降维估计融合的航迹间关联

Robin Forsling*†、Zoran Sjanic*†、Fredrik Gustafsson* 和 Gustaf Hendeby*
* 林雪平大学电气工程系，瑞典林雪平，电子邮箱：{first name.lastname}@liu.se
† Saab AB，瑞典林雪平，电子邮箱：{firstname.lastname}@saabgroup.com

arXiv:2303.00611v2 [eess.SP] 3 Jun 2023

摘要——本文研究通信受限条件下的网络中心化多目标跟踪问题，其中各节点交换降维航迹估计。此前，该领域关于目标跟踪的研究主要关注融合方面，并依据融合性能推导降维的最优方法。本文提出一种新颖的问题形式化方法，根据关联性能对估计值进行降维。我们从理论上分析该问题，并推导其问题性质。理论分析得出一种优化策略，可用于在降低通信估计值维度时部分保持关联质量。数值结果表明，在多目标场景下，所提出的优化策略具有适用性。

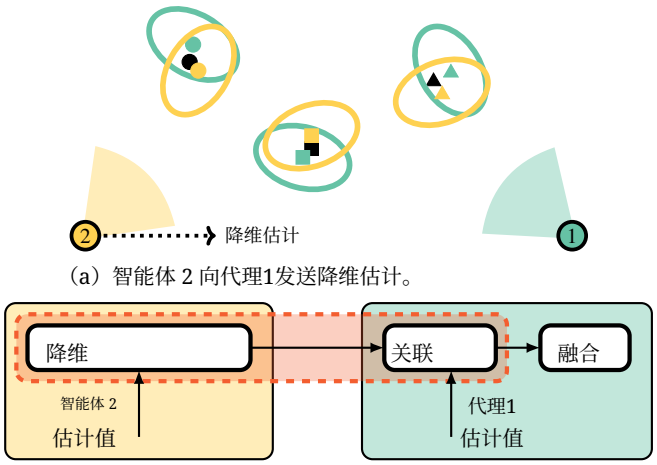
索引词——网络中心化估计、目标跟踪、航迹间关联、通信约束、降维估计。

I. 引言

多目标跟踪（MTT， [1]）问题已得到广泛研究。两种常见的经典多目标跟踪方法是全局最近邻（GNN）跟踪器和多假设跟踪器 [2, 3]。过去几十年来，涌现出多种基于随机有限集的多目标跟踪方法，为多目标贝叶斯滤波器提供了坚实的数学框架，参见例如 [4, 5]。所有这些多目标跟踪算法的一个关键特征，是它们如何处理将测量值分配给现有航迹的关联问题。在网络中心化多目标跟踪中也会出现关联问题：多个智能体对一组共同的目标进行估计，而通信传来的航迹必须与局部航迹关联。这就是航迹间关联问题。此外，通信信道是一种有限资源，在某些情况下，必须对交换的数据进行压缩 [6, 7], ，这通常会对关联质量产生负面影响。

图 1a 展示了一个采用降维估计的网络中心化多目标跟踪场景。此前已有研究探讨了降维测量值和估计值的融合问题：在 [8–10] 中，研究对象为集中式和分布式配置；在 [11–13] 中，研究对象则是去中心化传感器网络。然而，所有这些论文都假设

本研究得到瑞典政府创新系统署（VINNOVA）和 Saab AB 资助的工业卓越中心 LINK-SIC，以及瑞典研究理事会（VR）资助的“可扩展卡尔曼滤波器”项目的支持。G. Hendeby 获得了林雪平大学工业信息技术中心（CENIIT）17.12 号资助。



（a）智能体 2 向代理1发送降维估计。

(b) 所考虑多目标跟踪问题的示意图。

图1. 一种多智能体多目标跟踪场景，其中智能体 2 将降维估计传输给代理1。(a) 中带颜色和编号的圆圈表示智能体。黑色符号表示目标，相应的彩色符号和椭圆表示估计值。在融合收到的降维估计之前，代理1必须将这些估计与其局部估计关联起来。(b) 中红色虚线框标出了本文的研究范围。

关联过程（见图 1b）可以忽略，或可通过简单方法解决，因此降维只需针对融合性能进行优化。相应的关联问题——即在降维过程中考虑数据关联——尚未得到研究。

本文研究网络中心化多目标跟踪中基于降维估计的关联问题。主要目标是找到一种计算降维的方法，以获得令人满意的关联性能。本文对这一问题进行了形式化，并讨论了它与融合最优降维之间的关系。通过问题分析，本文提出了一种优化策略，用于计算能够实现良好关联性能的降维。本文的贡献如下。

- 我们提出了一种针对采用降维估计的网络中心化多目标跟踪中关联问题的新型形式化表述。该问题表述本质上涉及图神经网络跟踪器以及降维的计算，从而获得令人满意的关联质量。

已获准发表于《第26届IEEE信息融合国际会议论文集》，美国南卡罗来纳州查尔斯顿，2023年6月。© 2023 IEEE。允许个人使用本材料。对于任何其他用途，无论是在当前还是未来的媒体中，包括为广告或促销目的重新印刷/再版本材料、创作新的汇编作品、转售或再分发至服务器或列表，或在其他作品中重复使用本作品中任何受版权保护的部分，均须事先获得 IEEE 的许可。

- 我们对所提出的问题进行了理论分析，并推导出问题的性质。
- 基于问题分析，我们提出了一种用于降维计算的优化算法。

II. 使用降维估计的网络中心化目标跟踪

本节介绍所研究的多智能体多目标跟踪问题。所概述的估计模型构成了融合相关操作的基础，且主要与先前的工作有关。本文提出的关联模型是本文贡献的基础。我们给出一个示例，说明降维时为何应考虑关联属性。最后，我们将对所考虑的问题进行形式化描述。

A. 预备知识

令 $\mathbb{R}^n$ 和 $\mathbb{R}^{m\times n}$ 分别表示所有实值 n维向量的集合和所有实值 $m\times n$矩阵的集合。令 $\mathbb{S}^n_+$ 和 $\mathbb{S}^n_{++}$ 分别表示所有对称半正定 $n\times n$ 矩阵的集合和所有对称正定 $n\times n$ 矩阵的集合。

