一种无线传感器网络中目标跟踪的自适应节点调度算法

胡 波 王祺尧 冯 辉* 罗灵兵
(复旦大学信息科学与工程学院 上海 200433)
(复旦大学智慧网络与系统研究中心 上海 200433)

摘 要：在无线传感器网络目标跟踪的过程中进行节点调度，可以综合考虑跟踪误差和能量消耗，延长传感器网络的使用寿命。为了综合考虑节点调度的短期和长远损失，该文将问题建模为部分可观测马尔科夫决策过程(POMDP)以得到更优的调度策略，并提出一种近似求解算法C-QMDP。该算法利用马尔科夫链蒙特卡洛方法(MCMC)推导连续状态空间的置信状态的转移，并计算瞬时代价。使用状态离散化方法，基于马尔科夫决策过程(MDP)值迭代求解未来代价的近似值。仿真结果表明，相比现有POMDP近似算法，该文算法既可以降低跟踪过程中的累积损失，又可以将大量运算进行离线计算，减小了在线决策时的计算量。

关键词：无线传感器网络；目标跟踪；节点调度；部分可观测马尔可夫决策过程
中图分类号：TP393; TP391 **文献标识码**：A **文章编号**：1009-5896(2018)09-2033-09
DOI: 10.11999/JEIT171154

Adaptive Sensor Scheduling Algorithm for Target Tracking in Wireless Sensor Networks

HU Bo WANG Qiyao FENG Hui LUO Lingbing
(*School of Information Science and Technology, Fudan University, Shanghai* 200433)
(*Research Center of Smart Networks and Systems, Fudan University, Shanghai* 200433)

Abstract: In the process of target tracking, the sensor scheduling algorithm can achieve the tradeoff between the tracking error and the energy consumption so as to extend the service life of the sensor network. The issue can be modeled as a Partially Observable Markov Decision Process (POMDP), which takes both short- and long- term losses of sensor scheduling into account and makes a better decision. A C-QMDP approximation algorithm suitable for continuous state space is proposed. The Markov Chain Monte Carlo (MCMC) method is used to derive the transfer function of belief state and calculate the instantaneous cost. The state discretization method is used to solve the approximation of future cost based on Markov Decision Process (MDP) iteration. Simulation results show that compared to the existing POMDP approximation algorithms, the proposed algorithm can reduce the cumulative losses and computation load in the tracking process by offline computation.

Key words: Wireless Sensor Networks (WSN); Target tracking; Sensor scheduling; Partially Observable Markov Decision Process (POMDP)

1 引言

无线传感器网络(Wireless Sensor Networks, WSN)是由一系列能够对环境做出感知和观测的小型装置通过无线通信组成的自组织网络。由于无线传感器网络成本低廉、易于部署，使得其在监控、安防、交通、定位、医疗等军用和民用领域[1–3]发挥了越来越重要的作用。

目标跟踪[4–7]是无线传感器网络的一个重要应用场景。在实际应用中，传感器所需能量通常由电池提供，对整个传感器网络的电池进行充电或更换十分不易。如果所有传感器在所有时刻都开启观测，虽然可以得到较高的跟踪精度，但同时也会带来较大的能量开销。为了延长传感器网络的工作时间，通常需对传感器节点进行自适应的节点调度[4,5]，或称为节点规划[7]、节点选择[8]或传感器分配[9]等。

收稿日期：2017-12-06；改回日期：2018-05-04；网络出版：2018-07-12
*通信作者： 冯辉 hfeng@fudan.edu.cn
基金项目：国家自然科学基金(61501124)，上海市公安局科学技术发展基金(2017012)
Foundation Items: The National Natural Science Foundation of China (61501124), The Public Security Bureau Science and Technology Development Foundation of Shanghai (2017012)

在每一个决策周期中，节点调度算法选择最优的传感器子集对目标进行观测，以达到跟踪精度和能量功耗的折中。

国内外许多学者对目标跟踪的节点调度算法进行了研究。一些研究将该问题建模为最小化瞬时估计误差或最大化瞬时信息增益的优化问题，建立了基于某个准则下的代价函数，例如熵和相对熵、互信息、先验克拉美罗下界、条件克拉美罗下界、最小均方误差等等[8,10,11]。文献[9]根据目标预测位置，基于势博弈模型求解纳什均衡，达到了较快的收敛速度。这些贪婪算法仅考虑节点调度算法的瞬时表现，没有考虑对未来的影响，因此可被视为“短视”的策略。

在目标跟踪过程中，当前时刻传感器的调度策略会影响传感器对目标的观测结果，该观测结果会影响当前对目标状态的估计，状态估计又会影响下一时刻传感器的调度策略。显然这是一个序贯的决策过程，当前时刻的决策会影响未来的走向，因此必须综合考虑决策的瞬时表现和长远表现。

为了得到长远考虑的优化决策，可以将该问题建模为一个部分可观测马尔科夫决策过程(Partially Observable Markov Decision Process, POMDP)[12]。在POMDP中，由于观测噪声或其他原因，目标状态不能根据观测完全确定，文献中称为部分可观测。POMDP研究的是如何根据历史动作和观测序列，在考虑未来代价的情况下求解最优策略的问题。文献[13]提出了POMDP框架下的类卡尔曼估计，并被应用于传感器网络中的主动目标跟踪[14]。

由于POMDP固有的计算复杂度，只有问题的状态集、动作集和观测集规模很小的情况下，该问题才能被精确求解[12]。为了解决这个问题，学者们提出了许多近似求解POMDP问题的启发式算法[14,15]。当代价函数关于置信状态是线性函数时，近似算法有PBVI, PEMA, Perseus, HSVI, FSVI, SARSOP等，但是这些算法不适用于一般的非线性代价函数的情况。QMDP算法[16]为了处理非线性代价函数，简化未来代价的计算，假设当前状态经过一步转移后，目标状态变为完全可观测，从而基于MDP[17]计算未来代价，但是该算法不能直接应用于状态集是连续空间的情况。对于连续状态空间的情形，He等人[18]将Rollout算法[19]应用到POMDP框架下的节点调度算法中，在计算未来代价时使用基本策略代替最优策略，并采用在线模拟的方法近似Q-value，取得了不错的效果。Li等人[20,21]做出了和QMDP算法相同的假设，即经过一步转移后，目标状态变为完全可观测的，从而使用CO-Rollout方法[19]近似Q-value，求解优化策略，简化了未来代价的计算。然而，Rollout和CO-Rollout算法均使用基本策略代替最优策略，对Q-value的近似精度较低；此外，在线模拟的方式计算复杂度过高，难以应用于工程实践。

