分解

- 把一个原始 MDP 分解成一系列子 MDP (Dietterich, 1999)

- $M_i = \langle S_i, G_i, A_i, R_i \rangle$

- * 活动状态 S_i

- * 目标状态 G_i

- * 可选动作 A_i

- * 局部回报函数 R_i

- 局部策略 $\pi_i : S_i \rightarrow A_i$

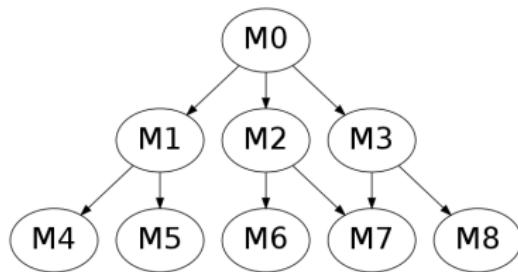


Figure 4 : MAXQ 分层结构

递归最优策略

- 分层策略： $\pi = \{\pi_0, \pi_1, \dots, \pi_n\}$
 - 局部策略的集合
- 递归最优策略 π^* 的值函数 V^* 满足：

$$V^*(i, s) = \begin{cases} R(s, i) & \text{如果 } M_i \text{ 是原子动作} \\ \max_{a \in A_i} Q^*(i, s, a) & \text{其他情况} \end{cases} \quad (4)$$

$$Q^*(i, s, a) = V^*(a, s) + C^*(i, s, a) \quad (5)$$

$$C^*(i, s, a) = \sum_{s', N} \Pr(s', N | s, a) V^*(i, s') \quad (6)$$

- π^* 满足：

$$\pi_i^*(s) = \operatorname{argmax}_{a \in A_i} Q^*(i, s, a) \quad (7)$$

完成函数

- 完成函数：

$$C^*(i, s, a) = \sum_{s', N} \Pr(s', N | s, a) V^*(i, s') \quad (8)$$

- 其中，

$$\begin{aligned} \Pr(s', N | s, a) &= \\ \sum_{\langle s, s_1, \dots, s_{N-1} \rangle} \Pr(s_1 | s, \pi_a^*(s)) \cdot \Pr(s_2 | s_1, \pi_a^*(s_1)) &\dots \Pr(s' | s_{N-1}, \pi_a^*(s_{N-1})) \Pr(N | s, a) \end{aligned} \quad (9)$$

- $\langle s, s_1, \dots, s_{N-1} \rangle$ 是策略 π^* 下从 s 到 s' 的状态转移路径
- 完全求解完成函数等价于求解整个问题
 - 在线规划时是不可能的

完成函数近似

- 引入子任务的终止分布：

$$\Pr(s' | s, a) = \sum_N \Pr(s', N | s, a) \quad (10)$$

- 完成函数重写为：

$$C^*(i, s, a) = \sum_{s'} \Pr(s' | s, a) V^*(i, s') \quad (11)$$

- 在线或离线近似估计 $\Pr(s' | s, a)$
 - 离线：离线统计子任务结束时的状态分布
 - 在线：近似估计子任务结束时的状态分布

MAXQ-OP 算法的主体结构

- 非原子任务：

$$V^*(i, s) \approx \max_{a \in A_i} \left\{ V^*(a, s) + \sum_{s'} \Pr(s' | s, a) V^*(i, s') \right\} \quad (12)$$

- 引入深度数组 d , 最大深度数组 D 以及启发函数 $H(i, s)$:

$$V(i, s, d) \approx \begin{cases} H(i, s) & \text{如果 } d[i] \geq D[i] \\ \max_{a \in A_i} \{ V(a, s, d) + \\ \sum_{s'} \Pr(s' | s, a) \\ V(i, s', d[i] \leftarrow d[i] + 1) \} & \text{其他情况} \end{cases} \quad (13)$$

标准测试：出租车问题

- 状态： $25 \times 5 \times 4 = 400$
 - 出租车位置： (x, y)
 - 乘客位置：R、Y、B、G、In
 - 目的地位置：R、Y、B、G
- 动作：6
 - North、South、East、West
 - * 成功概率：0.8
 - * 失败概率：0.2
 - Pickup、Putdown

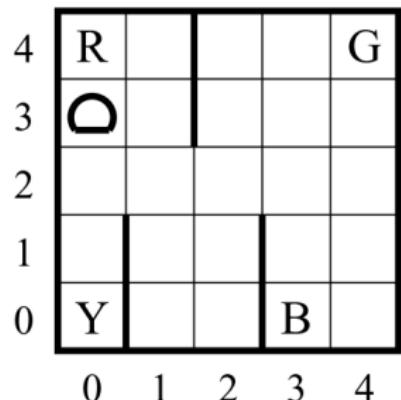


Figure 5 : 出租车问题

出租车问题实验结果

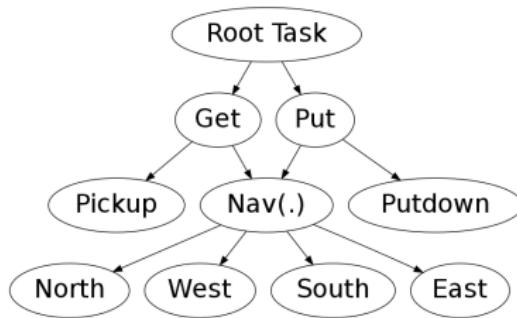


Figure 6 : 出租车问题任务图

Table 1 : 出租车问题实验结果

算法	平均回报*	在线计算时间 (ms)
MAXQ-OP	3.93 ± 0.16	0.20 ± 0.16
LRTDP	3.71 ± 0.15	64.88 ± 3.71
AOT	3.80 ± 0.16	41.26 ± 2.37
UCT	-23.10 ± 0.84	102.20 ± 4.24

* 出租车问题最优回报值是 4.01 ± 0.15 .

案例研究：RoboCup 仿真 2D 机器人足球

- 仿真的足球比赛
- 两队各 11 名球员
- 每个球员都是独立进程
- 每个决策周期（100ms）
 - 接收观察信息
 - 更新世界模型
 - 做出实时决策
 - 发送行动命令



Figure 7 : RoboCup 2D.

RoboCup 仿真 2D 机器人足球

- 关键特征
 - 仿真器引入的抽象
 - 关注于高层规划、学习和合作的任务
 - 无需考虑底层硬件问题
- 主要挑战
 - 完全分布式多智能体动态随机系统
 - 连续的状态、观察和动作空间

科大“蓝鹰”仿真 2D 机器人足球队

- 蓝鹰 (WrightEagle)
- 世界冠军：2006、2009、
2011、2013 和 2014
- 世界亚军：2005、2007、
2008、2010 和 2012
- 自 2007 年起成为主要成员
- 基于 MAXQ-OP 决策框架的
工作开始于 2009 年

RoboCup 2D Soccer Simulation League

Founded	1997
Region	International
Current champions	 WrightEagle (5th title)
Most successful team(s)	 WrightEagle (5 titles)
Website	www.robocup.org 

Figure 8 : 最成功的球队

MAXQ-OP 解决方案

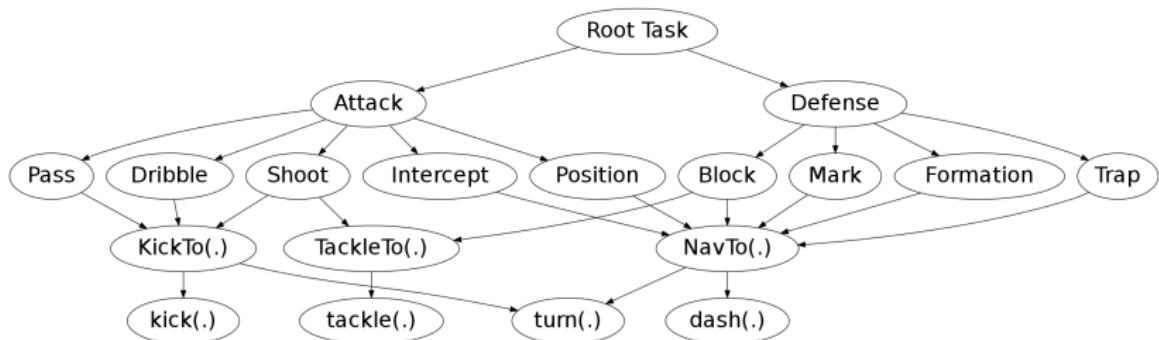


Figure 9 : 基于 MAXQ 分层结构的任务图

- 由于时间关系，在此不做展开

与传统在线规划算法的比较

- 传统在线规划算法：RTDP、AOT、UCT 等
 - 仅在状态空间里面进行搜索
 - 搜索路径： $[s_1 \rightarrow s_2 \rightarrow s_3 \rightarrow \dots \rightarrow s_H] \rightsquigarrow g$
- MAXQ-OP 算法
 - 同时在状态空间和层次结构上搜索
 - 搜索路径： $[s_1 \rightarrow \dots \rightarrow s_{H_1}] \rightsquigarrow [g_1/s'_1 \rightarrow \dots \rightarrow s'_{H_2}] \rightsquigarrow [g_2/s''_1 \rightarrow \dots \rightarrow s''_{H_3}] \dots \rightsquigarrow g$
- MAXQ-OP 通过利用分层结构可以搜索到更深的状态