目标和估计值用下标（i）区分，例如，第 i个目标的状态为 $x_{(i)}\in\mathbb{R}^n$。我们用粗体表示随机变量，用常规字体表示随机变量的实现，例如， y 是 $\mathbf{y}$ 的一个实现。期望算子记为 $\mathrm{E}(\cdot)$。若随机变量 $\mathbf{y}$ 的均值为 $\mu=\mathrm{E}(\mathbf{y})$，协方差矩阵为 $\Sigma=\mathrm{E}(\mathbf{y}-\mu)(\mathbf{y}-\mu)^\mathsf{T}$，且 $\mathbf{y}\sim\mathcal{N}(\mu,\Sigma)$，则称 $\mathbf{y}$ 服从高斯分布。

B. 估计模型

我们考虑两个智能体。令

$$y_{1(i)}=x_{(i)}+v_{1(i)},\qquad \mathbf{v}_{1(i)}\sim\mathcal{N}(0,R_{1(i)}),\tag{1a}$$
$$y_{2(i)}=x_{(i)}+v_{2(i)},\qquad \mathbf{v}_{2(i)}\sim\mathcal{N}(0,R_{2(i)}),\tag{1b}$$

分别表示智能体1和智能体2中$x_{(i)}$ 的局部估计。例如，$y_{1(i)}$ 是智能体1中第 i个目标的状态估计， $R_{1(i)}$ 是相应的协方差。假设所有交叉协方差 $R_{12(i)}=\mathrm{E}(\mathbf{v}^{1(i)}\mathbf{v}^\mathsf{T}_{2(i)})$均为零[1]。降维估计表示为

$$y_{\Psi(i)}=\Psi_{(i)}y_{2(i)},\qquad R_{\Psi(i)}=\Psi_{(i)}R_{2(i)}\Psi^\mathsf{T}_{(i)},\tag{2}$$

其中 $\Psi_{(i)}\in\mathbb{R}^{m\times n}$ 满足 $m<n$ ，且秩$(\Psi_{(i)})$为 $=m$。

智能体1和智能体2的局部估计集合为

$$\mathcal{Y}_1=\left\{(y_{1(1)},R_{1(1)}),\ldots,(y_{1(N)},R_{1(N)})\right\},\tag{3a}$$
$$\mathcal{Y}_2=\left\{(y_{2(1)},R_{2(1)}),\ldots,(y_{2(N)},R_{2(N)})\right\}.\tag{3b}$$

代理1和智能体 2跟踪的目标完全相同，因此航迹数量也相同。此外，假设 $\mathcal{Y}_1$ 和 $\mathcal{Y}_2$ 的元素按照$x_{(1)}$，....., $x_{(N)}$进行标记，例如，$(y_{1(i)},R_{1(i)})$和$(y_{2(i)},R_{2(i)})$是对同一目标$x_{(i)}$的估计。这听起来可能有些违背直觉，但这一假设并不构成限制，因为此处

[1]在网络中心化多目标跟踪中，估计值通常存在一定程度的相关性。这里假设估计值在传输之前已进行去相关处理，例如采用 [13, 14]中的技术。

算法1。融合最优 $\Psi_{(i)}$

输入： $R_{1(i)}$、 $R_{2(i)}\in\mathbb{S}^n_{++}$ 和 m

1：令 $Q_{(i)}=R^2_{1\ (i)}$ 和 $S_{(ii)}=R_{1(i)}+R_{2(i)}$。

2：计算 $\lambda_1\le\cdots\le\lambda_n$ 和 z_1，…， z_n，作为以下问题的解

$$Q_{(i)}z=\lambda S_{(ii)}z,$$

3：计算$V=\begin{bmatrix}v_1 & \ldots & v_m\end{bmatrix}$，其中 $v_a^\mathsf{T}v_b=\delta_{ab}$，使得$z_n$，.....， z_{n-m+1} 张成与 V的列相同的子空间。4：计算 $V_\intercal R_2V=U\Sigma U^\mathsf{T}$ 并令 $\Psi_{(i)}=U^\mathsf{T}V^\mathsf{T}$。

输出： $\Psi_{(i)}$

假设实际正确的关联结果已知，如后文所述，任务是计算 $\Psi_{(1)}$，…， $\Psi_{(N)}$。我们还定义

$$\mathcal{Y}_\Psi=\left\{(y_{\Psi(1)},R_{\Psi(1)}),\ldots,(y_{\Psi(N)},R_{\Psi(N)})\right\}.\tag{4}$$

由于 $R_{12(i)}=0$、$(y_{1(i)},R_{1(i)})$ 和 $(y_{\Psi(i)},R_{\Psi(i)})$ 是根据 [11]进行均方误差（ MSE ）最优融合的

$$\hat{x}_{(i)}=P_{(i)}\left(R^{-1}_{1(i)}y_{1(i)}+\Psi^\mathsf{T}_{(i)}R^{-1}_{\Psi(i)}y_{\Psi(i)}\right),\tag{5a}$$
$$P_{(i)}=\left(R^{-1}_{1(i)}+\Psi^\mathsf{T}_{(i)}R^{-1}_{\Psi(i)}\Psi_{(i)}\right)^{-1}.\tag{5b}$$

此融合规则称为卡尔曼融合器（KF）。对于 KF，使用算法 1 [11] 计算融合最优[2] $\Psi_{(i)}$。

C. 关联模型

关联问题被表述为一个线性分配问题 [15]。对于完整估计，指派矩阵为

$$\mathcal{A}_{\text{full}}=\begin{bmatrix}d^2_{(11)} & \cdots & d^2_{(1N)}\\ \vdots & \ddots & \vdots\\ d^2_{(N1)} & \cdots & d^2_{(NN)}\end{bmatrix},\tag{6}$$

其中 $d^2_{(ij)}$ 是马氏距离（MD），其定义为

$$d^2_{(ij)}=\bar{y}^\mathsf{T}_{(ij)}S^{-1}_{(ij)}\bar{y}_{(ij)},\tag{7a}$$
$$\bar{y}_{(ij)}=y_{1(i)}-y_{2(j)},\tag{7b}$$
$$S_{(ij)}=R_{1(i)}+R_{2(j)},\tag{7c}$$