本文提出一种近似求解算法C-QMDP，该算法通过两个步骤，分别计算节点调度的瞬时代价和未来代价。对于目标跟踪问题，利用MCMC方法推导置信状态的转移，跟踪目标位置，并计算瞬时代价；对于节点调度问题，将目标的状态空间离散化，基于MDP值迭代求解未来代价的近似值，综合考虑瞬时代价和未来代价选择传感器子集。使用C-QMDP算法进行节点调度的优势有以下几点：(1)将节点调度问题建模为POMDP，综合考虑了决策的瞬时表现和长远表现；(2)可以求解MDP最优策略下各个状态的最小损失来近似Q-value，跟踪更加准确；(3)计算未来损失的过程可以离线计算，将各个状态的最小损失存储起来，在线决策时节约了计算量，提高了系统的实时性。

2 问题建模

本文的研究场景描述如下：在一块区域中，部署了M个位置已知的传感器，用于跟踪单个目标。传感器观测结果包含噪声，并且观测范围有限。目标转移符合近似恒定速度(Nearly Constant Velocity, NCV)模型[22]。存在一个中心处理单元，接收传感器回传的观测数据，进行状态估计和决策，决定下一时刻系统中每个传感器的开关。决策的目标是选择打开一组传感器，综合考虑跟踪精度和传感器功耗，使得跟踪过程累积代价最小。

2.1 场景建模

(1)系统状态：系统状态包含了在2维平面上移动的目标的位置和速度：$\boldsymbol{S}=[x,\dot{x},y,\dot{y}]^{\mathrm{T}}$，其中$[x,y]$表示目标在笛卡尔坐标系下的位置，$\dot{x}$和$\dot{y}$分别表示目标在对应方向上的速度。此外，采用$\boldsymbol{p}_s=[x,y]$表示目标的位置。

(2)动作动作：$\boldsymbol{A}$表示对传感器网络中的节点做出开启或关闭的动作：$\boldsymbol{A}=[a_1,a_2,\cdots,a_M]^{\mathrm{T}}$，$a_m\in\{0,1\}$，其中$M$表示传感器网络中传感器的个数，$a_m\in\{0,1\}$表示关闭或打开第$m$个传感器。

(3)状态转移模型：状态转移模型刻画了目标如何从一个状态转移到另一个状态。假设目标的速度变化缓慢，可以使用近似恒定速度模型[22]：

$$\begin{aligned}\boldsymbol{S}_{k+1}&=\left[x_{k+1},\dot{x}_{k+1},y_{k+1},\dot{y}_{k+1}\right]^{\mathrm{T}}\\&=\begin{bmatrix}1&T_s&0&0\\0&1&0&0\\0&0&1&T_s\\0&0&0&1\end{bmatrix}\boldsymbol{S}_k+\begin{bmatrix}T_s^2/2&0\\T_s&0\\0&T_s^2/2\\0&T_s\end{bmatrix}\\&\quad\cdot\begin{bmatrix}v_x\\v_y\end{bmatrix}\end{aligned}\tag{1}$$

其中，T_s表示决策周期的时长。v_x和v_y是均值为0，方差为σ_x^2和σ_y^2的高斯变量，分别表示x和y方向上的加速度的不确定性。

(4)观测模型：传感器观测模型使用电子通信领域常见的RSSI (Received Signal Strength Indicator)观测模型[23,24]，传感器观测目标发出的能量信号强度与传感器间距离的λ次方成反比(通常$\lambda\geq 2$)。于是观测模型为

$$z_m=\frac{E_0}{\|\boldsymbol{p}_s-\boldsymbol{p}_m\|^{\lambda}}+w_m\tag{2}$$

其中，E_0表示已知的目标信号功率，$\boldsymbol{p}_m$为第m个传感器的位置。w_m是一个均值为0，方差为σ_m^2的高斯噪声，表示传感器m的观测噪声。所有传感器的观测组成的集合可以写作$\boldsymbol{Z}=[z_1,z_2,\cdots,z_M]^{\mathrm{T}}$，其中$z_m$表示第$m$个传感器采集的观测，$k$时刻的观测矢量表示为$\boldsymbol{Z}_k$=$[z_{k,1},z_{k,2},\cdots,z_{k,M}]^{\mathrm{T}}$。

此外，定义$N_m=\dfrac{z_m}{\sigma_m^2}$衡量传感器$m$获得观测$z_m$时的相对接收强度，并定义$N_{\mathrm{th}}$表示传感器的观测阈值。当$N_m<N_{\mathrm{th}}$时，可认为此时观测是无效的，目标位置超出了传感器$m$的观测范围。

事实上，本文算法并不局限于RSSI观测模型，由于后续采用粒子滤波进行状态估计，因此也可运用于其他观测模型。

(5)置信状态：每个时刻可基于历史观测和采取的动作序列得到目标状态的后验分布，该分布被称为目标的置信状态，记作$\boldsymbol{b}$。由于目标状态不可直接观测，因此在POMDP中，使用置信状态$\boldsymbol{b}$的迁移来刻画目标的运动轨迹。k时刻的置信状态$\boldsymbol{b}_k=\mathrm{P}(\boldsymbol{S}_k|\boldsymbol{Z}_{1:k},\boldsymbol{A}_{0:k-1})$，其中$\boldsymbol{Z}_{1:k}$表示时刻1到$k$传感器网络回传的观测序列，$\boldsymbol{A}_{0:k-1}$表示时刻0到$k-1$传感器网络采取的动作序列。在初始时刻，目标置信状态为$\boldsymbol{b}_0$，表示目标状态的先验。

(6)瞬时代价函数：瞬时代价函数表示目标在状态转移过程中每一步所产生的瞬时代价，包含两个方面：跟踪误差以及传感器功耗。使用常数α权衡跟踪误差和传感器功耗对最终决策影响的重要性，代价函数可表示为

$$\begin{aligned}L&=\alpha\mathbb{E}\left[\|\hat{\boldsymbol{p}}_s-\boldsymbol{p}_s\|^2\right]+\sum_{m=1}^{M}c_m^{\mathrm{on}}a_m\\&=\alpha\int_{p_s}\|\hat{\boldsymbol{p}}_s-\boldsymbol{p}_s\|^2\,\boldsymbol{b}(\boldsymbol{p}_s)\mathrm{d}\boldsymbol{p}_s+\sum_{m=1}^{M}c_m^{\mathrm{on}}a_m\end{aligned}\tag{3}$$

其中，$\hat{\boldsymbol{p}}_s=[\hat{x},\hat{y}]$表示目标位置的估计，$c_m^{\mathrm{on}}$表示第$m$个传感器在单位决策时间内开启的功耗。期望是对$\boldsymbol{p}_s=[x,y]$求取的，根据当前置信状态$\boldsymbol{b}$，可以得到目标真实位置的后验分布$\boldsymbol{b}(\boldsymbol{p}_s)$。$\hat{\boldsymbol{p}}_s$是该后验分布的均值，可视为置信状态$\boldsymbol{b}$的函数，因此代价函数式(3)相对于置信状态$\boldsymbol{b}$是非线性的。