小结

- 基于 MAXQ 分层分解的在线规划算法 MAXQ-OP
 - 在线利用 MAXQ 分层结构
 - 完成函数近似计算方法
- 标准出租车问题测试
 - MAXQ-OP 可以找到近似最优解
 - 相比于传统在线规划算法，只需要极少的计算时间
- RoboCup 2D 长期案例研究
 - WrightEagle 取得了非常优秀的比赛成绩
 - 证实了解决大规模问题的可行性和可扩展性

主要内容

- ① 不确定环境下自主规划和感知问题
- ② 基于 MAXQ 分层分解的在线规划算法
- ③ 基于后验动作采样的蒙特卡洛在线规划算法
- ④ 基于集合粒子滤波的多对象跟踪算法
- ⑤ 总结和展望

蒙特卡洛规划

- 显式给出大规模问题的转移模型 $T(s' | s, a)$ 是不实际的
 - 具体的分布函数很难获得
- 状态转移的采样规则往往容易获得
 - 采样规则 $s' \sim T(s' | s, a)$
 - 表现为规划问题的一个仿真器
- 蒙特卡洛树搜索 (MCTS)
 - 蒙特卡洛仿真
 - 最优优先搜索

MCTS 算法流程

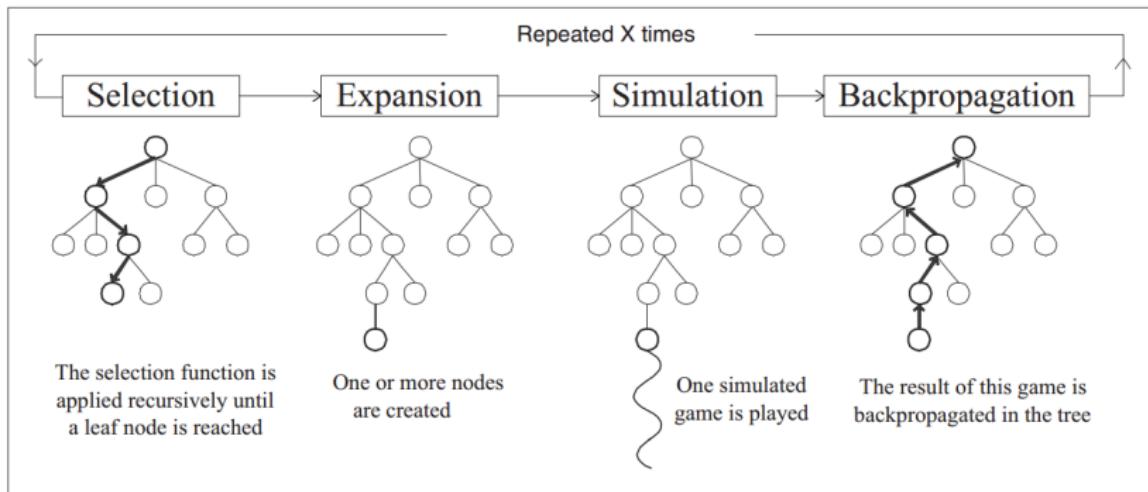


Figure 10 : 蒙特卡洛树搜索主要流程 (Chaslot et al., 2008).

多臂赌博机问题

- MCTS 的决策节点
- 多臂赌博机问题 (MAB)
 - N 个赌博机，即可选动作
 - 未知回报值分布
- 策略：动作—回报历史 \rightarrow 动作
- 累计剩余值 (CR)：

$$R_T = \mathbb{E} \left[\sum_{t=1}^T (X_{a^*} - X_{a_t}) \right] \quad (14)$$

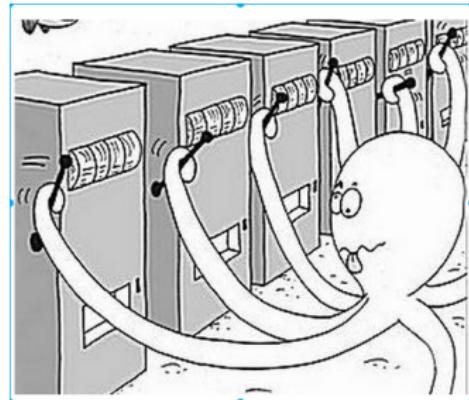


Figure 11 : MAB 问题

探索和利用困境

- MAB 问题的主要挑战
 - 探索和利用之间的平衡
 - * 不仅需要选择目前看似最好的动作
 - * 还需持续探索尚未尝试充分的动作

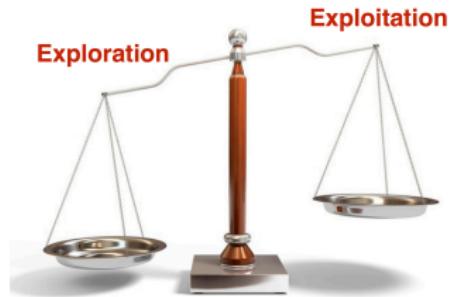


Figure 12 : 探索和利用平衡

Thompson 采样

- MAB 问题的一个渐进最优策略
- 根据一个动作成为最优动作的后验概率来随机选择该动作
 - 两个动作：a、 b
 - $\Pr(a \text{ 最优} \mid \text{历史}) = 0.3, \Pr(b \text{ 最优} \mid \text{历史}) = 0.7$
- 形式上 (Thompson, 1933)：

$$\Pr(a) = \int \mathbf{1} \left[a = \operatorname{argmax}_{a'} \mathbb{E}[X_{a'} \mid \theta_{a'}] \right] \prod_{a'} \Pr(\theta_{a'} \mid Z) d\theta \quad (15)$$

- Z ：行动一回报历史数据
- θ_a ：参数化回报分布 X_a 的未知参数

Thompson 采样 (续)

- 可以使用采样方法高效实现
 - 采样一组隐藏参数 $\theta_a \sim \Pr(\theta_a | Z)$
 - 选择具有最高期望值 $\mathbb{E}[X_a | \theta_a]$ 的动作
- Thompson 采样是近年来 MAB 问题的研究热点
- 实验上效果上比流行的 UCB 算法更好
- 通过指定不同先验分布的方式更好地利用领域知识

研究动机

- 基本想法
 - 分别针对 MDP 和 POMDP 问题
 - 建模 MCTS 搜索树上行动回报的参数化分布
 - 通过贝叶斯方法更新得到参数的后验分布
 - 使用 Thompson 采样进行动作选择

DNG-MCTS 算法

- DNG-MCTS : Dirichlet-NormalGamma MCTS
- $X_{s,\pi}$: 从状态 s 开始服从策略 π 获得的累计回报值
- $X_{s,a,\pi}$: 从状态 s 开始先执行动作 a , 再服从策略 π 获得的累计回报值
- 根据定义 :

$$X_{s,a,\pi} = R(s, a) + \gamma X_{s',\pi} \quad (16)$$

- $s' \sim T(s' | s, a)$

DNG-MCTS 算法 (续)