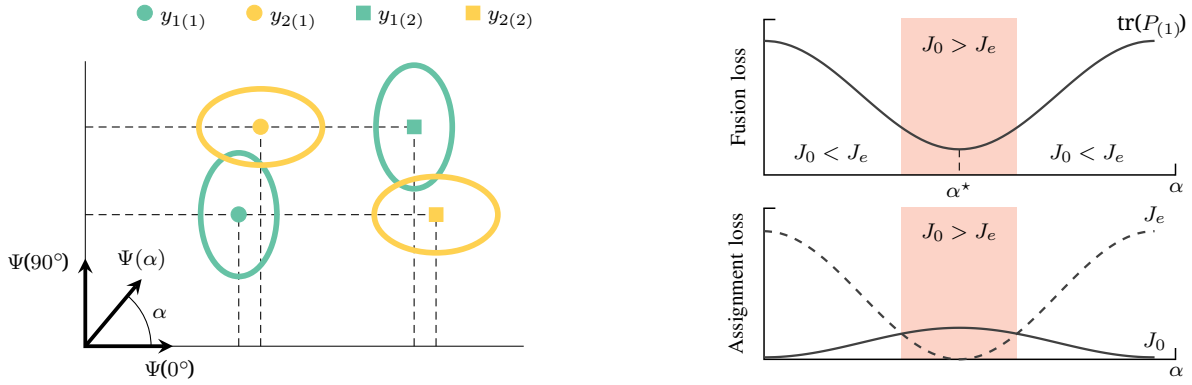
由于 $\mathrm{E}(\mathbf{v}^{1(i)}\mathbf{v}^\mathsf{T}_{2(j)})=0$。类似地，降维指派矩阵 $\mathcal{A}_{\text{red}}$ 定义为

$$\mathcal{A}_{\text{red}}=\begin{bmatrix}r^2_{(11)} & \cdots & r^2_{(1N)}\\ \vdots & \ddots & \vdots\\ r^2_{(N1)} & \cdots & r^2_{(NN)}\end{bmatrix},\tag{8}$$

其中， $r^2_{(ij)}$ 为马氏距离，其定义为

$$\begin{aligned}r^2_{(ij)}&=(\Psi_{(j)}y_{1(i)}-y_{\Psi(j)})^\mathsf{T}\left(\Psi_{(j)}R_{1(i)}\Psi^\mathsf{T}_{(j)}+R_{\Psi(j)}\right)^{-1}\\&\quad\times(\Psi_{(j)}y_{1(i)}-y_{\Psi(j)})\\&=\bar{y}^\mathsf{T}_{(ij)}\Psi^\mathsf{T}_{(j)}\left(\Psi_{(j)}S_{(ij)}\Psi^\mathsf{T}_{(j)}\right)^{-1}\Psi_{(j)}\bar{y}_{(ij)}.\end{aligned}\tag{9}$$

[2]在融合最优意义上，该 $\Psi_{(i)}$ 在融合 $(y_{1(i)},\ R_{1(i)})$ 和 $(y_{\Psi(i)},\ R_{\Psi(i)})$ 时产生最小均方误差。



(a) 目标跟踪示例。 (b) 关于 α的融合与关联损失函数。

图 2. 示例。两个智能体对两个目标进行估计。根据构造，$\Psi_{(1)} = \Psi_{(2)} = \Psi$，其中 $\Psi(\alpha) = \begin{bmatrix}\cos\alpha & \sin\alpha\end{bmatrix}$ 且$\alpha \in [0°, 180°]$。(a)中的虚线表示沿 $\Psi(0°)$ 和 $\Psi(90°)$ 方向的状态估计投影。通过改变 α来评估 Ψ 对融合性能和关联性能的影响。融合损失函数为 $\mathrm{tr}(P_{(i)})$，关联损失函数为 $\mathrm{tr}(\Pi\mathcal{A}_{\mathrm{red}})$，其中 J_0 和 J_e 分别定义为正确分配和错误分配对应的损失。融合最优的 Ψ 由$\alpha^\star = 90°$给出。当 $\alpha^\star$时，$\Psi y_{1(1)} = \Psi y_{2(2)}$ 且 $\Psi y_{1(2)} = \Psi_{2(1)}$，这意味着 $J_e = 0 < J_0$。

代理1接收来自智能体 2 的估计值，并使用以下优化形式求解关联问题。令 $\mathbb{P}^N$ 为所有 $N \times N$ 置换矩阵的集合，即，

$$\mathbb{P}^N = \left\{\Pi \in \mathbb{R}^{N\times N} \;\middle|\; [\Pi]_{ij} \in \{0,1\}, \Pi\Pi^{\mathsf{T}} = I\right\}. \tag{10}$$

置换矩阵 $\Pi \in \mathbb{P}^N$ 将 $\mathcal{Y}_1$ 中的每个估计值分别指派给 $\mathcal{Y}_\Psi$中的恰好一个估计值。对于某个指派矩阵 $\mathcal{A}$，最优 Π 使用 [15]计算得到

$$\begin{aligned} \underset{\Pi}{\text{minimize}} \quad & \mathrm{tr}(\Pi\mathcal{A}) \\ \text{subject to} \quad & \Pi \in \mathbb{P}^N. \end{aligned} \tag{11}$$

在此表述中，正确分配由 $\Pi_0 = I$给出。

注 1. 设 $Z = [z_{ij}]$，其中 $z_{ij} \in \{0,1\}$。(11) 式中的问题是矩阵形式的

$$\begin{aligned} \underset{Z}{\text{minimize}} \quad & z_{ij}[\mathcal{A}]_{ij} \\ \text{subject to} \quad & \sum_i z_{ij} = 1, \quad \forall i, \\ & \sum_j z_{ij} = 1, \quad \forall j. \end{aligned}$$

这种表述在多目标跟踪文献 [16]中更为常见。不过，此处我们使用 (11) 式。

D. 示例

现在我们将说明 $\Psi_{(i)}$ 的选择如何影响关联性能。考虑图 2a 所示的场景，其中 $N = 2$、$n = 2$ 和 $m = 1$。每个智能体对图 2a 中定义的两个目标分别有一个局部估计，其中 $R_{1(1)} = R_{1(2)}$ 和 $R_{2(1)} = R_{2(2)}$。假设

$$\Psi_{(1)} = \Psi_{(2)} = \Psi = \begin{bmatrix}\cos\alpha & \sin\alpha\end{bmatrix},$$