2.2 POMDP求解过程

POMDP的迭代过程如下：在k时刻，目标真实状态为$\boldsymbol{S}_k$时，得到观测$\boldsymbol{Z}_k$，此时可计算出目标状态的后验，即置信状态$\boldsymbol{b}_k$，根据置信状态进行决策，采取动作$\boldsymbol{A}_k$并获得代价L_k。然后，系统状态将转移到$\boldsymbol{S}_{k+1}$，得到观测$\boldsymbol{Z}_{k+1}$，从而置信状态转移到$\boldsymbol{b}_{k+1}$，再进行k+1时刻的决策。节点调度问题k时刻贝叶斯网络如图1所示，灰色节点表示已知变量，白色节点表示未知变量。

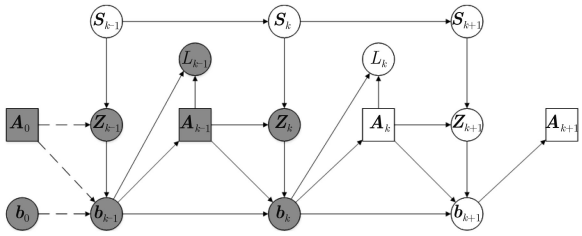


图 1 贝叶斯网络

在POMDP框架下，节点调度的目标是，寻找一个策略，即一个置信状态$\boldsymbol{b}$到动作$\boldsymbol{A}$的映射π，使得能够根据目标的置信状态，动态选取最优的传感器子集对目标进行观测，从而最小化累积损失的期望U。若已给定策略π，则在第k步时，累积损失的期望U为

$$\begin{aligned}U(\boldsymbol{b}_k,\pi)&=\mathbb{E}\left(\sum_{t=k}^{\infty}\beta^{t-k}L\left(\boldsymbol{b}_t,\boldsymbol{A}_t\right)|\pi,\boldsymbol{b}_k\right)\\&=L(\boldsymbol{b}_k,\boldsymbol{A}_k)+\beta\,\mathbb{E}(U(\boldsymbol{b}_{k+1},\pi)|\boldsymbol{b}_k,\boldsymbol{A}_k)\end{aligned}\tag{4}$$

其中，$\boldsymbol{A}_t=\pi(\boldsymbol{b}_t)$，期望是对所有可能的置信状态和观测求取的。式(4)被称为Bellman方程，$L(\boldsymbol{b}_k,\boldsymbol{A}_k)$表示在置信状态为$\boldsymbol{b}_k$时采取动作$\boldsymbol{A}_k$后当前时刻的损失，$\mathbb{E}(U(\boldsymbol{b}_{k+1},\pi)|\boldsymbol{b}_k,\boldsymbol{A}_k)$表示采取动作$\boldsymbol{A}_k$后未来累积损失的期望，$0\leq\beta<1$为递减因子，用来权衡瞬时代价和未来代价，并使累积损失有界。

为了更方便地在第k步时求解最优动作$\boldsymbol{A}_k$，定义Q-value为$Q(\boldsymbol{b}_k,\boldsymbol{A})$表示在置信状态为$\boldsymbol{b}_k$时采取动作$\boldsymbol{A}$，未来采取最优策略的总期望损失，于是，

$$Q(\boldsymbol{b}_k,\boldsymbol{A})=L(\boldsymbol{b}_k,\boldsymbol{A})+\beta\mathbb{E}(V^*(\boldsymbol{b}_{k+1})|\boldsymbol{b}_k,\boldsymbol{A}) \tag{5}$$

$$\boldsymbol{A}_k=\pi^*(\boldsymbol{b}_k)=\arg\min_{\boldsymbol{A}}Q(\boldsymbol{b}_k,\boldsymbol{A}) \tag{6}$$

其中，$V^*(\boldsymbol{b}_k)$表示置信状态为$\boldsymbol{b}_k$时，当前及之后每步都采取最优策略所能达到的期望最小累积损失，π^*为要寻找的最优策略。式(5)中，Q-value左半部分表示当前损失，右半部分表示未来损失，其中期望是对所有可能的$\boldsymbol{b}_{k+1}$求取的。因此，当置信状态为$\boldsymbol{b}_k$时，如果能够求得每个动作的Q-value，即可采取Q-value最小的动作作为当前置信状态下的最优策略。

3 节点调度算法

前一节基于POMDP框架对传感器网络进行目标跟踪的场景进行了建模，本节将对该问题进行近似求解。在每个决策周期中，传感器得到目标的观测，并产生下一时刻需要采取的动作。这里，要解决两个方面的问题：(1)如何跟踪目标的置信状态；(2)如何根据置信状态产生下一时刻要采取的动作。

首先，考虑如何描述状态迁移的问题。在状态转移模型和观测模型皆为线性方程、噪声均为高斯噪声的场景下，可以使用卡尔曼滤波器求得目标置信状态迁移过程的解析解。在更一般的情况下，线性高斯假设不一定成立，例如本文中的观测模型，此时无法获得解析解。在非线性系统中，粒子滤波[25]是一种常用的近似计算目标置信状态迁移的方法。它使用蒙特卡洛采样近似后验概率的积分，可以解决转移方程或观测方程非线性的问题。

接下来，考虑节点调度问题。根据式(6)，只需找到让$Q(\boldsymbol{b}_k,\boldsymbol{A})$最小的动作，即可保证当前时刻到未来的累积损失的期望最小。因此，决策过程的重点是估计Q-value。精确地求解Q-value是十分困难的，尤其是当状态集、观测集较大的时候，计算复杂度十分庞大。不过，Q-value并不需要被精确计算，因为Q-value的意义在于度量动作的好坏，寻找最优的动作。即使Q-value近似有误差，只要依然能够通过Q-value的大小对动作的优劣进行排序，选出最优动作即可。本文基于C-QMDP的算法框架进行Q-value近似，离线计算未来损失，在线计算当前损失，可以大大降低计算复杂度，提高决策的实时性。

3.1 求解当前时刻瞬时代价

在每个决策周期中，需要根据现有观测，更新对目标状态的估计。在第1节中，已经定义了置信状态$\boldsymbol{b}_k=\mathrm{P}(\boldsymbol{S}_k|\boldsymbol{Z}_{1:k},\boldsymbol{A}_{0:k-1})$来表示$k$时刻的目标状态的后验分布。将$(\boldsymbol{Z}_{1:k-1},\boldsymbol{A}_{0:k-1})$记作$\boldsymbol{I}_k$，表示$k$时刻之前的历史观测和动作。由贝叶斯迭代[26]公式，当没有获得k时刻的观测$\boldsymbol{Z}_k$时，根据上一时刻的置信状态$\boldsymbol{b}_{k-1}$，可以得到k时刻目标状态$\boldsymbol{S}_k$的预测分布：

$$\begin{aligned}\mathrm{P}(\boldsymbol{S}_k|\boldsymbol{I}_k)&=\int_{\boldsymbol{S}_{k-1}}\mathrm{P}(\boldsymbol{S}_k|\boldsymbol{S}_{k-1},\boldsymbol{I}_k)\mathrm{P}(\boldsymbol{S}_{k-1}|\boldsymbol{I}_k)\mathrm{d}\boldsymbol{S}_{k-1}\\&=\int_{\boldsymbol{S}_{k-1}}\mathrm{P}(\boldsymbol{S}_k|\boldsymbol{S}_{k-1})\boldsymbol{b}_{k-1}\mathrm{d}\boldsymbol{S}_{k-1}\end{aligned} \tag{7}$$