- 基本假设
 - $X_{s,\pi}$ 服从正态分布 (马尔科夫链上的中心极限定理)
 - $X_{s,a,\pi}$ 服从正态分布的混合分布
- 贝叶斯建模和推理
 - $X_{s,\pi} \sim \mathcal{N}(\mu_s, 1/\tau_s);$
 $(\mu_s, \tau_s) \sim \text{NormalGamma}(\mu_{s,0}, \lambda_s, \alpha_s, \beta_s)$
 - $T(\cdot | s, a) \sim \text{Dirichlet}(\rho_{s,a})$
- 动作选择策略：Thompson 采样
- 以概率 1 找到根节点的最优动作

D²NG-POMCP 算法

- D²NG-POMCP : Dirichlet-Dirichlet-NormalGamma partially observable Monte-Carlo planning
- $X_{b,a}$: 信念状态 b 上执行动作 a 的立即回报
- $X_{s,b,\pi}$: 从联合状态 $\langle s, b \rangle$ 开始服从策略 π 的累计回报
- $X_{b,\pi}$: 从信念状态 b 开始服从策略 π 的累计回报
- 根据定义 :

$$\Pr(X_{b,a} = r) = \sum_{s \in S} \mathbf{1}[R(s, a) = r] b(s) \quad (17)$$

$$f_{X_{b,\pi}}(x) = \sum_{s \in S} b(s) f_{X_{s,b,\pi}}(x) \quad (18)$$

D²NG-POMCP 算法 (续)

- 基本假设
 - $X_{b,a}$ 服从多项分布
 - $X_{s,b,\pi}$ 服从正态分布 (马尔科夫链上的中心极限定理)
 - $X_{b,\pi}$ 服从正态分布的混合分布
- 贝叶斯建模和推理
 - $X_{b,a} \sim \text{Multinomial}(\mathbf{p}_{b,a})$; $\mathbf{p}_{b,a} \sim \text{Dirichlet}(\boldsymbol{\psi}_{b,a})$
 - $X_{s,b,\pi} \sim \mathcal{N}(\mu_{s,b}, 1/\tau_{s,b})$; $(\mu_{s,b}, \tau_{s,b}) \sim \text{NormalGamma}(\mu_{s,b,0}, \lambda_{s,b}, \alpha_{s,b}, \beta_{s,b})$
 - $\Omega(\cdot | b, a) \sim \text{Dirichlet}(\boldsymbol{\rho}_{b,a})$
- 动作选择策略：Thompson 采样
- 以概率 1 找到根节点的最优动作

DNG-MCTS 实验

- MDP 标准测试问题 (基于成本的模型)
 - 加拿大旅行者问题 (CTP)
 - 赛车问题 (Racetrack)
- 主要与基于 UCB 的 UCT 算法作比较
- 评估方法
 - 从当前状态开始迭代运行算法若干次
 - 根据返回的值函数选择执行最优动作
 - 重复以上循环直到终止条件
 - 报告累计折扣成本

加拿大旅行者问题

- 图上的路径搜索问题
- 信息不完整
- 边以一定的概率不能通行
- 建模成信念 MDP 问题
- 状态空间大小： $n \times 3^m$

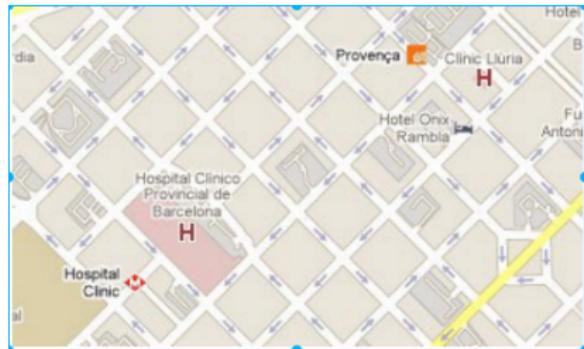


Figure 13 : CTP 问题

加拿大旅行者问题 (续)

Table 2 : 20 节点的 CTP 问题实验结果

实例	状态数	随机的仿真策略		乐观的仿真策略	
		UCT	DNG	UCT	DNG
20-1	20×3^{49}	216.4±3	223.9±4	180.7±3	177.1±3
20-2	20×3^{49}	178.5±2	178.1±2	160.8±2	155.2±2
20-3	20×3^{51}	169.7±4	159.5±4	144.3±3	140.1±3
20-4	20×3^{49}	264.1±4	266.8±4	238.3±3	242.7±4
20-5	20×3^{52}	139.8±4	133.4±4	123.9±3	122.1±3
20-6	20×3^{49}	178.0±3	169.8±3	167.8±2	141.9±2
20-7	20×3^{50}	211.8±3	214.9±4	174.1±2	166.1±3
20-8	20×3^{51}	218.5±4	202.3±4	152.3±3	151.4±3
20-9	20×3^{50}	251.9±3	246.0±3	185.2±2	180.4±2
20-10	20×3^{49}	185.7±3	188.9±4	178.5±3	170.5±3
total		2014.4	1983.68	1705.9	1647.4

赛车问题

- 一组初始状态
- 向终点移动
- 向 8 个方向进行加速
 - 成功概率 : 0.9
 - 失败概率 : 0.1
- 状态空间大小 : 22,534

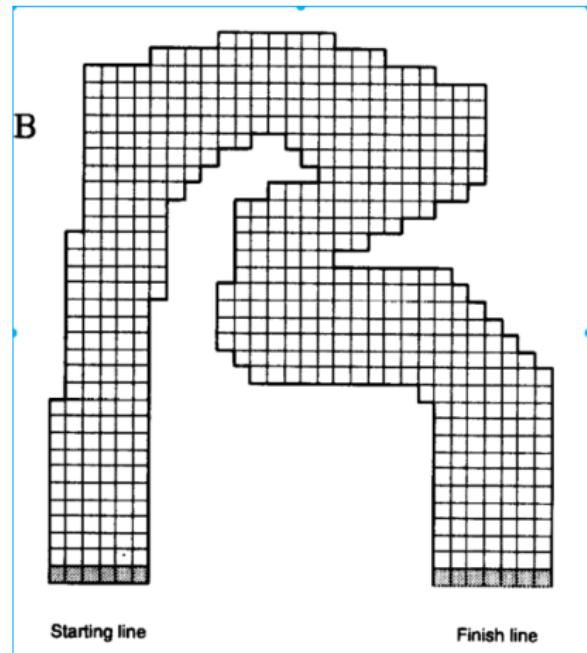


Figure 14 : 赛车问题

赛车问题 (续)

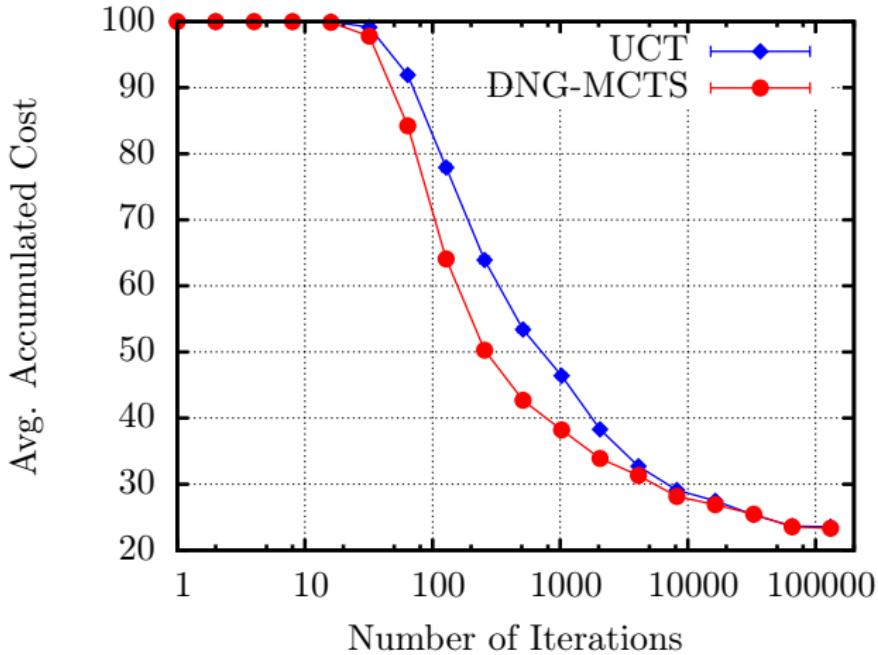


Figure 15 : 赛车问题实验结果

D²NG-POMCP 实验

- POMDP 标准测试问题
 - 岩石采样问题 (RockSample)
 - 吃豆人问题 (PocMan)
- 主要与基于 UCB 的 POMCP 算法作比较
- 评估方法
 - 从当前状态开始迭代运行算法若干次
 - 根据返回的值函数选择执行最优动作
 - 重复以上循环直到终止条件
 - 报告累计折扣回报

RockSample 问题

- 格子世界
- 采样岩石
- 传感器有误差
- RockSample[7,8]
 - 12,545 个状态
 - 13 个动作
 - 2 个观察

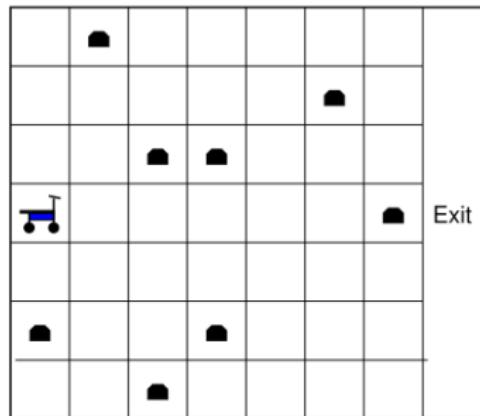


Figure 16 : RockSample[7,8].