其中 α 是一个角度。基于这种参数化，可以将 $\mathcal{A}_{\mathrm{red}}$ 定义为 α 的函数。令

$$J_0 = \mathrm{tr}(\Pi\mathcal{A}_{\mathrm{red}})|_{\Pi=\Pi_0} = r^2_{(11)} + r^2_{(22)},$$
$$J_e = \mathrm{tr}(\Pi\mathcal{A}_{\mathrm{red}})|_{\Pi=\left[\begin{smallmatrix}0&1\\1&0\end{smallmatrix}\right]} = r^2_{(12)} + r^2_{(21)},$$

分别表示与正确分配和错误分配相对应的代价。根据构造，J_0、J_e 以及 $\mathrm{tr}(P_{(1)}) = \mathrm{tr}(P_{(2)})$都是 α的函数。

通过对每个 $\alpha \in [0°, 180°]$计算 J_0、J_e 和 $\mathrm{tr}(P_{(i)})$，评估关于 α 的融合和关联性能。结果如图 2b 所示。融合最优的 Ψ 对应于 $\alpha^\star = 90°$。然而，此 Ψ位于区间内，其中 $J_0 > J_e$，这将导致分配错误。为在降维情形下实现正确分配，同时保持良好的融合性能，所选的 Ψ 应使 $\mathrm{tr}(P_{(i)})$ 在满足 $J_0 < J_e$的条件下最小化。

E. 问题形式化

假设目标 $x_{(1)}$、…、$x_{(N)}$ 彼此充分分离，因此使用 $\mathcal{A} = \mathcal{A}_{\mathrm{full}}$求解式 (11) 中的分配问题可得到 Π_0。此外，假设智能体 2 不知道关于 $\mathcal{Y}_1$的任何信息。问题是，在智能体 2 处计算$\Psi_{(1)}$、…、$\Psi_{(N)} \in \mathbb{R}^{1\times n}$，使得当智能体 1 使用 $\mathcal{A} = \mathcal{A}_{\mathrm{red}}$ 求解式 (11) 时，解 Π 尽可能接近 Π_0。换言之，由于在降维情形下一般无法获得正确关联，我们希望以不会过度降低关联性能的方式计算 $\Psi_{(1)}$、…、$\Psi_{(N)}$。本文重点关注 $m = 1$的情形。不过，部分结果也适用于任意 $m \geq 1$。

注 2. 所考虑的问题并非网络中心化多目标跟踪中的常见关联问题；在该问题中，接收航迹与局部航迹进行关联，而正确分配Π_0 未知。这里，正确分配是根据构造已知的，因此，为便于阐述，我们可以自由定义 $\mathcal{A}_{\mathrm{full}}$ 和 $\mathcal{A}_{\mathrm{red}}$，使得 $\Pi_0 = I$。

III. 问题分析

本节考察所考虑关联问题的性质，并给出正确分配的充分条件。通过一个示例说明，内在随机性使该问题进一步复杂化。最后推导该问题的统计特性，以供后续章节使用。

A. 正确分配的充分条件

现在从预言机的角度来考虑。第 II-D 节的示例说明了该问题的一个重要特性。也就是说，对于 $\Psi_{(j)} \neq 0$ 和 $\bar{y}_{(ij)} \neq 0$

$$\Psi_{(j)} \perp \bar{y}_{(ij)}^{\mathsf{T}} \iff \Psi_{(j)} \bar{y}_{(ij)} = 0,$$

其中 $\bar{y}_{(ij)} = y_{1(i)} - y_{2(j)}$。由此可以推断，对于我们希望实现的关联

$$\Psi_{(j)} \bar{y}_{(jj)} = 0 \quad \wedge \quad i \neq j \implies \Psi_{(j)} \bar{y}_{(ij)} \neq 0, \tag{12}$$

其中 $\wedge$ $r_{(jj)}^2 = 0$ $r_{(ij)}^2 > 0$ 是逻辑与，因为在这种情况下，而且如果 $i \neq j$。因此，正确分配的充分条件是，对于所有 j，(12) 都成立，因为这将意味着 $\mathrm{tr}(\mathcal{A}_{\mathrm{red}}) = 0$。然而，根据假设，智能体 2 不知道 $\mathcal{Y}_1$，因此在缺乏更多信息的情况下，智能体 2 无法计算 $\Psi_{(j)}$ 以满足 (12)。

B. 问题特性

在上一节的例子中，融合最优 Ψ 导致了错误关联。幸运的是，融合最优 Ψ 通常并不会导致错误分配。遗憾的是，无法对融合与关联性能之间的权衡作出一般性判断。其主要原因如下所述。

考虑 $\Psi_{(j)}^{\mathsf{T}} \in \mathbb{R}^n$，并令 $Q_{(j)} = R_{1(j)}^2 \in \mathbb{S}_{++}^n$ 和 $S_{(jj)} = R_{1(j)} + R_{2(j)} \in \mathbb{S}_{++}^n$。在融合情形下，最优 $\Psi_{(j)}$ 求解 [11]

$$\underset{\|\Psi_{(j)}\|=1}{\text{maximize}} \quad \frac{\Psi_{(j)} Q_{(j)} \Psi_{(j)}^{\mathsf{T}}}{\Psi_{(j)} S_{(jj)} \Psi_{(j)}^{\mathsf{T}}}. \tag{13}$$

因此，针对某个目标 $x_{(j)}$ 的融合最优 $\Psi_{(j)}$ 可以独立于其他目标求解。而在关联问题中，情况并非如此：针对某个目标 $x_{(j)}$ 的最优 $\Psi_{(j)}$，会通过 $\mathcal{A}_{\mathrm{red}}$ 依赖于 $\mathcal{Y}_1$ 和 $\mathcal{Y}_2$ 中的所有估计值。

一个限制稍少的正确分配充分条件（参见式 (12)）是，对每个 j

$$r_{(jj)}^2 < r_{(ij)}^2, \quad \forall i \neq j. \tag{14}$$

若满足此条件，最近邻 [16] 关联将得出与 GNN 关联相同的结果。(14) 中的条件也可以表示为 [11]

$$\frac{\Psi_{(j)} \bar{y}_{(jj)} \bar{y}_{(jj)}^{\mathsf{T}} \Psi_{(j)}^{\mathsf{T}}}{\Psi_{(j)} S_{(jj)} \Psi_{(j)}^{\mathsf{T}}} < \frac{\Psi_{(j)} \bar{y}_{(ij)} \bar{y}_{(ij)}^{\mathsf{T}} \Psi_{(j)}^{\mathsf{T}}}{\Psi_{(j)} S_{(ij)} \Psi_{(j)}^{\mathsf{T}}}, \quad \forall i \neq j, \tag{15}$$