得到k时刻观测$\boldsymbol{Z}_k$后，可以求得目标状态$\boldsymbol{S}_k$的后验分布，即

$$\begin{aligned}\boldsymbol{b}_k&=\mathrm{P}(\boldsymbol{S}_k|\boldsymbol{Z}_k,\boldsymbol{I}_k)=\frac{1}{\gamma}\mathrm{P}(\boldsymbol{Z}_k|\boldsymbol{S}_k,\boldsymbol{I}_k)\mathrm{P}(\boldsymbol{S}_k,\boldsymbol{I}_k)\\&=\frac{1}{\gamma}\mathrm{P}(\boldsymbol{Z}_k|\boldsymbol{S}_k,\boldsymbol{A}_{k-1})\int_{\boldsymbol{S}_{k-1}}\mathrm{P}(\boldsymbol{S}_k|\boldsymbol{S}_{k-1})\boldsymbol{b}_{k-1}\mathrm{d}\boldsymbol{S}_{k-1}\end{aligned} \tag{8}$$

其中，$\gamma=\int_{\boldsymbol{S}_k}\mathrm{P}(\boldsymbol{Z}_k|\boldsymbol{S}_k,\boldsymbol{A}_{k-1})\mathrm{P}(\boldsymbol{S}_k|\boldsymbol{I}_k)\mathrm{d}\boldsymbol{S}_k$。

根据式(8)，可以获得置信状态$\boldsymbol{b}_{k-1}$到$\boldsymbol{b}_k$的状态迁移。由于积分难以获得解析解，粒子滤波使用蒙特卡洛模拟的方式，使用N个粒子去近似置信状态$\boldsymbol{b}_k$，以数值方法求解上述积分：

$$\boldsymbol{b}_k=\mathrm{P}(\boldsymbol{S}_k|\boldsymbol{Z}_k,\boldsymbol{I}_k)\approx\sum_{i=1}^{N}w_k^{(i)}\delta\left(\boldsymbol{S}_k-\boldsymbol{S}_k^{(i)}\right) \tag{9}$$

其中，$\boldsymbol{S}_k^{(i)}$表示第k步时第i个粒子的状态值，$w_k^{(i)}$表示第k步时第i个粒子的权重，δ为狄拉克函数。

粒子滤波的实现有很多种，本文使用常见的SIR滤波器[25]。权值更新公式为$w_k^{(i)}=w_{k-1}^{(i)}\mathrm{P}\left(\boldsymbol{Z}_k|\boldsymbol{S}_k^{(i)}\right)$。假设不同传感器之间的观测噪声相互条件独立，因此有

$$\mathrm{P}\left(\boldsymbol{Z}_k|\boldsymbol{S}_k^{(i)}\right)=\prod_{m=1}^{M}\mathrm{P}\left(z_{k,m}|\boldsymbol{S}_k^{(i)}\right)^{a_{k,m}} \tag{10}$$

其中，$a_{k,m}$表示k时刻传感器m的开关状态，当$a_{k,m}$=1时，传感器m的观测参与计算。

重采样后粒子权重均为$1/N$，即$w_{k-1}^{(i)}=1/N$。因此，权值更新公式可以写作

$$w_k^{(i)}=\frac{1}{N}\prod_{m=1}^{M}\mathrm{P}\left(z_{k,m}|\boldsymbol{S}_k^{(i)}\right)^{a_{k,m}} \tag{11}$$

之后对权值进行归一化：

$$\tilde{w}_k^{(i)}=w_k^{(i)}\Big/\sum_{j=1}^{N}w_k^{(j)} \tag{12}$$

根据当前N个粒子的状态和权重，可以求得当前目标状态的MMSE估计，即后验均值：

$$\hat{\boldsymbol{S}}_k = \sum_{i=1}^{N} \tilde{w}_k^{(i)} \boldsymbol{S}_k^{(i)} \tag{13}$$

基于粒子滤波算法，可以更新目标置信状态的状态迁移。于是，在粒子滤波的时刻k，可以将此时的粒子集$\boldsymbol{b}_k$代入式(3)，求解当前时刻的瞬时代价：

$$L = \alpha \sum_{i=1}^{N} \left\| \hat{\boldsymbol{p}}_{s,k} - \boldsymbol{p}_{s,k}^{(i)} \right\|^2 \tilde{w}_k^{(i)} + \sum_{m=1}^{M} c_m^{\mathrm{on}} a_m \tag{14}$$

其中，$\hat{\boldsymbol{p}}_{s,k}$表示目标状态估计$\hat{\boldsymbol{S}}_k$中的目标位置部分，$\boldsymbol{p}_{s,k}^{(i)}$表示粒子状态$\boldsymbol{S}_k^{(i)}$中的目标位置部分。

3.2 基于C-QMDP算法求解未来代价

根据式(4)，由于$\boldsymbol{b}_k$和$\boldsymbol{A}$都是已知的，可直接计算当前损失。然而，求解未来损失，即$\mathbb{E}(V^*(\boldsymbol{b}_{k+1})|\boldsymbol{b}_k, \boldsymbol{A})$，是十分困难的。

C-QMDP算法中将Q-value近似计算分为两部分，在POMDP的框架下，基于3.1节中粒子滤波算法更新置信状态，将$\boldsymbol{b}_k$和$\boldsymbol{A}$直接代入式(3)计算当前损失。求解未来损失时，假设经过一步转移后，目标状态变为完全可观测的，此时使用目标状态$\boldsymbol{S}$而不是置信状态$\boldsymbol{b}$的迁移来描述目标的移动，未来损失由求解$\mathbb{E}(V^*(\boldsymbol{b}_{k+1})|\boldsymbol{b}_k, \boldsymbol{A})$转变为求解$\mathbb{E}(V^*(\boldsymbol{S}_{k+1})|\boldsymbol{b}_k, \boldsymbol{A})$，其中目标状态用粒子的值表示。