岩石采样问题 (续)

Table 3 : RockSample 实验结果 (每个动作允许的计算时间是 1 秒)

RockSample	[7, 8]	[11,11]	[15,15]
States s	12,544	247,808	7,372,800
AEMS2	21.37 ± 0.22	N/A	N/A
HSVI-BFS	21.46 ± 0.22	N/A	N/A
SARSOP	21.39 ± 0.01	21.56 ± 0.11	N/A
POMCP	20.71 ± 0.21	20.01 ± 0.23	15.32 ± 0.28
D ² NG-POMCP	20.87 ± 0.20	21.44 ± 0.21	20.20 ± 0.24

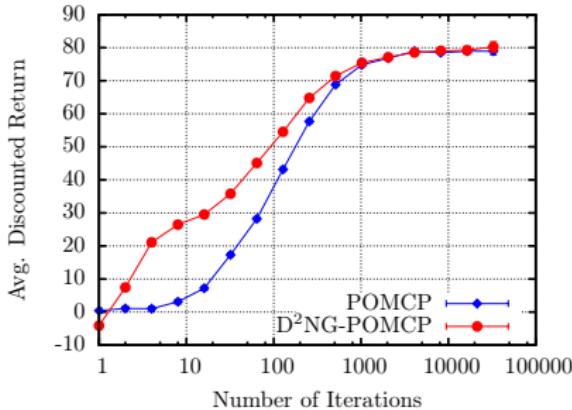
吃豆人问题

- 尽量多地获取食物
- 17×19 迷宫世界
- 4 个巡逻的 Ghost
 - 碰到就会死
- 问题规模
 - 10^{56} 个状态
 - 4 个动作
 - 1,024 个观察

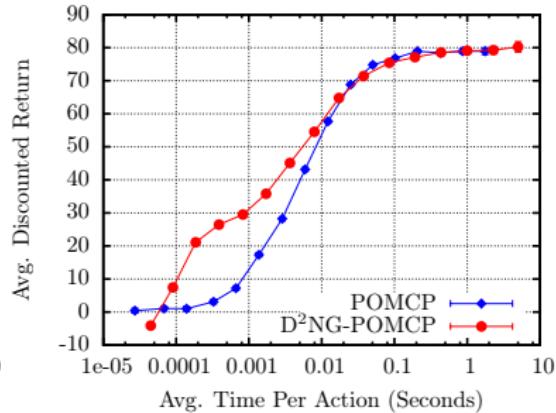


Figure 17 : 吃豆人问题

吃豆人问题 (续)



(a) 回报值 - 迭代次数曲线



(b) 回报值 - 计算时间曲线

Figure 18 : 吃豆人问题实验结果

小结

- 基于后验动作采样的 MDP 和 POMDP 在线规划算法：
DNG-MCTS 和 D²NG-POMCP
 - 建模 MCTS 搜索树上累计回报的参数化分布
 - 根据贝叶斯方法维护分布参数的后验分布
 - 根据 Thompson 采样进行动作选择
- 最优性：以概率 1 找到根节点的最优动作
- 标准测试问题上的实验效果比领域前沿更好

主要内容

- ① 不确定环境下自主规划和感知问题
- ② 基于 MAXQ 分层分解的在线规划算法
- ③ 基于后验动作采样的蒙特卡洛在线规划算法
- ④ 基于集合粒子滤波的多对象跟踪算法
- ⑤ 总结和展望

人-机器人交互中的多人跟踪问题

- 成功的人-机器人交互需要机器人至少能够知道
 - 周围人的数目
 - 每个人的状态信息
- 检测-跟踪（Tracking-by-detection）框架
 - 对象探测器单独地探测每一帧图像中潜在的目标
 - 对象跟踪器根据时序的探测结果更新信念状态

机器人多人跟踪的主要挑战

- 事先不知道真实的人数
- 探测结果不携带 ID 信息
- 不可避免的误报和漏报
- 实时性约束

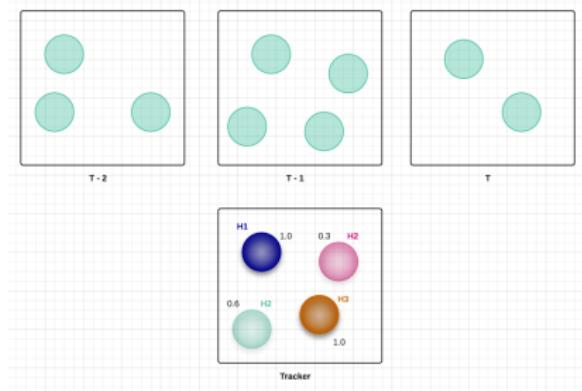


Figure 19 : 多人跟踪问题

CoBot 机器人上的探测结果举例

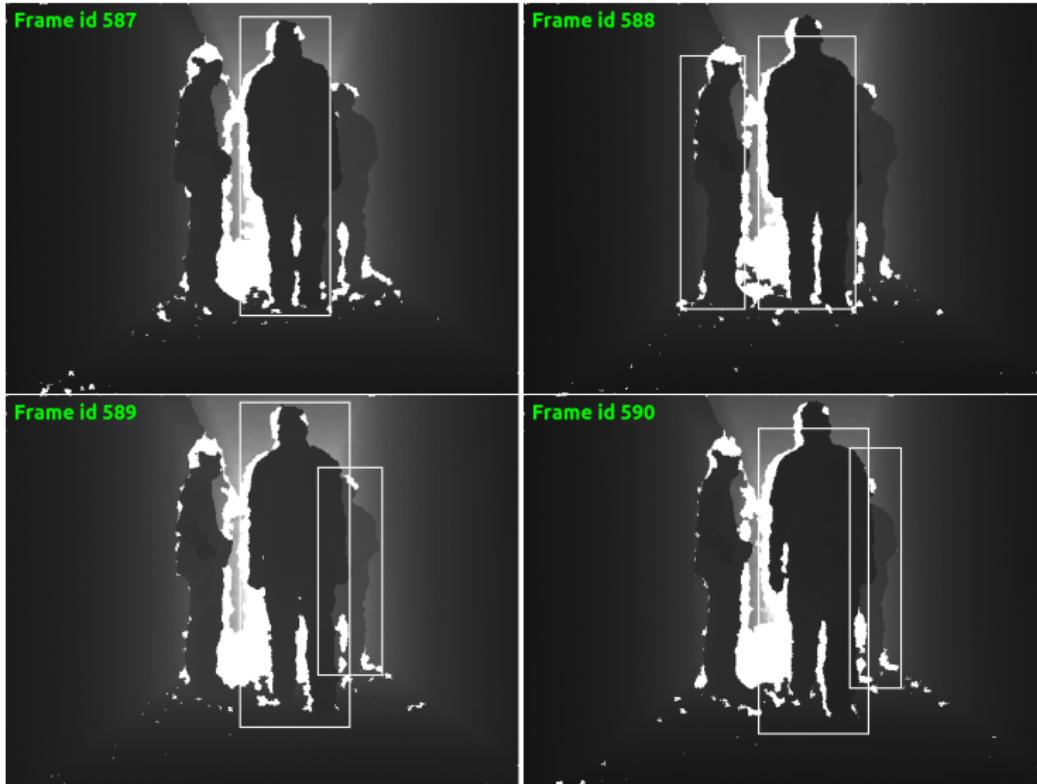


Figure 20 : 连续的原始探测结果

研究动机

- 多对象跟踪领域的大多数方法
 - 假设一个数据关联
 - * 每个探测结果关联到一个潜在对象
 - 对每个潜在对象单独进行贝叶斯更新
 - * 卡尔曼滤波
 - * 粒子滤波
- 缺点：假设出错以后算法准确性骤降，很难恢复
- 想法：不事先假定数据关联，在联合空间里面进行推理

