**仿真实验结果**

## 模型建立

我们的实验选取两个运动目标和四个传感器（雷达）。传感器0在该区域的中心，其他的3个传感器到传感器0 的距离为30公里，彼此以120度分离。所有四个传感器具有相同的观测范围，，并且具有相同的启动/停止成本0.1。传感器0比其它传感器使用成本更高，传感器3比其它传感器更准确。传感器的概要参数如下：

传感器0：在（0,0）km，花费0.5 /步骤，，，.

传感器1：在（-30，0）km，花费0.1 /步骤，，，。

传感器2：在（15，26）km，花费0.1 /步骤，，，。

传感器3：在（15，-26）km，花费0.1 /步骤，，，。

这两个运动目标由具有不确定加速度的NCV模型决定，在此例中使用的目标运动模型是具有高斯不确定性加速度的近匀速（NCV）模型[15]：

这里的噪音和分别表示目标i在x和y方向上的加速度不确定性。我们假设它们是独立的并且服从均值为0的高斯。

采样间隔为，我们模拟运行300秒。在我们的实验中，目标检测概率为，并且误报率，对应于PF的误报概率。我们在粒子滤波中使用2000个粒子。

## 问题公式化

一个POMDP [7,8]是由它的状态空间X,动作空间U，观测空间Y，状态转换律（X，X’∈X，u∈U），观测律（Z∈Y），初始状态分布P⁰，和一步成本函数r（X，u）指定。它基本上是一个马尔可夫决策过程（MDP），其中状态空间仅仅是通过L部分可观的。

在初始时刻,初始状态X⁰服从已知的分配P⁰，一个POMDP演变如下。在时间步骤k中，该系统的状态是值和观测是已知的。然后选择一个动作会产生一个花费。在此之后，系统根据转化律，和由观测率随机生成观测

由于状态是不能直接观测到的，一个POMDP保持跟踪belief state ，被定义为，以观测和操作历史记录为条件的状态的后验概率分布条件的。这里的目标是要基于 belief state ，从一组可用动作U（）选择一个动作使得预期总成本最小化。策略定义为从belief state 到动作的映射的序列。

让预计总成本从初始belief state 开始并且使用策略,经过H步后得到，其中，以及期望由所有可能的状态和观测序列接管。这里的目的是为了找到最佳策略最小化。

我们定义在状态采取动作u的花费为.则在belief state 采取动作的Q值是

其中是开始于下一个belief state 经过 H -k-1的时间步长的最佳值。贝尔曼对于POMDP问题的最优律表明最低期望总成本是由给出，并且，在第k步选择动作是最优的。因为一个动作的Q值总结了采取这一动作的未来成本，贝尔曼的原则产生了称为”lookahead”的控制方法。

当H很大的时候，最佳的策略可以被假定为静止的。在这种情况下，最佳的策略可以被近似为在每个步骤中仍然剩余H步。因此，最佳的操作是由下式给出

（2）

这种方法被称为receding horizon control。由此产生的最优策略在当前的belief state下选择在h步之后最小化Q值的动作。

在这里所研究的传感器调度的情况下，有M个传感器分布在传感器区域内跟踪T个目标。中央控制器收集并且处理来自这些传感器的数据，并管理传感器的激活状态。我们的目标是选择可以折衷跟踪精度和传感器使用的传感器的数量和组合。

### 系统状态，动作和状态转换律

系统状态向量包含T个目标的状态，M个传感器的状态，和滤波器的状态。在时间步骤k中，系统状态被写成

在这里，是目标的状态，其中包括该目标的位置和在笛卡尔坐标系下的速度。向量元素是传感器的激活状态。如果传感器在时间步骤k中激活，那么= 1;否则 = 0。滤波器状态可以描述任何用于belief state估计的滤波器。由于传感器调度的目标是要权衡传感器成本与跟踪误差，我们所需要滤波器的状态来估计跟踪误差，并且选择用于观测的传感器。在本文所考虑的特定设置下，滤波器是一个粒子滤波器，并且包含N个粒子，每个粒子是目标状态的一个采样

动作是一个M维的向量，其中， = 1或 = 0来指定第m个传感器在时间步骤k+1是否激活以生成在时间k+1基于系统状态的观测。

状态转换律是由状态动态值定义，其中代表在状态转化过程中的随机性。如果我们假设目标独立地移动，则状态动态可以分解成每个目标的状态动态、传感器的状态转变和滤波器的状态转变

传感器状态转换是由给出。滤波器状态的进展是由第3.1节的粒子滤波算法唯一定义

### 单步成本

一步成本结合了目标跟踪误差和传感器的使用成本。在我们的实验中，我们使用

在这里，为目标i的由滤波器状态决定的估计位置，和分别表示上述传感器m的操作成本（每步）和开始/停止成本，是用来调整跟踪误差和传感器成本的相对重要性的权重因子。我们在第4节探讨改变的效果。