为了求解$\mathbb{E}(V^*(\boldsymbol{S}_{k+1})|\boldsymbol{b}_k, \boldsymbol{A})$，需要计算置信状态为$\boldsymbol{b}_k$时执行动作$\boldsymbol{A}$后的置信状态$\boldsymbol{b}_{k+1}$，即$k$+1时刻$\boldsymbol{S}_{k+1}$可能的位置和概率。由于$k$+1时刻的观测$\boldsymbol{Z}_{k+1}$在当前时刻是未知的，因此需要根据$\boldsymbol{b}_k$和$\boldsymbol{A}$预测$\boldsymbol{Z}_{k+1}$的值，从而计算$\boldsymbol{b}_{k+1}$的预测分布。$\hat{\boldsymbol{S}}_k$经过式(1)一步转移后，作为$k$+1时刻$\boldsymbol{S}_{k+1}$位置的估计。采取动作$\boldsymbol{A}$后，根据式(2)计算$\boldsymbol{Z}_{k+1}$的预测值，代入3.1节中即可计算$k$+1时刻粒子的位置和权重。由于假设计算未来代价时目标完全可观测，因此k+1时刻开始可用粒子集合中的粒子代表目标状态$\boldsymbol{S}$，粒子权重w表示目标在这个状态的概率。在MDP的框架下，使用值迭代离线计算各状态$\boldsymbol{S}$采取最优策略时对应的损失$V^*(\boldsymbol{S})$。于是，置信状态$\boldsymbol{b}_k$采取动作$\boldsymbol{A}$对应的未来损失的期望为

$$\begin{aligned}
&\mathbb{E}\left(V^*(\boldsymbol{S}_{k+1})\middle|\boldsymbol{b}_k, \boldsymbol{A}\right) \\
&\quad = \int_{\boldsymbol{S}_{k+1}} V^*(\boldsymbol{S}_{k+1}) \mathrm{P}(\boldsymbol{S}_{k+1}|\boldsymbol{b}_k, \boldsymbol{A}) \mathrm{d}\boldsymbol{S}_{k+1} \\
&\quad = \sum_{i=1}^{N} \tilde{w}_{k+1}^{(i)} V^*\left(\boldsymbol{S}_{k+1}^{(i)}\right)
\end{aligned} \tag{15}$$

其中，概率$\mathrm{P}(\boldsymbol{S}_{k+1}|\boldsymbol{b}_k, \boldsymbol{A})$是通过蒙特卡洛采样方式计算的，使用粒子的权重表示概率的值。然而，MDP只能直接应用于状态空间离散的情况。为了在连续状态空间中应用MDP算法计算式(15)中的$V^*(\boldsymbol{S}_{k+1}^{(i)})$，可采取将连续状态空间离散化的方法。

基于C-QMDP求解最优策略的步骤如下：

(1)首先，将目标的状态空间进行离散化，以便应用MDP求解。使用网格划分的方法，将目标状态的各个分量在取值范围内等间隔划分。设离散化后的目标状态一共有D个，由$\tilde{\boldsymbol{S}}$表示，则$\tilde{\boldsymbol{S}}_d$表示目标处于第d个状态。

(2)然后，根据状态转移方程，计算离散状态两两之间的转移概率$\mathrm{P}(\tilde{\boldsymbol{S}}'|\tilde{\boldsymbol{S}})$。离散状态共有$D$个，故共需计算$D(D-1)/2$个概率值。

求解转移概率的解析解是困难的，转移概率多重积分的解析解无法获得，可以使用蒙特卡洛采样代替积分运算。当计算转移概率$\mathrm{P}\left(\tilde{\boldsymbol{S}}'|\tilde{\boldsymbol{S}}\right)$时，在状态$\tilde{\boldsymbol{S}}$中等概率采样$\tilde{N}$个点，记作$\left[\tilde{\boldsymbol{S}}^{(1)}, \tilde{\boldsymbol{S}}^{(2)}, \cdots, \tilde{\boldsymbol{S}}^{(\tilde{N})}\right]$。状态$\tilde{\boldsymbol{S}}'$是根据网格等间隔划分出的一个离散状态区域，假设该离散区域所代表的x, y方向上坐标和速度的界限为$[x_{\min}, x_{\max}, \dot{x}_{\min}, \dot{x}_{\max}, y_{\min}, y_{\max}, \dot{y}_{\min}, \dot{y}_{\max}]$。以$x$方向为例，对于采样点$\tilde{\boldsymbol{S}}^{(i)}$，根据状态转移模型式(1)，有

$$\left.\begin{aligned}
&x_{\min} < x^{(i)} + T_s \dot{x}^{(i)} + \frac{T_s^2}{2} v_x < x_{\max} \\
&\dot{x}_{\min} < \dot{x}^{(i)} + T_s v_x < \dot{x}_{\max}
\end{aligned}\right\} \tag{16}$$

其中，$x^{(i)}$,$\dot{x}^{(i)}$,T_s均为已知量，根据式(16)，可以计算出v_x的上下界。由于v_x是高斯分布，因此可以计算出$\mathrm{P}_x^{(i)}\left(\tilde{\boldsymbol{S}}'|\tilde{\boldsymbol{S}}\right)$的值。如果下界大于上界，则该概率为0。进一步，由$x$和$y$方向状态转移概率独立，可得$\mathrm{P}^{(i)}\left(\tilde{\boldsymbol{S}}'|\tilde{\boldsymbol{S}}\right) = \mathrm{P}_x^{(i)}\left(\tilde{\boldsymbol{S}}'|\tilde{\boldsymbol{S}}\right)\mathrm{P}_y^{(i)}\left(\tilde{\boldsymbol{S}}'|\tilde{\boldsymbol{S}}\right)$。最后，离散状态的状态转移概率$\mathrm{P}\left(\tilde{\boldsymbol{S}}'|\tilde{\boldsymbol{S}}\right)$就是所有的采样点求得的转移概率的均值。

(3)接下来，根据代理代价函数[19]，计算每个状态对应的瞬时代价$\tilde{L}$。在POMDP框架中，代价函数是置信状态和动作到代价的映射；在MDP框架中，代价函数是离散的目标状态和动作到代价的映射。因此，不能沿用式(3)中定义的代价函数。

代理代价函数选取的标准是可以有效反映真实代价，计算出的Q-value可用于评判动作的优劣。本文使用累积观测质量的倒数作为代理代价函数，即

$$\tilde{L}(\boldsymbol{S},\boldsymbol{A})=\alpha\frac{1}{\sum_{m=1}^{M}N_m(\boldsymbol{S})a_m}+\sum_{m=1}^{M}c_m^{\mathrm{on}}a_m \tag{17}$$

其中，$N_m(\boldsymbol{S})$表示传感器m在状态$\boldsymbol{S}$处相对接收强度，

$$N_m(\boldsymbol{S})=\frac{E_0}{\|\boldsymbol{p}_s-\boldsymbol{p}_m\|^{\lambda}\sigma_m^2} \tag{18}$$

当计算状态$\tilde{\boldsymbol{S}}_d$的代价时，只需对式(18)作二重积分或使用蒙特卡洛算法即可。

(4)此时，已经求得了MDP框架下所有要素的值。可以使用MDP值迭代[17]的方法，求解各个离散状态对应的期望最小损失。离散状态$\tilde{\boldsymbol{S}}_d$对应的期望最小损失记作$\hat{V}\left(\tilde{\boldsymbol{S}}_d\right)$。