隐马尔科夫过程形式化

- HMM 形式化

- 状态：人的集合 $S = \{s_0, s_1, \dots, s_n\}$
 - 观察：探测结果的集合 $O = \{o_1, o_2, \dots, o_m\}$
 - 联合转移函数： $\Pr(S' | S)$
 - 联合观察函数： $\Pr(O | S)$

- 多人跟踪问题

- 观察历史 \rightarrow 联合状态的后验分布
 - * $\Pr(S_t | O_0, O_1, \dots, O_t)$

运动模型

- 状态为人的集合： $S = \{s_0, s_1, \dots, s_n\}$
 - $s = (x, y, \dot{x}, \dot{y})$
- 假设人与人是互相独立的
- 人数服从生灭过程
- 每个单独的人作随机运动

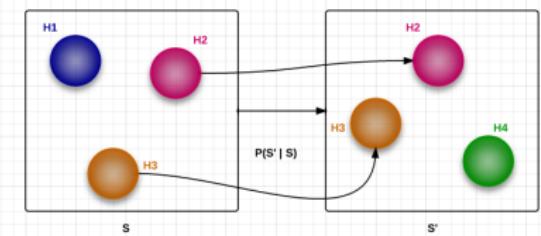


Figure 21 : 运动模型

观察函数

- 状态： $S = \{s_0, s_1, \dots, s_n\}$
- 观察： $O = \{o_0, o_1, \dots, o_m\}$
- 误报和漏报集合： $F \subseteq O$ 、 $M \subseteq S$
 - 约束条件： $|O - F| = |S - M|$
 - 所有 $F-M$ 对： $O \circ S = \{\langle F_0, M_0 \rangle, \dots, \langle F_z, M_z \rangle\}$
- 误报和漏报满足 Poisson 过程： ν 、 $|S|\xi$
- 联合观察函数：

$$\Pr(O | S) = \sum_{\langle F, M \rangle \in O \circ S} \Pr(O - F | S - M) \cdot (\nu\tau)^{|F|} e^{-\nu\tau} \prod_{o \in F} \Pr(o | \emptyset) \frac{(|S|\xi\tau)^{|M|} e^{-|S|\xi\tau}}{|M|!} \frac{1}{\binom{|S|}{|M|}} \quad (19)$$

观察函数——近似计算

- 完整观察函数的项数：

$$\sum_{0 \leq i \leq \min\{|O|, |S|\}} \binom{|O|}{i} \binom{|S|}{i} i! = \Omega\left(\left(\frac{\max\{|O|, |S|\}}{e}\right)^{\min\{|O|, |S|\}}\right)$$

- 剪枝近似：

- 误报一漏报剪枝：根据概率递减的顺序找到 F-M 对，直到某一概率阈值
- 分配剪枝：根据概率递减的顺序找到匹配的状态到观察分配，直到某一概率阈值

集合粒子滤波

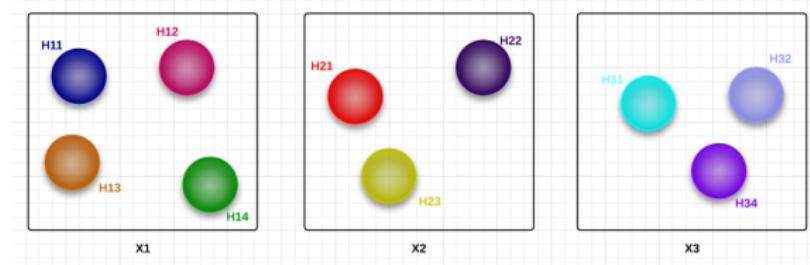


Figure 22 : 联合后验概率的粒子表示

- 一个粒子表示一个联合状态 $X = \{s_0, s_1, \dots, s_n\}$
- 包含 N 个粒子的集合近似表示 $\Pr(S_t | O_t)$:

$$\mathcal{P}_t = \{\langle X_t^{(0)}, w_t^{(0)} \rangle, \langle X_t^{(1)}, w_t^{(1)} \rangle, \dots, \langle X_t^{(N)}, w_t^{(N)} \rangle\} \quad (20)$$

$$- \sum_{i=1}^N w = 1$$

标准数据集测试

- 集合粒子滤波算法 (PFS)
- PETS2009 数据集 (Ferryman & Shahrokni, 2009)
 - 795 帧原始图像、探测结果和真实数据
 - 帧率 ≈ 7 fps
 - 探测结果
 - * 限位框 (x, y, h, w)
 - * 置信度 c
 - 摄像头校正数据
 - * 图像坐标系 \rightarrow 世界坐标系

PETS2009 数据集定性实验结果

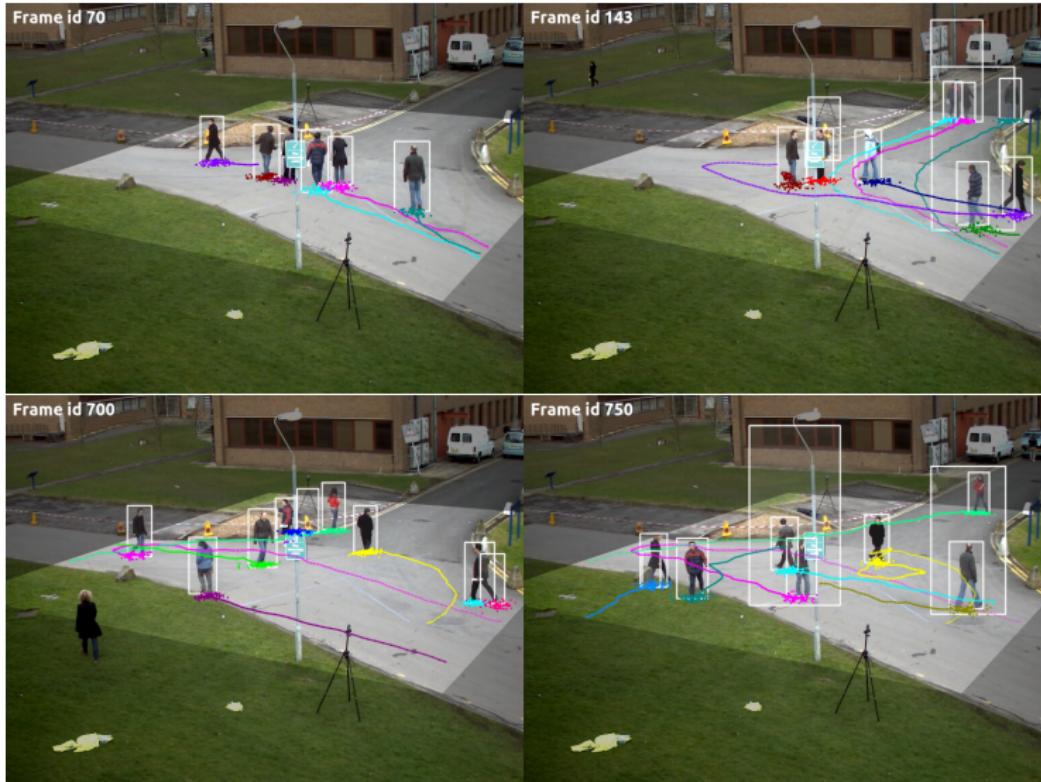


Figure 23 : PETS2009 数据集定性实验结果

PETS2009 数据集定量实验结果

Table 4 : PETS2009 数据集定量实验结果

Algorithm	MOTA 准确度	MOTP 精确度	IDS	MT	FM
PFS ¹ (proposed)	93.1%	76.1%	3.6	18.0	16.0
PFS ¹² (proposed)	90.6%	74.5%	4.8	17.6	20.4
Milan (2014)	90.6%	80.2%	11	21	6
Milan et al. (2013)	90.3%	74.3%	22	18	15
Segal & Reid (2013)	92%	75%	4	18	18
Segal & Reid (2013) ²	90%	75%	6	17	21
Zamir et al. (2012) ²	90.3%	69.0%	8	-	-
Andriyenko & Schindler (2011)	81.4%	76.1%	15	19	21
Breitenstein et al. (2011) ²	56.3%	79.7%	-	-	-