### 观测和观测律

观测律取决于传感器模型。我们使用控制跟踪和数据合并的相关联的传感器模型[16]，在该模型中，每个传感器测量由若干观测组成。这些观测既可能是从既定目标有效的测量，或是来自杂波或者目标逃脱的误报。整体观测是来自所有激活的传感器的观测的集合，并且每个传感器扫描可以输出多个观测。假设在时刻k，传感器m的输出的包括个观测，整体观测可以写成

连接每个传感器的是一个覆盖区域。在每个覆盖区域中，测量的质量与目标和传感器之间的距离相关。设表示目标i和传感器m之间的距离。如果或，传感器m不生成来自目标的任何观测。否则，传感器m生成基于目标的状态和检测概率的至多一个测量。

每个单独的观测包括目标或者错误警报的距离，角度和距离变化率。如果由目标生成，然后它取决于目标的如下的状态：

其中，

在这里，和是传感器m的（X，Y）位置，，和代表测量噪声，假定为零均值的高斯。由于测量方差与1 / SNR成比例，并且SNR与成比例[17]，方差，，和正比于。

我们假设误报的数量是具有参数（利率）β的泊松分布。在空间上，误报在整个监测体积上均匀分布。

## 算法流程

### 粒子滤波流程

如果我们忽略控制变量，在每一步belief state可以使用贝叶斯滤波更新，更新分为两个步骤：预测和更新[9,10]。POMDP的特点是在状态转移律和观测映射中的控制变量u。预测步骤由下式给出[18,19]

（6）

更​​新步骤由下式给出

（7）

粒子滤波是上述通过蒙特卡罗模拟的最优递归​​贝叶斯滤波的实现。此方法使用一组N个粒子（采样）来近似belief state。在我们的问题中，只有系统状态（）中的目标状态部分（）的需要被估计，所以每个粒子代表目标的状态：。这里是狄拉克函数，是粒子的重要性权重。

在粒子滤波中，当新的观测变得可知时，滤波器状态递归更新。有许多粒子滤波算法的变中。我们使用重要性重采样（SIR）算法（也称为标准粒子滤波）。在SIR中，滤波器状态更新分为三个步骤完成。首先，根据之前的粒子通过运动学先验分布，对采样产生新的粒子。然后，根据其与观测的吻合度更新粒子权重：

最后，重采样步骤防止粒子退化。

对于多目标跟踪，有一个数据关联的问题：确定哪个观测与哪个目​​标相关联。这个问题来自于相关的测量模型，为了相关练得目标和观测配对，在目标状态上的观测依赖性必须明确给出。在粒子滤波上，这个问题是在多目标的情况下不能直接从得到。为了解决这个问题，我们用联合概率数据关联（JPDA）算法。在JPDA中的目标观测的关联假说的概率计算为已知的目标观测配对提供了和观测律之间的良好联系[20]。

我们首先考虑单一传感器的情况（假设它是传感器m）然后在后面包含多传感器数据融合。设，是所有可能的关联假设。假设的数目，，是由目标数量T和观测数量决定。设是根据关联假设 对于目标在中的观测索引，而0意味着有检测到目标。未分配到任何目标的观测被认为是误报。使用全概率定理，可以计算如下：

（8）

在JPDA中，概率，使用下式计算得到

（9）

这里和分别是在关联中误报和侦测到目标的数量，的是从目标获得观测的可能性，它是由测量噪声的分布来确定的。（见第6章中[21]的细节和例子。）对于2.3节中的传感器模型，从目标在传感器上得到观测的可能性由下式给出

，

（10）

如果多个传感器被同时激活，需要应用多传感器融合来提高跟踪精度。从JPDA扩展到并行执行的多传感器JPDA（MS-JPDA）的扩展由[22]给出。后来，对持续MS-JPDA [23]进行了研究，并已证明其在跟踪性能方面是比并行计算更高效性能更优越。在持续MS-JPDA中，每次只有一个传感器的观测被解析。在每个传感器读数后，JPDA和卡尔曼滤波被用来计算中间belief state估计，然后下一个传感器的观测被用来进一步提高该中间状态估计。在这里，我们融合了持续MS-JPDA和粒子滤波，即在每个传感器读数后我们用粒子滤波而不是卡尔蛮滤波更新滤波器状态