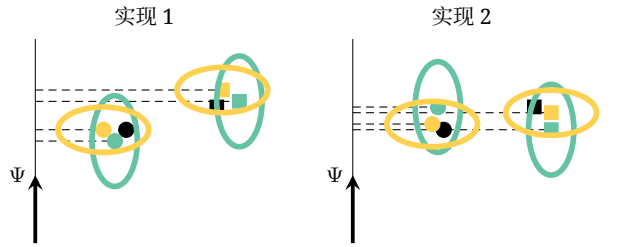


图3。同一场景的两种噪声实现。在两种实现中，目标状态以及协方差 $R_{1(1)} = R_{1(2)}$ 和 $R_{2(1)} = R_{2(2)}$ 均相同。不同的是状态估计值 $y_{1(1)}$、$y_{2(1)}$、$y_{1(2)}$ $y_{2(2)}$ 和 。在实现 1 中得到正确分配，而在实现 2 中得到错误分配。

其中，每个分式在结构上都与 (13) 中的分式相似。不过，与融合情形相比，一个复杂之处在于，$r_{(ij)}^2$ 是一个随机变量的实现

$$\mathbf{r}_{(ij)}^2 = \bar{\mathbf{y}}_{(ij)}^{\mathsf{T}} \Psi_{(j)}^{\mathsf{T}} \left(\Psi_{(j)} S_{(ij)} \Psi_{(j)}^{\mathsf{T}} \right)^{-1} \Psi_{(j)} \bar{\mathbf{y}}_{(ij)},$$

其中 $\bar{\mathbf{y}}_{(ij)} = \mathbf{y}_{1(i)} - \mathbf{y}_{2(j)}$。因此，假设智能体 2 可以访问 $R_{1(i)}$ 并且对 $x_{(i)}$ 有良好的估计，则可以计算出融合最优的 $\Psi_{(i)}$，但由于随机性，仍难以预测 $r_{(ij)}^2$。图3展示了各随机变量的两种可能实现

$$\begin{aligned} \mathbf{y}_{1(1)} &= x_{(1)} + \mathbf{v}_{1(1)}, & \mathbf{y}_{2(1)} &= x_{(1)} + \mathbf{v}_{2(1)}, \\ \mathbf{y}_{1(2)} &= x_{(2)} + \mathbf{v}_{1(2)}, & \mathbf{y}_{2(2)} &= x_{(2)} + \mathbf{v}_{2(2)}, \end{aligned}$$

其中 $\mathbf{v}_{1(1)}$、$\mathbf{v}_{1(2)} \sim \mathcal{N}(0, R_1)$，以及 $\mathbf{v}_{2(1)}$、$\mathbf{v}_{2(2)} \sim \mathcal{N}(0, R_2)$。由于两种情况下协方差相同，且根据假设 $R_{1(1)} = R_{1(2)} = R_1$ 和 $R_{2(1)} = R_{2(2)} = R_2$，因此融合最优的 $\Psi_{(j)}$ 在两种情况下均满足 $\Psi_{(1)} = \Psi_{(2)} = \Psi$。在实现 1 和实现 2 中计算 $\mathcal{A}_{\mathrm{red}}(\Psi)$，得到

$$\mathcal{A}_1 = \begin{bmatrix} 0.05 & 1.01 \\ 0.31 & 0.05 \end{bmatrix}, \qquad \mathcal{A}_2 = \begin{bmatrix} 0.11 & 0.01 \\ 0.01 & 0.11 \end{bmatrix},$$

分别而言。因此，在实现 1 中，我们将得到正确分配 Π_0，而在实现 2 中则会选择错误的组合。这个例子说明，由于内在随机性，通常无法判断融合最优 $\Psi_{(j)}$ 会导致正确分配还是错误分配，除非知道实际实现。

C. 统计特性

假设 $m \geq 1$。根据构造

$$\Psi_{(j)} \bar{\mathbf{y}}_{(ij)} \sim \mathcal{N}\left(\Psi_{(j)} \bar{x}_{(ij)}, \Psi_{(j)} S_{(ij)} \Psi_{(j)}^{\mathsf{T}} \right),$$

其中 $\bar{x}_{(ij)} = x_{(i)} - x_{(j)}$。因此 [17]

$$\mathbf{r}_{(ij)}^2 \sim \begin{cases} \chi_m^2, & \text{if } i = j, \\ \chi_{m,\nu}^2, & \text{if } i \neq j, \end{cases} \tag{16}$$

其中，χ_m^2 是具有 m 个自由度的中心卡方分布，而 $\chi_{m,\nu}^2$ 是非中心卡方

分布，其中 ν 是非中心参数。期望值为

$$\mathsf{E}\left(\mathbf{r}_{(ij)}^2\right) = m + \nu_{(ij)}, \tag{17}$$

其中，$\nu_{(ij)} = \bar{x}_{(ij)}^{\mathsf{T}} \Psi_{(j)}^{\mathsf{T}} \left(\Psi_{(j)} S_{(ij)} \Psi_{(j)}^{\mathsf{T}}\right)^{-1} \Psi_{(j)} \bar{x}_{(ij)}$ 是非中心参数。方差由 [17] 给出

$$\operatorname{var}\left(\mathbf{r}_{(ij)}^2\right) = 2m + 4\nu_{(ij)}. \tag{18}$$

一个结论是，随着 $\nu_{(ij)}$ 增大，随机性的相对影响会减小，因为 $\mathsf{E}(\mathbf{r}_{(ij)}^2)$ 按 $\nu_{(ij)}$ 的比例变化，而 $\sqrt{\operatorname{var}(\mathbf{r}_{(ij)}^2)}$ 仅按 $\sqrt{\nu_{(ij)}}$ 的比例变化。这一结果十分重要，并用于下一节提出的解。