¹16 次运行的平均结果

²全部区域（没有作矩形切割）上的评估结果

真实机器人演示

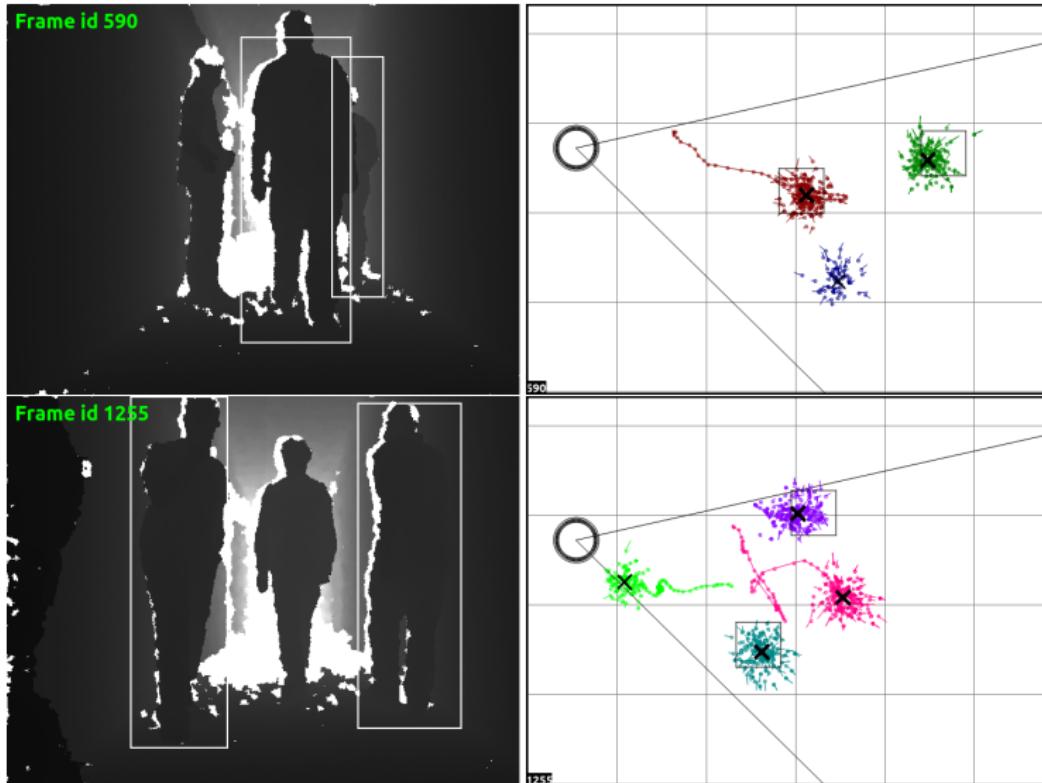


Figure 24 : CoBot 机器人的定性实验结果

小结

- 多对象跟踪问题的集合粒子滤波算法
 - 联合空间上的集合形式化
 - 无需进行显式的观察到目标的数据联合
 - * 相比于传统算法在复杂误报和漏报环境下更稳定
- PETSc2009 数据集上实验结果比领域前沿更好
- CoBot 机器人上的实验进一步表明算法的有效性

主要内容

- ① 不确定环境下自主规划和感知问题
- ② 基于 MAXQ 分层分解的在线规划算法
- ③ 基于后验动作采样的蒙特卡洛在线规划算法
- ④ 基于集合粒子滤波的多对象跟踪算法
- ⑤ 总结和展望

主要创新点

- MAXQ-OP
 - 利用 MAXQ 值函数分解方法进行在线规划
 - 在线近似计算 MAXQ 结构中的完成函数
- DNG-MCTS 和 D²NG-POMCP
 - 使用贝叶斯方法更新动作回报的后验分布
 - 使用 Thompson 采样方法进行动作选择
- PFS
 - 基于集合理论建模多对象跟踪问题
 - 提出联合空间上的粒子滤波算法

不足之处

- MAXQ-OP
 - 分层结构的自动化获取
 - 最优性证明
- DNG-MCTS 和 D²NG-POMCP
 - 理论上的样本复杂度
 - 一般情况下的先验分布
- PFS
 - 有效性的理论证明
 - 基于意图的运动模型

已取得相关论文成果

- Bai, A., Simmons, R., Veloso, M., & Chen, X. (2014a), Intention-aware multi-human tracking for human-robot interaction via particle filtering over sets, In AAAI 2014 Fall Symposium: AI for Human-Robot Interaction (AI-HRI 2014), Arlington, United States
- Bai, A., Wu, F., Zhang, Z., & Chen, X. (2014b), Thompson sampling based Monte-Carlo planning in POMDPs, (In Proceedings of the 24th International Conference on Automated Planning and Scheduling (ICAPS 2014)pp.29–37), Portsmouth, United States
- Bai, A., Wu, F., & Chen, X. (2013a), Bayesian mixture modelling and inference based Thompson sampling in Monte-Carlo tree search, (In Advances in Neural Information Processing Systemspp.1646–1654)
- Bai, A., Wu, F., & Chen, X. (2013b), Towards a principled solution to simulated robot soccer, (In RoboCup 2012: Robot Soccer World Cup XVI vol.7500pp.141–153), Springer
- Bai, A., Wu, F., & Chen, X. (2012b), Online planning for large mdps with maxq decomposition, (In Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 3pp.1215–1216), International Foundation for Autonomous Agents and Multiagent Systems
- Bai, A., Wu, F., & Chen, X. (2012c), Online planning for large MDPs with MAXQ decomposition, In Proc. of the Autonomous Robots and Multirobot Systems workshop (at AAMAS 2012)
- Bai, A., Chen, X., MacAlpine, P., Urieli, D., Barrett, S., & Stone, P. (2012a), WrightEagle and UT Austin Villa: RoboCup 2011 simulation league champions, (In RoboCup 2011: Robot Soccer World Cup XV vol.7416pp.1–12), Springer

致谢

- 感谢各位答辩老师！
- 感谢论文评审老师！
- 感谢导师陈小平老师！
- 感谢答辩秘书陈荣亚！

参考文献 |

- Andriyenko, A., & Schindler, K. (2011), Multi-target tracking by continuous energy minimization, (In Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference onpp.1265–1272), IEEE.
- Bai, A., Chen, X., MacAlpine, P., Urieli, D., Barrett, S., & Stone, P. (2012a), WrightEagle and UT Austin Villa: RoboCup 2011 simulation league champions, (In RoboCup 2011: Robot Soccer World Cup XVvol.7416pp.1–12), Springer.
- Bai, A., Simmons, R., Veloso, M., & Chen, X. (2014a), Intention-aware multi-human tracking for human-robot interaction via particle filtering over sets, In AAAI 2014 Fall Symposium: AI for Human-Robot Interaction (AI-HRI 2014), Arlington, United States.
- Bai, A., Wu, F., & Chen, X. (2012b), Online planning for large mdps with maxq decomposition, (In Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 3pp.1215–1216), International Foundation for Autonomous Agents and Multiagent Systems.
- Bai, A., Wu, F., & Chen, X. (2012c), Online planning for large MDPs with MAXQ decomposition, In Proc. of the Autonomous Robots and Multirobot Systems workshop (at AAMAS 2012).
- Bai, A., Wu, F., & Chen, X. (2013a), Bayesian mixture modelling and inference based Thompson sampling in Monte-Carlo tree search, (In Advances in Neural Information Processing Systemspp.1646–1654).
- Bai, A., Wu, F., & Chen, X. (2013b), Towards a principled solution to simulated robot soccer, (In RoboCup 2012: Robot Soccer World Cup XVIvol.7500pp.141–153), Springer.
- Bai, A., Wu, F., Zhang, Z., & Chen, X. (2014b), Thompson sampling based Monte-Carlo planning in POMDPs, (In Proceedings of the 24th International Conference on Automated Planning and Scheduling (ICAPS 2014)pp.29–37), Portsmouth, United States.