(5)以上步骤均为离线计算。在线决策时，如当前置信状态为$\boldsymbol{b}_k$，根据粒子滤波置信状态迁移得到$\boldsymbol{b}_{k+1}$的粒子集合。根据连续状态离散化的结果，将每一个粒子$\boldsymbol{S}_{k+1}^{(i)}$对应到相应的离散状态$\tilde{\boldsymbol{S}}_{k+1}^{(i)}$中。然后可基于$\hat{V}\left(\tilde{\boldsymbol{S}}_d\right)$，计算出对应的未来损失的期望：

$$\mathbb{E}(V^*(\boldsymbol{S}_{k+1})|\boldsymbol{b}_k,\boldsymbol{A})=\sum_{i=1}^{N}\tilde{w}_{k+1}^{(i)}\hat{V}\left(\tilde{\boldsymbol{S}}_{k+1}^{(i)}\right) \tag{19}$$

求得未来损失后，代入式(5)，由算法1第8行即可计算置信状态$\boldsymbol{b}_k$对应的Q-value的近似值，然后选取使得Q-value最小的动作$\boldsymbol{A}_k$作为当前最优策略。

整体算法过程如表1的算法1所示。

表 1 C-QMDP算法

算法 1 C-QMDP算法

输入：置信状态(包括粒子状态和粒子权重)：

$$\boldsymbol{b}_k=\left(\boldsymbol{S}_k^{(1)},\tilde{w}_k^{(1)},\boldsymbol{S}_k^{(2)},\tilde{w}_k^{(2)},\cdots,\boldsymbol{S}_k^{(N)},\tilde{w}_k^{(N)}\right)$$

输出：最优动作$\boldsymbol{A}$

(1) **function** C-QMDP($\boldsymbol{b}_k$)

(2) $\hat{V}\leftarrow$ MDP_discrete_value_iteration()

(3) **for** all control actions $\boldsymbol{A}$ **do**

(4) $\boldsymbol{b}_{k+1}\leftarrow$ Particle_filter($\boldsymbol{b}_k,\boldsymbol{A}$)

(5) **for** $i=1:N$ **do**

(6) $\tilde{\boldsymbol{S}}_{k+1}^{(i)}\leftarrow\boldsymbol{S}_{k+1}^{(i)}$

(7) **end for**

(8) $Q(\boldsymbol{b}_k,\boldsymbol{A})=L(\boldsymbol{b}_k,\boldsymbol{A})+\beta\sum_{i=1}^{N}\tilde{w}_k^{(i)}\hat{V}\left(\tilde{\boldsymbol{S}}_{k+1}^{(i)}\right)$

(9) **end for**

(10) **return** $\arg\max_{\boldsymbol{A}}Q(\boldsymbol{b}_k,\boldsymbol{A})$

(11) **end function**

4 仿真结果

为了验证基于C-QMDP近似Q-value的节点调度算法的性能，本文与传感器全部开启、文献[27]的最近点方法(Closest Point Approach, CPA)算法、文献[21]的CO-Rollout算法进行仿真比较。选用CO-Rollout算法比较的原因是该算法适用于本文中观测方程与代价函数均为非线性的场景，且同样是将本文场景建模为POMDP问题并求解。传感器全部开启的方法指的是在每个时刻将所有的传感器开启，对目标状态进行观测。CPA是一种“贪婪”算法，在每个决策时间，根据对目标状态的估计，选择距离目标最近的m个传感器对目标进行观测。由于观测能量信号强度与目标和传感器间距离的λ次方成反比，因此选择距离目标较近的传感器进行观测是合理的。但是，CPA也有其不合理性，一方面它没有考虑传感器间的差异，不同传感器具有不同的观测精度和使用功耗，另一方面没有考虑长期损失，只考虑了当前时刻的观测效果。

仿真中，将M=20个观测精度不同的传感器随机分布在160 m×120 m的平面区域内，对移动目标进行跟踪。x轴坐标范围为[–80, 80], y轴坐标范围为[0, 160]。目标根据式(1)定义的转移模型移动，初始位置为(2,7)，初始速度为(1,2)，方差σ_x^2=0.3, σ_y^2=0.5。决策周期T_s=2 s。目标初始能量强度E_0=1000, λ=2。在粒子滤波中，使用N=10000个粒子近似目标的后验分布，更新目标的置信状态。粒子滤波初始化时，从均值为0，方差为3的高斯分布中采样N个粒子，每个粒子是一个包含位置和速度的4维向量。

通过仿真验证C-QMDP算法的跟踪性能，跟踪结果如图2所示。可以看出，当目标在平面区域随机移动时，C-QMDP算法能够在误差允许的情况下，实现对目标位置的实时跟踪。

接下来，比较不同算法下目标跟踪的累积误差，计算50次后取平均，结果如图3所示。跟踪误差的计算公式为$\mathbb{E}\left[\|\hat{\boldsymbol{p}}_s-\boldsymbol{p}_s\|^2\right]$，$k$时刻累积误差$\Delta$为时刻1到$k$跟踪误差的和。可见，传感器全部开启时，拥有较高的跟踪精度。C-QMDP算法近似Q-value进行节点调度，使用较少的传感器组合对目标进行观测，跟踪效果接近传感器全部开启的跟踪精度。CO-Rollout算法计算未来代价时使用基本策略代替最优策略进行估计，跟踪效果略差于C-QMDP算法的跟踪精度。CPA算法只是贪婪地选取距离目标最近的传感器，没有考虑每个传感器的

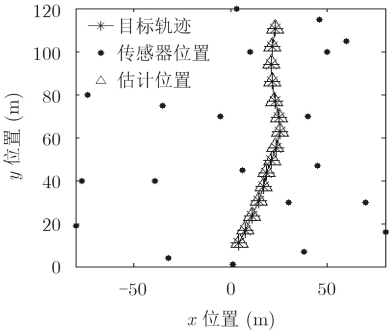


图 2 目标真实轨迹与估计轨迹比较

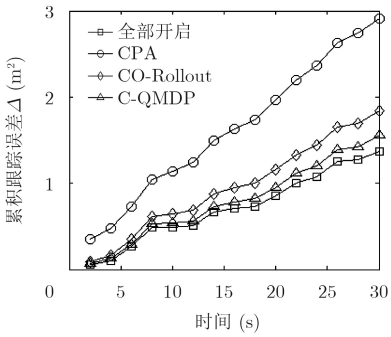


图 3 目标跟踪过程中的累积跟踪误差

特性和长期损失，因此跟踪精度较其余算法更差。不过该算法的实现和计算都十分简单，易于使用。

节点调度算法的目的是对跟踪误差和能量消耗进行权衡。在式(3)中，使用α对两者的量纲进行统一，并对重要性进行权衡。α是一个预先给定的常数，取值应在合理的区间内。仿真中，取α=20，使得跟踪误差和传感器功耗对最终决策的影响大致相当，从而可以较好地体现节点调度算法的作用。传感器每一步的代价为0.1～0.8，不同传感器具有不同的使用代价。比较不同算法目标跟踪过程中的累积总代价L，时刻k的累积总代价表示时刻1到时刻k总代价的和，计算50次取平均，结果如图4所示。