IV. 使用降维估计保持正确分配

本节提出一种保持高关联质量的方法。基于第 III 节的分析，本文给出了用于计算 $\Psi_{(j)}$ 的优化形式。由此提出了基于下降的优化策略，其中一个关键要素和贡献是自适应步长。最后，我们给出一个数值示例，并对该优化策略进行一些评述。

A. 近似分配矩阵

所提解基于上一节的分析。具体而言，我们使用式（17）中的 $\mathsf{E}(\mathbf{r}_{(ij)}^2)$ 来估计 $r_{(ij)}^2$。为计算 $r_{(ij)}^2$，智能体 2 必须同时获得 $(y_{1(i)}, R_{1(i)})$ 和 $(y_{2(j)}, R_{2(j)})$，但 $(y_{1(i)}, R_{1(i)})$ 对智能体 2 而言是未知的。对于 $(y_{1(i)}, R_{1(i)})$，一个已在本地可用的近似是 $(y_{2(i)}, R_{2(i)})$。令

$$\hat{r}_{(ij)}^2 = \hat{y}_{(ij)}^{\mathsf{T}} \Psi_{(j)}^{\mathsf{T}} \left(\Psi_{(j)} \hat{S}_{(ij)} \Psi_{(j)}^{\mathsf{T}}\right)^{-1} \Psi_{(j)} \hat{y}_{(ij)}, \tag{19}$$

$\hat{}$，其中 $\hat{y}_{(ij)} = y_{2(i)} - y_{2(j)}$ 和 $S_{(ij)} = R_{2(i)} + R_{2(j)}$ 使得 $\hat{\mathbf{y}}_{(ij)} \sim \mathcal{N}(\bar{x}_{(ij)}, \hat{S}_{(ij)})$。这与 $\mathbf{r}_{(ij)}^2$ 一致，因为

$$\mathsf{E}\left(\hat{\mathbf{r}}_{(ij)}^2\right) = \begin{cases} m, & \text{if } i = j, \\ m + \bar{x}_{(ij)}^{\mathsf{T}} \Psi_{(j)}^{\mathsf{T}} \left(\Psi_{(j)} \hat{S}_{(ij)} \Psi_{(j)}^{\mathsf{T}}\right)^{-1} \Psi_{(j)} \bar{x}_{(ij)}, & \text{if } i \neq j. \end{cases}$$

$\hat{}$ 该式与 (17) 相同，只是将 $S_{(ij)}$ 替换为 $S_{(ij)}$。随后，我们将近似分配矩阵定义为

$$\hat{\mathcal{A}}_{\text{red}} = \begin{bmatrix} \hat{r}_{(11)}^2 & \cdots & \hat{r}_{(1N)}^2 \\ \vdots & \ddots & \vdots \\ \hat{r}_{(N1)}^2 & \cdots & \hat{r}_{(NN)}^2 \end{bmatrix}. \tag{20}$$

B. 所提解

$\hat{}$ 由于我们只能获得 $\mathcal{A}_{\text{red}}$ 的近似值 $\hat{\mathcal{A}}_{\text{red}}$，因此 $\Psi_{(j)}$ 是根据第 III 节中的充分条件计算得出的，参见 (14)。采用该条件是因为我们希望在选择 $\Psi_{(j)}$ 时留有一定余量，以避免在 $i \neq j$ 时 $r_{(ij)}^2$ 为零或非常小。此外，如果 $\Psi_{(j)}$ 满足这一充分条件，那么计算 $\Psi_{(j)}$ 时无需考虑其他 $\Psi_{(i)}$、$i \neq j$ ——无论如何都能得到正确分配。

现在考虑某个 j 和 $\Psi_{(j)}$。令

$$f_i(z) = \frac{z^{\mathsf{T}} \hat{Y}_{(ij)} z}{z^{\mathsf{T}} \hat{S}_{(ij)} z}, \qquad \hat{Y}_{(ij)} = \hat{y}_{(ij)} \hat{y}_{(ij)}^{\mathsf{T}}, \tag{21}$$

对所有 $i \neq j$ 均有定义。要同时对所有 i 最大化 $f_i(z)$ 通常是不可能的，因为这是一个多目标优化问题。不过，我们可以采用最坏情况方法，使最小的 $f_i(z)$ 最大化。这意味着采用极大极小形式，其中 $\Psi_{(j)}$ 的计算方式为

$$\underset{\Psi_{(j)}}{\text{maximize}} \quad \left(\min_{i \neq j} \quad f_i(\Psi_{(j)}^{\mathsf{T}})\right). \tag{22}$$

(22) 中的问题是一个非凸问题，涉及在有限集的二次型比值上进行优化。一般而言，该问题难以求解，因此提出以下优化策略。

C. 优化策略

对于每个单独的 $f_i(z)$，已知使 $f_i(z)$ 最大化的 z 是对应于 [11] 的最大特征值 λ 的特征向量 u

$$\hat{Y}_{(ij)} u = \lambda \hat{S}_{(ij)} u. \tag{23}$$

由于 $\hat{}$ $Y_{(ij)} \in \mathbb{S}_+^n$ 且 $\operatorname{rank}(\hat{Y}_{(ij)}) = 1$，该特征值问题只有一个严格为正的特征值 λ，其对应的特征向量记为 u_i。由于不同的 i 通常对应不同的 u_i，因此不可能同时最大化所有 $f_i(z)$。不过，对于某个 z，我们知道所有 $f_i(z)$ 的值，因此能够计算

$$i^* = \arg\min_{i \neq j} \quad f_i(z). \tag{24}$$

为了提高 f_{i^*}，建议

$$z \leftarrow z + \alpha u_{i^*}, \tag{25}$$

其中，α 类似于沿着 u_{i^*}移动的步长。若 $|\alpha|$ 过大，则存在这样的风险：对于某个其他的 $i \neq i^*$，f_i 会大幅降低。$|\alpha|$ 过小则意味着收敛缓慢。根据命题 3，沿着 αu_{i^*} 方向在 z 处对 f_i 进行一阶近似，可得

$$f_i(z + \alpha u_{i^*}) \approx f_i(z) + 2\alpha \frac{u_{i^*}^{\mathsf{T}} (\hat{Y}_{(ij)} - f_i(z) \hat{S}_{(ij)}) z}{z^{\mathsf{T}} \hat{S}_{(ij)} z}. \tag{26}$$