参考文献 II

- Breitenstein, M. D., Reichlin, F., Leibe, B., Koller-Meier, E., & Van Gool, L. (2011), Online multiperson tracking-by-detection from a single, uncalibrated camera, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(9), 1820–1833.
- Chaslot, G., Bakkes, S., Szita, I., & Spronck, P. (2008), Monte-carlo tree search: A new framework for game AI, In C. Darken, & M. Mateas (Eds.) *Proceedings of the Fourth Artificial Intelligence and Interactive Digital Entertainment Conference*, October 22-24, 2008, Stanford, California, USA, The AAAI Press.
URL <http://www.aaai.org/Library/AIIDE/2008/aiide08-036.php>
- Dietterich, T. G. (1999), Hierarchical reinforcement learning with the maxq value function decomposition, *Journal of Machine Learning Research*, 13(1), 63.
- Ferryman, J., & Shahrokn, A. (2009), PETSc2009: Dataset and challenge, (In 2009 Twelfth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS-Winter)pp.1–6), IEEE.
- Kaelbling, L. P., Littman, M. L., & Cassandra, A. R. (1998), Planning and acting in partially observable stochastic domains, *Artificial Intelligence*, 101(1-2), 99–134.
- Kocsis, L., & Szepesvári, C. (2006), Bandit based Monte-Carlo planning, (In European Conference on Machine Learningpp.282–293).
- Milan, A. (2014), Energy Minimization for Multiple Object Tracking, PhD, TU Darmstadt, Darmstadt.
- Milan, A., Schindler, K., & Roth, S. (2013), Detection-and trajectory-level exclusion in multiple object tracking, (In Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference onpp.3682–3689), IEEE.
- Puterman, M. L. (1994), *Markov Decision Processes: Discrete Stochastic Dynamic Programming*, John Wiley & Sons, Inc.

参考文献 III

- Segal, A. V., & Reid, I. (2013), Latent data association: Bayesian model selection for multi-target tracking, (In Computer Vision (ICCV), 2013 IEEE International Conference onpp.2904–2911), IEEE.
- Silver, D., & Veness, J. (2010), Monte-Carlo planning in large POMDPs, (In Advances in Neural Information Processing Systemspp.2164–2172).
- Thompson, W. R. (1933), On the likelihood that one unknown probability exceeds another in view of the evidence of two samples, *Biometrika*, 25, 285–294.
- Zamir, A. R., Dehghan, A., & Shah, M. (2012), Gmcp-tracker: Global multi-object tracking using generalized minimum clique graphs, (In Computer Vision–ECCV 2012pp.343–356), Springer.

求解 MDP

- 策略： $\pi(s) : S \rightarrow A$
- 值函数： $V^\pi(s_0) = \mathbb{E} \left[\sum_{t \geq 0} \gamma^t R(s_t, \pi(s_t)) \right]$
- 贝尔曼最优性等式：

$$V^*(s) = \max_{a \in A} \left\{ R(s, a) + \gamma \sum_{s' \in S} T(s' | s, a) V^*(s') \right\} \quad (21)$$

- 最优策略：

$$\pi^*(s) = \operatorname{argmax}_{a \in A} V^*(s) \quad (22)$$

求解 POMDP

- 策略： $\pi(b) : \mathcal{B} \rightarrow \mathcal{A}$
 - 信念空间： $\mathcal{B} = \{b\}$

- 贝尔曼最优性等式：

$$V^*(b) = \max_{a \in A} \left\{ r(b, a) + \gamma \sum_{o \in O} \Omega(o | b, a) V^*(\zeta(b, a, o)) \right\} \quad (23)$$

- 信念空间上的回报函数： $r(b, a) = \sum_{s \in S} b(s) R(s, a)$
- 最优策略：

$$\pi^*(b) = \operatorname{argmax}_{a \in A} V^*(b) \quad (24)$$

RoboCup 2D 建模成 MDP

- 状态空间：包含所有 23 个对象的高维向量空间
- 动作空间：仿真器定义的原子动作
- 转移函数
 - 仿真器定义的原子动作转移模型
 - 基于队友和对手模型的其他智能体状态转移估计
- 回报函数
 - 服务器定义的原始回报函数
 - 每个宏动作的局部回报函数

实验评估：场景测试

- 三个测试版本
 - FULL：基于 MAXQ-OP 的完整球队
 - RANDOM：球可踢但射门无解时，随机选择 Pass 和 Dribble 行为
 - HAND-CODED：球可踢但射门无解时，根据手工策略选择 Pass 和 Dribble 行为
 - * 智能体 3 米以内没有其他对手：Dribble
 - * 智能体 3 米以内有其他对手：Pass

实验评估：场景测试（续）

- 以固定场景作为开始状态测试运行不同版本的球队和相同的对手
- 每次测试在以下情况终止
 - **Success** : WrightEagle 进球
 - **Failure** : 球的 x 坐标 < -10
 - **Timeout** : 持续周期数 > 200
- 重复 100 次场景测试
- 统计不同版本球队的测试结果

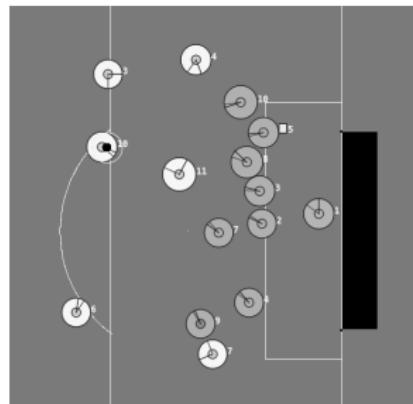


Figure 25 : 开始状态

实验评估：场景测试结果

Table 5 : WrightEagle 重复性场景测试实验结果

版本	试验次数	Success	Failure	Timeout
FULL	100	28	31	41
RANDOM	100	15	44	41
HAND-CODED	100	17	38	45

- 基于 MAXQ-OP 动作选择策略的球队成功的概率最高
- 手工编写的看似合理的策略跟随机策略差不多
 - 手工编写策略具有很大的局限性
 - 进一步说明了 MAXQ-OP 方法的优越性

实验评估：比赛测试

- 评估 FULL 版本球队
- 测试的对手球队
 - BrainsStomers08 (RoboCup 2008 世界冠军)
 - Helios10 (RoboCup 2010 世界冠军)
 - Helios11 (RoboCup 2011 世界亚军)
 - Oxsy11
- 分别进行 100 场比赛，统计比赛结果

实验评估：比赛测试结果

Table 6 : WrightEagle 完整比赛实验结果

对方球队	比赛场数	平均进球	平均积分	获胜概率
BrainsStomers08	100	3.09 : 0.82	2.59 : 0.28	82.0 ± 7.5%
Helios10	100	4.30 : 0.88	2.84 : 0.11	93.0 ± 5.0%
Helios11	100	3.04 : 1.33	2.33 : 0.52	72.0 ± 8.8%
Oxsy11	100	4.97 : 1.33	2.79 : 0.16	91.0 ± 5.6%

- “蓝鹰”有很大的概率可以获胜，达到了很高的竞赛水平

置信区间上界启发值

- 置信区间上界 (UCB) 启发值：

$$UCB(a) = \bar{R}(a) + c \sqrt{\frac{\log T}{N(a)}} \quad (25)$$

- $\bar{R}(a)$ 是执行动作 a 的实验平均回报
- T 是目前为止的所有行动次数
- $N(a)$ 是执行动作 a 的次数
- c 是探索—利用平衡因子

- MAB 问题的渐进最优策略：

$$a^* = \operatorname{argmax}_{a \in A} UCB(a) \quad (26)$$

UCT 算法

- 树上置信区间上界算法 (UCT) (Kocsis & Szepesvári, 2006):

$$UCB(s, a) = \bar{Q}(s, a) + c \sqrt{\frac{\log N(s)}{N(s, a)}} \quad (27)$$

- 以概率 1 找到根节点的最优动作
 - 合适的 c
- 没有原理性的方法可以确定 c 的值

UCT 算法扩展到 POMDP

- POMCP (Silver & Veness, 2010) :

$$UCB(h, a) = \bar{Q}(h, a) + c \sqrt{\frac{\log N(h)}{N(h, a)}} \quad (28)$$

- 以概率 1 找到根节点的最优动作
 - 合适的 c
- 没有原理性的方法可以确定 c 的值

集合粒子滤波 (续)