对文中各算法的性能进行定量分析，跟踪误差的计算公式为$\mathbb{E}\left[\|\hat{\boldsymbol{p}}_s-\boldsymbol{p}_s\|^2\right]$，传感器功耗的计算公式为$\sum_{m=1}^{M} c_m^{\mathrm{on}} a_m$，总代价的计算见式(3)，仿真中取$\alpha$=20，计算50次后取平均，各项指标在整个跟踪过程的累积值的比较如表2所示。

由图4和表2可知，虽然将所有传感器全部开启可以获得较高的跟踪精度，但是会带来很大的能量消耗，导致整体的代价很高，累积总代价上涨速度较快。当迭代到第5次后，全部开启的累积总代价超过了CPA算法。通过CO-Rollout和C-QMDP近似Q-value的节点调度算法每次选择了较少的传感器组合进行目标跟踪，能量消耗明显小于全部开启的方法，因此保持了较小的累积总代价。由于CO-

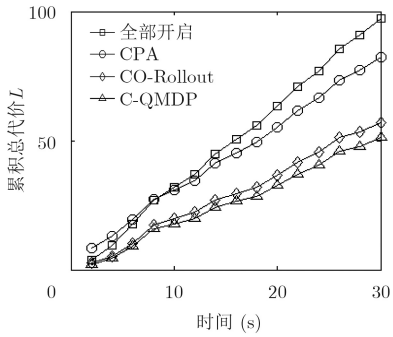


图 4 目标跟踪过程中的累积总代价

表 2　C-QMDP与其余算法性能比较

算法	跟踪误差	传感器功耗	总代价
全部开启	1.3647	70.0722	97.3662
CPA	2.9132	24.0217	82.2857
CO-Rollout	1.8371	20.3493	57.0913
C-QMDP	1.5542	20.0906	51.1746

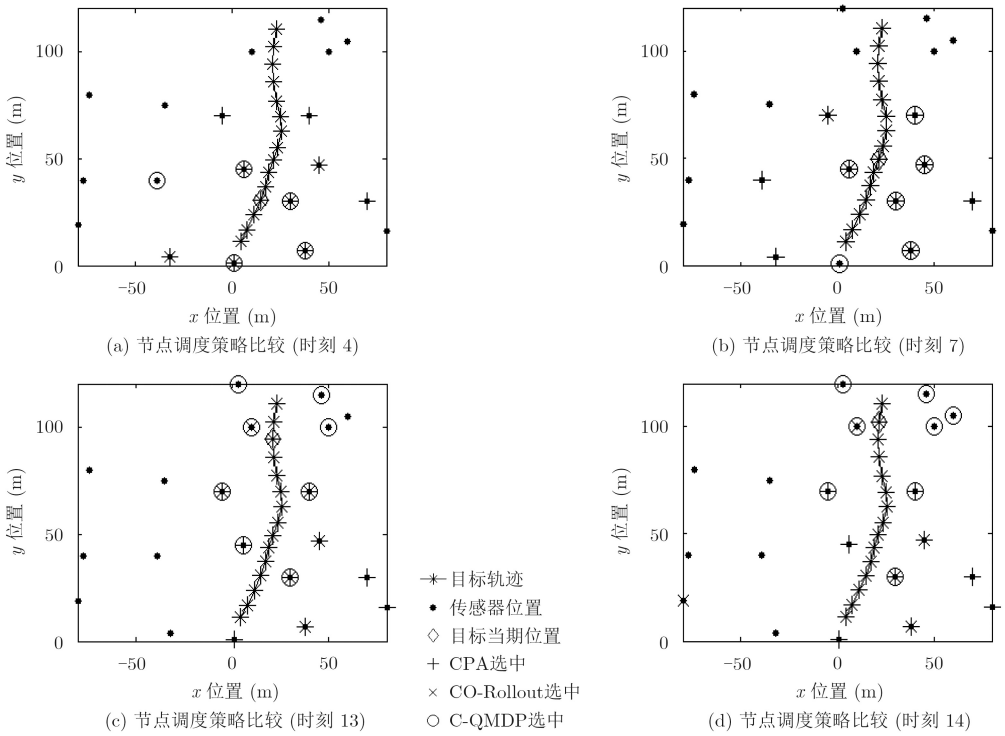
Rollout算法计算未来代价时使用基本策略代替最优策略进行估计，估计误差较C-QMDP算法更大，因此总代价高于基于C-QMDP近似Q-value的节点调度算法。

比较不同算法各时刻的节点调度策略，如图5所示。可以看出，在每个时刻，CPA算法根据对目标位置的估计，选择最近的m个传感器。在各个时刻不同的算法选择了不同数量和位置的传感器子集对目标进行观测，其中C-QMDP算法往往可以选择更加合理的传感器子集。同时，C-QMDP算法可以根据目标位置估计的精确程度，灵活调整传感器子集的数量，当目标位置估计较为精确时，将选用较少数量的传感器以减少功耗；反之，则选用较多数量的传感器以降低跟踪误差。

此外，由于C-QMDP算法将未来代价的计算在线下进行，节约了在线计算时间。因此在相同的粒子数目、传感器数目的情况下，C-QMDP算法相比CO-Rollout算法需要更少的计算资源。

5　结束语

本文将无线传感器网络中的目标跟踪问题建模为部分可观测马尔科夫决策过程(POMDP)，并提出了C-QMDP算法对非线性代价函数的POMDP问题进行近似求解。C-QMDP算法基于粒子滤波推导当前置信状态的转移，然后根据置信状态计算当前损失；计算未来损失时，将目标状态离散化，使用MDP框架的值迭代方法求解最小期望损失。C-QMDP算法一方面提高了计算准确度，得到了更好的传感器调度策略，另一方面将未来损失的计算



(a) 节点调度策略比较 (时刻 4)　(b) 节点调度策略比较 (时刻 7)

(c) 节点调度策略比较 (时刻 13)　(d) 节点调度策略比较 (时刻 14)

图 5 不同时刻节点调度策略比较

转移到线下进行，减小了在线决策时的计算量。仿真结果表明，C-QMDP算法在计算准确度和在线计算时间方面均比CPA和CO-Rollout算法有所改进。

值得注意的是，C-QMDP算法也可适用于其他类似的观测方程和代价函数非线性的序贯决策问题中。如果观测方程线性且噪声是高斯噪声，可以使用卡尔曼滤波代替文中的粒子滤波推导置信状态的迁移，进一步降低计算量和准确性。对于连续状态的离散化，本文使用的是等间隔划分的方法，后续可以研究是否有更好的离散化方法，或采用近似方法直接求解连续状态空间的MDP问题。此外，可以对代理代价函数做更多的尝试，选择更能准确近似真实Q-value的代理代价函数。

参 考 文 献

[1] ALCALA J M, URENA J U, HERNANDEZ A, *et al*. Sustainable homecare monitoring system by sensing electricity data[J]. *IEEE Sensors Journal*, 2017, 17(23): 7741–7749. doi: 10.1109/JSEN.2017.2713645.