我们接下来求解

$$f_i(z)+2\alpha\frac{u_{i^*}^{\mathsf{T}}(\hat{Y}_{(ij)}-f_i(z)\hat{S}_{(ij)})z}{z^{\mathsf{T}}\hat{S}_{(ij)}z}$$
$$=f_{i^*}(z)+2\alpha\frac{u_{i^*}^{\mathsf{T}}(\hat{Y}_{i^*j}-f_i(z)\hat{S}_{i^*j})z}{z^{\mathsf{T}}\hat{S}_{i^*j}z},\qquad(27)$$

对于每个 $i\neq j$、i^*。这将为 α 产生 $N-2$ 个解，其中一些可能为负，另一些为正。由于任务是在不使其他 f_i 降低太多的情况下增加 f_{i^*}，因此选择 α，使得在所有满足条件的解中，$|\alpha|$ 最小

$$\alpha\frac{u_{i^*}^{\mathsf{T}}(\hat{Y}_{i^*j}-f_i(z)\hat{S}_{i^*j})z}{z^{\mathsf{T}}\hat{S}_{i^*j}z}>0.\qquad(28)$$

引入最后这个条件是为了确保为 α 选择正确的符号。

(24)–(28) 中的操作会反复执行，直到满足某个终止准则。优化算法总结于算法2中。

命题 3. 设 u、$z\in\mathbb{R}^n$、Y、$S\in\mathbb{R}^{n\times n}$ 和 $f(z)=(z^{\mathsf{T}}Yz)/(z^{\mathsf{T}}Sz)$，其中 $z\neq 0$ 且 $\mathrm{rank}(S)=n$。则对于任意标量 α，$f(z+\alpha u)$ 的一阶近似为

$$f(z+\alpha u)\approx f(z)+2\alpha\frac{u^{\mathsf{T}}(Y-f(z)S)z}{z^{\mathsf{T}}Sz}.$$

证明。由 [18] 可得

$$\frac{\partial f(z)}{\partial z}=-\frac{2Szz^{\mathsf{T}}Yz}{(z^{\mathsf{T}}Sz)^2}+\frac{2Yz}{z^{\mathsf{T}}Sz}=-\frac{2f(z)Sz}{z^{\mathsf{T}}Sz}+\frac{2Yz}{z^{\mathsf{T}}Sz}$$
$$=2\frac{(Y-f(z)S)}{z^{\mathsf{T}}Sz}z.$$

$f(z+\alpha u)$ 的一阶近似为

$$f(z+\alpha u)\approx f(z)+\alpha u^{\mathsf{T}}\frac{\partial f(z')}{\partial z'}\Big|_{z'=z}$$
$$=f(z)+2\alpha\frac{u^{\mathsf{T}}(Y-f(z)S)z}{z^{\mathsf{T}}Sz}.$$

□

D. 示例

作为所提出优化策略的示例，考虑一个包含 $N=3$ 和 $n=4$ 的场景。假设 $j=3$。由于$N=3$，我们考虑两个损失函数

$$f_1(z)=\frac{z^{\mathsf{T}}\hat{Y}_{(13)}z}{z^{\mathsf{T}}\hat{S}_{(13)}z},\qquad f_2(z)=\frac{z^{\mathsf{T}}\hat{Y}_{(23)}z}{z^{\mathsf{T}}\hat{S}_{(23)}z}.$$

同时最大化 f_1 和 f_2 的多目标问题无法求解，因此我们将采用最大最小方法和算法2。原始算法2采用自适应步长 $\alpha\in[\alpha_{\min},\alpha_{\max}]$。我们将其与采用以下步长的同一算法进行比较：(i) 较小的固定步长 $\alpha=\alpha_{\min}$，以及(ii) 较大的固定步长 $\alpha=\alpha_{\max}$。

这三种情形的优化结果如图 4 所示，它们均使用相同的初始向量 z_0，适用于 $k_{\max}=25$

算法2. 基于关联质量的 $\Psi_{(j)}$ $\alpha_{\min}$ $\alpha_{\max}$

输入：$\mathcal{Y}_2$、j、1 和：对于每个 $i\neq j$：令 $\hat{Y}_{(ij)}=\hat{y}_{(ij)}\hat{y}^{\mathsf{T}}{}_{(ij)}$，$\hat{S}_{(ij)}=R_{2(i)}+R_{2(j)}$ 以及 $f_i(z)=(z^{\mathsf{T}}\hat{Y}_{(ij)}z)/(z^{\mathsf{T}}\hat{S}_{(ij)}z)$。令 $u_{i(ij)}$ 为特征向量对应于以下矩阵的最大特征值 λ_i $\hat{S}_{(ij)}u$。

2：令 $k=0$ 和 $z_0\leftarrow z_0/\|z_0\|$，其中 $z_0=\sum_{i=1,i\neq j}^{N}\frac{1}{\lambda_i}u_i$.
3：令 $k\leftarrow k+1$。计算

$$i^*=\arg\min_{i\neq j}\quad f_i(z_{k-1}).$$

4：对于每个 $i\neq j$：定义

$$\hat{f}_i(z_{k-1}+\alpha u_{i^*})$$
$$=f_i(z_{k-1})+2\alpha\frac{u_{i^*}^{\mathsf{T}}(\hat{Y}_{(ij)}-f_i(z_{k-1})\hat{S}_{(ij)})z_{k-1}}{z_{k-1}^{\mathsf{T}}\hat{S}_{(ij)}z_{k-1}}.$$

5：对于每个 $i\neq j,i^*$：使用 $\hat{f}_i=\hat{f}_{i^*}$ 求解 α。将不同的 α 存储在向量 a中。
6：如果

$$\alpha\frac{u_{i^*}^{\mathsf{T}}(\hat{Y}_{i^*j}-f_i(z_{k-1})\hat{S}_{i^*j})z_{k-1}}{z_{k-1}^{\mathsf{T}}\hat{S}_{i^*j}z_{k-1}}>0,$$