- 粒子滤波步骤：
 - 粒子提议 : $\hat{X}_t \sim \pi(\cdot | X_{t-1}, O_t)$
 - 粒子更新 : $w_t \leftarrow w_{t-1} \frac{m_t o_t}{p_t}$
 - * 运动权值 : $m_t = \Pr(\hat{X}_t | X_{t-1})$
 - * 观察权值 : $o_t = \Pr(O_t | \hat{X}_t)$
 - * 提议权值 : $p_t = \pi(\hat{X}_t | X_{t-1}, O_t)$
 - 归一化、重采样

粒子改进策略

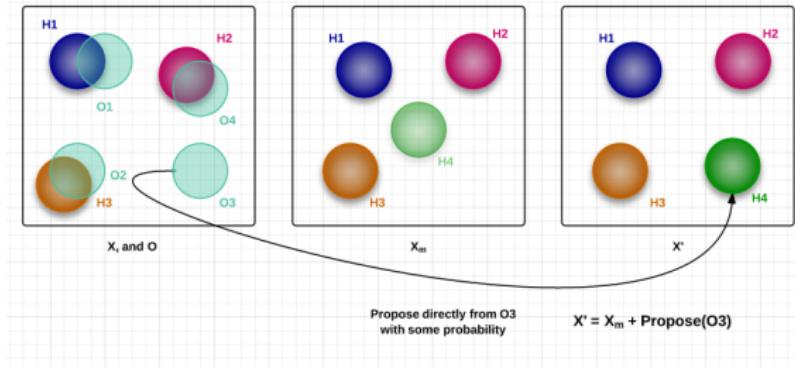


Figure 26 : 粒子改进策略

- 直接由运动模型提议出的粒子匹配观察的概率极低
- 根据运动模型提出粒子，并根据观察进行粒子改进

运动和提议权值近似

- 运动模型的提议： $\mathcal{P}' = \{X' | X' \sim \Pr(\cdot | X), X \in \mathcal{P}\}$
- 改进后的提议： $\mathcal{P}'' = \{X'' | X'' \sim \pi_r(\cdot | X'), X' \in \mathcal{P}'\}$
- 概率密度估计
 - 运动权值： $\Pr(X'' | X) \approx \Pr(X'' | \mathcal{P}')$
 - 提议权值： $\pi_r(X'' | X) \approx \Pr(X'' | \mathcal{P}'')$

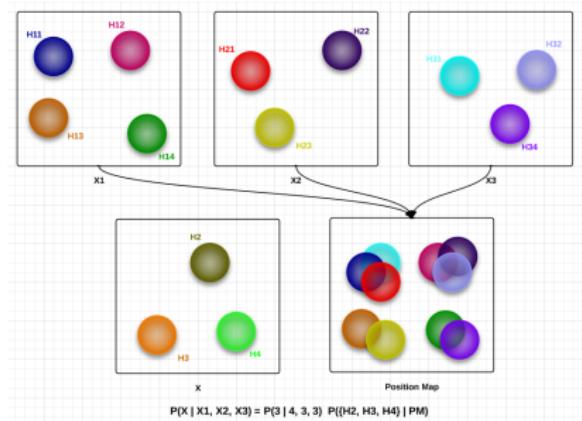


Figure 27 : 概率密度估计

标准数据集测试——参数

	参数	PETS2009
λ	人数出生率 (1/s)	0.0
μ	人数死亡率 (1/s)	0.02
σ_p	加速度大小标准差 (m^2/s)	1.0
ν	误报探测率 (1/s)	6.0
ξ	漏报探测率 (1/s)	2.0
τ	更新时间间隔 (s)	0.14
T'	分配剪枝阈值	0.1
T''	误报—漏报剪枝阈值	0.001
Σ	观察协方差	0.5I
R	报告个体的最小置信度阈值	0.4
N	粒子数目	128
H	最大 EM 迭代次数	10

Table 7 : PETS2009 测试使用的 PFS 参数

CLEAR MOT 指标

- 多对象跟踪准确度 (MOTA)

$$\text{MOTA} = \left(1 - \frac{\sum_t (g_t + a_t - 2n_t + m_t)}{\sum_t g_t} \right) \times 100\% \quad (29)$$

- g_t : 真实目标的数目
- a_t : 算法报告的目标数目
- n_t : 跟真实目标匹配的报告目标的数目
- m_t : ID 交换错误的数目

CLEAR MOT 指标 (续)

- 多对象跟踪精确度 (MOTP)

$$\text{MOTP} = \left(1 - \frac{\sum_t \sum_{1 \leq i \leq n_t} d_t^{(i)}}{\sum_t n_t} \right) \times 100\% \quad (30)$$

- n_t : 跟真实目标匹配的报告目标的数目
- $d_t^{(i)}$: 匹配的报告目标跟真实目标的距离

个体确认

- 个体确认问题

- 输入： $\{X_0, \dots, X_N\}$
- 输出： $\{h_0, \dots, h_z\}$

- 确认的个体

- 状态池：
$$\mathcal{H}(h) \subseteq \mathcal{H}(\mathcal{P}_t)$$
- 状态估计：
$$s = \frac{1}{|\mathcal{H}(h)|} \sum_{s' \in \mathcal{H}(h)} s'$$
- 置信度：
$$c = \frac{|\mathcal{H}(h)|}{N} \in [0, 1]$$

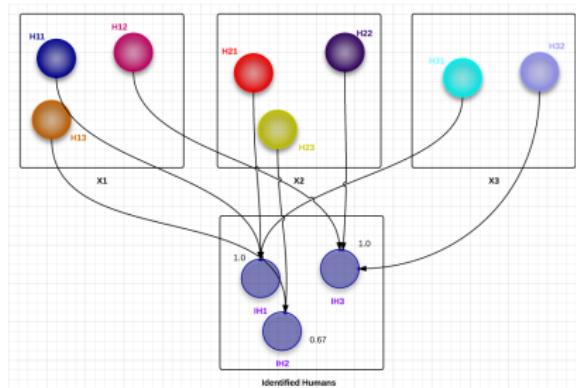


Figure 28 : 个体确认问题

个体确认——形式化

- 周期 t 时的候选个体集合： C_t
- 标记：对 $s \in X$, 其中 $X \in \mathcal{P}_t$, 有 $l(s) \in C_t$
- 状态分布和状态分布集合： f_h 和 $\mathbf{P} = \{f_h \mid h \in C_t\}$
- 个体确认形式化：

$$\mathbf{P}^* = \operatorname{argmax}_{\mathbf{P}} \max_l \Pr(\mathcal{P}_t, l \mid \mathbf{P}) \quad (31)$$

- 期望—最大化 (EM) 求解
 - E-步骤： $\mathbf{P} \rightarrow l$, 退化为 N 个最优分配子问题
 - M-步骤： $l \rightarrow \mathbf{P}$, 极大似然估计 (MLE)
 - 迭代： $l \rightarrow \mathbf{P} \rightarrow l' \rightarrow \mathbf{P}' \rightarrow \dots \rightarrow l^* \rightarrow \mathbf{P}^*$

个体确认——期望—最大化

- E-步骤： $l^{(k)} = \operatorname{argmax}_l \Pr(\mathcal{P}_t, l | \mathbf{P}^{(k-1)})$
 - 找到最大化 $\prod_{s \in X} f_{l(s)}(s)$ 的标记方法
 - 退化为 N 个最优分配子问题
- M-步骤： $\mathbf{P}^{(k)} = \operatorname{argmax}_{\mathbf{P}} \Pr(\mathcal{P}_t, l^{(k-1)} | \mathbf{P})$
 - 极大似然估计 (MLE)
 - 根据当前的观察 O_t 进行近似计算
- $l \rightarrow \mathbf{P} \rightarrow l' \rightarrow \mathbf{P}' \rightarrow \dots \rightarrow l^* \rightarrow \mathbf{P}^*$