[2] MARTELLI T, BONGIOANNI C, COLONE F, *et al*. Security enhancement in small private airports through active and passive radar sensors[C]. 17th IEEE International Conference on Radar Symposium (IRS), Krakow, Poland, 2016: 1–5.

[3] SHI W Y and CHIAO J C. Neural network based real-time heart sound monitor using a wireless wearable wrist sensor[C]. IEEE Conference on Circuits and Systems Conference (DCAS), Arlington, USA, 2016: 1–4.

[4] ANGLEY D, SUVOROVA S, RISTIC B, *et al*. Sensor scheduling for target tracking in large multistatic sonobuoy fields[C]. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Arlington, USA, 2017: 3146–3150.

[5] SONG R, WEI Q, and XIAO W. ADP-based optimal sensor scheduling for target tracking in energy harvesting wireless sensor networks[J]. *Neural Computing and Applications*, 2016, 27(6): 1543–1551. doi: 10.1007/s00521-015-1954-4.

[6] YANG X, ZHANG W A, CHEN M Z Q, *et al*. Hybrid sequential fusion estimation for asynchronous sensor network-based target tracking[J]. *IEEE Transactions on Control Systems Technology*, 2017, 25(2): 669–676. doi: 10.1109/TCST.2016.2558632.

[7] 唐显锭, 冯辉, 杨涛, 等. 无线传感器网络中用于目标跟踪的节点规划算法[J]. 太赫兹科学与电子信息学报, 2014, 12(3): 355–361. doi: 10.11805/TKYDA201403.0355.

TANG Xianding, FENG Hui, YANG Tao, *et al*. Sensor scheduling for target tracking in wireless sensor networks[J]. *Journal of Terahertz Science and Electronic Information Technology*, 2014, 12(3): 355–361. doi: 10.11805/TKYDA201403.0355.

[8] ZHANG H, AYOUB R, and SUNDARAM S. Sensor selection for Kalman filtering of linear dynamical systems: Complexity, limitations and greedy algorithms[J]. *Automatica*, 2017, 78: 202–210. doi: 10.1016/j.auto

matica.2016.12.025.

[9] 冉晓旻, 方德亮. 基于势博弈的分布式目标跟踪传感器分配算法[J]. 电子与信息学报, 2017, 39(11): 2748–2754. doi: 10.11999/JEIT170229.
RAN Xiaomin and FANG Deliang. Distributed sensor allocation algorithm for target tracking based on potential game[J]. *Journal of Electronics & Information Technology*, 2017, 39(11): 2748–2754. doi: 10.11999/JEIT170229.

[10] SINGH P, CHEN M, CARLONE L, *et al*. Supermodular mean squared error minimization for sensor scheduling in optimal Kalman filtering[C]. IEEE Conference on American Control Conference (ACC), Seattle, USA, 2017: 5787–5794.

[11] ASGHAR A B, JAWAID S T, and SMITH S L. A complete greedy algorithm for infinite-horizon sensor scheduling[J]. *Automatica*, 2017, 81: 335–341. doi: 10.1016/j.automatica.2017.04.018.

[12] SPAAN M T J. Partially Observable Markov Decision Processes[M]. Berlin Heidelberg: Springer, 2012: 387–414.

[13] ZOIS D S, LEVORATO M, and MITRA U. Active classification for POMDPs: A Kalman-like state estimator[J]. *IEEE Transactions on Signal Processing*, 2014, 62(23): 6209–6224. doi: 10.1109/TSP.2014.2362098.

[14] ZOIS D S and MITRA U. Active state tracking with sensing costs: Analysis of two-states and methods for *n*-states[J]. *IEEE Transactions on Signal Processing*, 2017, 65(11): 2828–2843. doi: 10.1109/TSP.2017.2664049.

[15] SHANI G, PINEAU J, and KAPLOW R. A survey of point-based POMDP solvers[J]. *Autonomous Agents and Multi-Agent Systems*, 2013, 27(1): 1–51. doi: 10.1007/s10458-012-9200-2.

[16] LITTMAN M L, CASSANDRA A R, and KAELBLING L P. Learning policies for partially observable environments: Scaling up[C]. Proceedings of the 12th International Conference on Machine Learning, Tahoe City, USA, 1995: 362–370.

[17] RUSSELL S. Artificial Intelligence: A Modern Approach. Making Complex Decisions (Ch-17)[M]. Englewood Cliffs: Prentice-Hall, 2004: 645–692.

[18] HE Y and CHONG K P. Sensor scheduling for target tracking in sensor networks[C]. 43rd IEEE Conference on Decision and Control(CDC), Nassau, Bahamas, 2004: 743–748.

[19] CHONG E K P, KREUCHER C M, and HERO III A O. POMDP Approximation Using Simulation and Heuristics[M]. Boston, MA: Springer, 2008: 95–119.

[20] LI Y, KRAKOW L W, CHONG E K P, *et al*. Dynamic sensor management for multisensor multitarget tracking[C]. IEEE 40th Annual Conference on Information Sciences and Systems, Princeton, USA, 2006: 1397–1402.

[21] LI Y, KRAKOW L W, CHONG E K P, *et al*. Approximate stochastic dynamic programming for sensor scheduling to track multiple targets[J]. *Digital Signal Processing*, 2009, 19(6): 978–989. doi: 10.1016/j.dsp.2007.05.004.

[22] BAR-SHALOM Y, LI X R, and KIRUBARAJAN T. Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software[M]. New York: John Wiley & Sons, 2004: 199–266.

[23] ALIPPI C and VANINI G. A RSSI-based and calibrated centralized localization technique for Wireless Sensor Networks[C]. Fourth Annual IEEE International Conference on Pervasive Computing and Communications Workshops, Pisa, Italy, 2006: 301–305.

[24] NIU R and VARSHNEY P K. Target location estimation in sensor networks with quantized data[J]. *IEEE Transactions on Signal Processing*, 2006, 54(12): 4519–4528. doi: 10.1109/TSP.2006.882082.

[25] ARULAMPALAM M S, MASKELL S, GORDON N, *et al*. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking[J]. *IEEE Transactions on Signal Processing*, 2002, 50(2): 174–188. doi: 10.1109/78.978374.

[26] RUSSELL S. Artificial Intelligence: A Modern Approach. Probabilistic Reasoning Over Time (Ch-15)[M]. Englewood Cliffs: Prentice-Hall, 2004: 566–609.

[27] HE Y and CHONG E K P. Sensor scheduling for target tracking: A Monte Carlo sampling approach[J]. *Digital Signal Processing*, 2006, 16(5): 533–545. doi: 10.1016/j.dsp.2005.02.005.

胡　波：男，1968 年生，教授，研究方向为数字信号处理、数字通信和系统设计.

王祺尧：男，1993 年生，硕士生，研究方向为传感器网络、强化学习、序贯决策等研究.

冯　辉：男，1980 年生，副教授，研究方向为分布式信号处理理论与应用.

罗灵兵：男，1992 年生，硕士生，研究方向为图像处理.