集成持续MS-JPDA和粒子滤波的算法如下：

（ⅰ）初始化。

每个粒子，来自初始分布的采样，并设置，。

（ⅱ）预测。

对于每个粒子，

（a）对于每个目标根据预测

（b）使用下式构造一个新的粒子

（ⅲ）更新权重。

对于每个粒子。

（a）对于每个传感器，使用（8），（9）和（10）计算权重

（b）计算权重。

规一化：对于， 。

（ⅳ）重采样。基于，从选择的N个粒子。

（ⅴ）设，并去预测步骤。

随着粒子的数量变得非常大，则粒子滤波器接近的在（6）和（7）中的确切贝叶斯更新。然而，受限于计算机内存和计算复杂度，在模拟中使用的粒子的数目必然是有限的。这会在测量的方差远小于由粒子分布的方差的时候导致问题。在这种情况下，大部分粒子具有非常小的权重（有时比计算机可处理的最小数更小）除了最接近测量的少数粒子。这就导致在重采样阶段损失粒子多样性。随着时间的推移，跟踪可能会丢失。为了确保跟踪的稳定性，粒子的数目必须足够大以匹配测量精度。对于我们的传感器模型，测量的方差正比于R4并且在一个很宽的范围内变化。当目标非常接近于传感器，需要的粒子数超出了我们的计算能力。我们对于这个问题的解决方案是利用固定数量的粒子（在我们的模拟中是2000）并且轻微的改变似然度计算。如果，其中为在和之间的一些预定义的阈值时，我们在（10）中使用固定，，和（而不是很小，和），人为地增加了粒子的权重和重采样后的多样性。通过这个变化，跟踪器是更稳定，并具有多更小的跟踪误差。

### Rollout 算法流程

在单传感器单目标跟踪问题中，CPA策略是常用的[11,12]。由于测量方差正比于，选择最接近目标的传感器是很自然的。但是CPA是一种“贪婪”的方法，因为它没有考虑到不同的传感器的成本或它们的差错统计。

让我们定义的目标和传感器的有效距离，如果为否则为。对于单目标（目标0）和单传感器的情况，CPA选择使得最小的传感器来观测目标0。此策略可以推广到包括多个目标，即，要选择传感器使得最小的作为多个目标的最近传感器。对于多传感器多目标跟踪问题，我们构造了一个基于CPA策略的最接近传感器策略（CSP），该策略选择动作使得所有目标和它们最近的激活的传感器距离之和最小：

为了激活最多两个传感器来跟踪两个目标，CSP对每个目标激活最接近的传感器。如果对两个目标来说最接近传感器是相同的，只有该传感器被激活。为了基本策略和CO-rollout之间公平比较，通过CO-rollout策略选择激活的传感器的最大数量也是两个。需要注意的是，只要底层目标状态来计算，根据CO-rollout近似算法的要求，在CSP基策略具有映射目标状态到行动的属性。

## 实验结果

我们比较CSP策略（12）（最多激活两个传感器）及使用CSP作为底层基础策略的CO-rollout策略（11）的性​​能。图1分别给出了采用CSP和CO-rollout算法的一个例子目标的真正的轨迹（由实线示出），目标位置估计（用标记示出），和所选择的传感器（用不同的形状标记）。在这个例子中，目标0在(2.5,25)，处开始并且以222m/s的速度向南，目标1开始于(-5,-20)，以120m/s的速度向东移动。我们注意到，在这个例子中我们的多传感器多目标跟踪器效果很好：估计的目标位置非常接近目标轨迹，并且目标交叉处理的很正确。接下来，我们注意到这两项策略在传感器选择的不同之处。首先，因为传感器0的位置是最接近至少其中一个目标的，CSP在所有的时间均激活传感器0，然而考虑到它教高的使用成本，CO-rollout仅在一个短的时间激活传感器0。第二，CSP在目标交叉后当传感器3成为最接近目标0的传感器激活传感器3，而CO-rollout由于其观测的高精确度在目标交叉之前激活传感器3。有时CSP仅选择一个传感器，当它是最接近两个目标的传感器时，但CO-rollout可能会发现激活两个传感器更有益。然而在其他时间，CSP激活2个传感器，而CO-rollout决定一个就够了。

图2展示了四个策略的定量比较：有1或2个传感器激活的CSP，和有1或2个传感器激活的CO-rollout。跟踪误差，传感器的使用，以及总成本是超过100次随机模拟的平均值和90％置信区间。这里的总成本在（4）中定义且。注意到通过激活最多两个传感器，该总成本在CSP中减少了15％以及在CO-rollout中降低30％。我们还可以得出结论，我们的CO-rollout策略单传感器激活提高20％、双传感器激活提高40％。

显然这是一个在跟踪误差和传感器的使用之间的折衷。我们在我们的算法中研究这种权衡并在图3中展示出来。通过调整（4）中的值，优先级可以被放置在跟踪误差或传感器的成本上。这里，结合的优先级得到，跟踪优先级得到，和使用优先级得到。正如我们在图3中看到的，随着减小，跟踪误差增加，而传感器使用减少