则令 α_k 为 a中的最小正元素。否则，令 α_k 为 a中的最大负元素。如果 $|\alpha_k|<\alpha_{\min}$，则令 $\alpha_k\leftarrow\mathrm{sign}(\alpha_k)\alpha_{\min}$。如果 $|\alpha_k|>\alpha_{\max}$，则令 $\alpha_k\leftarrow\mathrm{sign}(\alpha_k)\alpha_{\max}$。7：令 $z_k\leftarrow z_k/\|z_k\|$，其中 $z_k=z_{k-1}+\alpha_k u_{i^*}$。8：如果满足预定义的停止准则，则以 $\Psi_{(j)}=z_k^{\mathsf{T}}$ 终止。否则，返回步骤 3。

输出：$\Psi_{(j)}$

迭代次数。在图 4a 中，f_1 相对于 f_2绘制。黄色圆点类似于随机采样的 z处的 f_1 和 f_2。图 4b 展示了

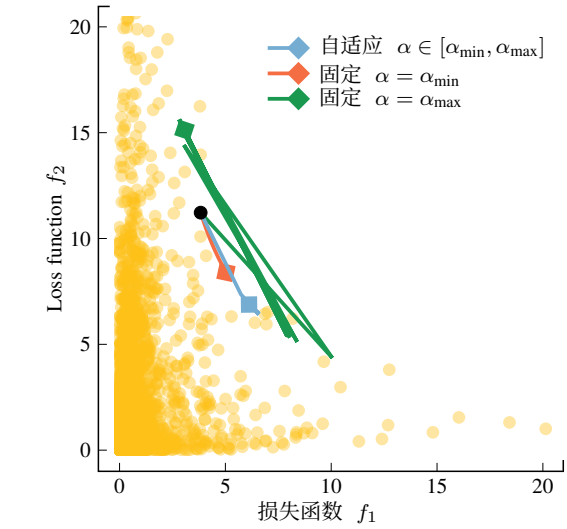
$$f_{\min}=\min\left(f_1,f_2\right),$$

对于每次迭代 $k=1$、2、.....、$k_{\max}$。在这种情况下，自适应步长能得到最佳结果。步长过小会导致收敛缓慢，而步长过大则会因不够准确而产生振荡。无法断定算法2是否已达到全局最大值或驻点。

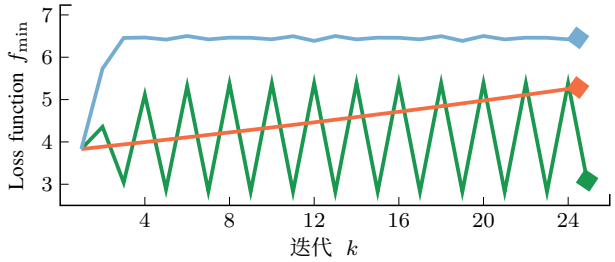
E. 评论

本质上，算法2中提出的优化策略是一种迭代下降优化方法，其下降方向从预先定义的有限方向集中选取。按照这种解释，步骤4–6对应于回溯线搜索，其中选定步长 α。算法2将 $\alpha_{\min}>0$ 作为输入，以避免陷入局部极小值，并将 $\alpha_{\max}>\alpha_{\min}$ 作为输入，以免式(26)给出的线性近似变得过差。本文采用的停止准则为 $k>k_{\max}$，即算法在经过 $k_{\max}$ 次迭代后终止。

可以采用更复杂的优化技术来提升性能，但这些技术不在本文的讨论范围之内。需要强调的是，并不能保证算法2收敛到全局



(a) 损失函数 f_1 与 f_2。



（b）损失函数 $f_{\min}$ 与 k。

图4. 使用 $N = 3$ 和 $n = 4$时所提出优化策略的示例。将算法2与采用固定步长的相同算法进行比较。黑色圆圈标记共同初始值 z_0。方块标记每种情况的最终点。黄色点表示在随机生成的 z处评估的 f_1 和 f_2。步长较小时，收敛速度较慢。步长较大时，结果不够准确，但收敛速度可能更快。自适应步长优于另外两种步长。

最大值（相对于式（22）中的问题而言）。事实上，仿真验证了通常只能达到局部最大值。

V. 数值评估

本节对算法2进行数值评估。比较使用算法2计算 $\Psi_{(j)}$ 时的关联性能与使用算法1计算 $\Psi_{(j)}$ 时的关联性能。

A. 仿真设定

假设存在一个包含 $N = 10$ 个目标的目标跟踪场景。假设其维度 $n = 6$，此处我们将其解释为二维空间中的匀加速模型 [19]。对于每个目标$x_{(i)}$，定义一对协方差 $R_{1(i)}$ 和 $R_{2(i)}$，并在整个仿真过程中保持固定。采用蒙特卡洛（MC）方法，在每次 MC 运行中，分别使用 $R_{1(i)}$ 和 $R_{2(i)}$以及模型 (1) 对状态估计$y_{1(i)}$ 和$y_{2(i)}$ 进行采样。我们还使用缩放因子 c 来缩放两个空间不确定性分量。因此，c 越大，关联

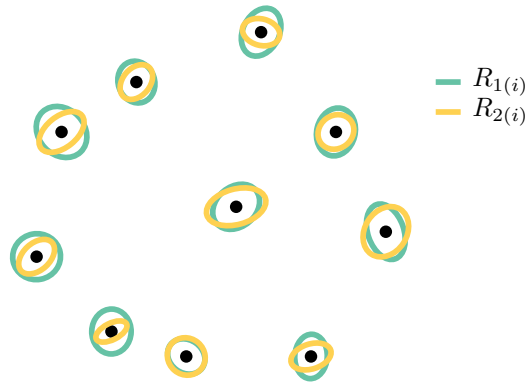


图5. 数值场景。目标以黑点表示。目标周围的椭圆表示对应估计在两个空间维度上的不确定性。

问题就越难解决；c越小，关联问题就越容易解决。图 5 展示了假设的目标跟踪场景，其中 $c = 1$。

为评估关联性能，对于给定的 c，错误分配率 q_e 计算为蒙特卡洛运行次数中错误分配数量的总体均值除以N。我们针对以下情况计算 q_e：

- $(\mathcal{Y}_1, \mathcal{Y}_2)$：完整估计配置，其中代理1从代理2接收 $\mathcal{Y}_2$。
- $(\mathcal{Y}_1, \mathcal{Y}_\Psi) + \Psi_{(j)}$ 采用算法1：一种降维配置，其中代理1从代理2接收 $\mathcal{Y}_\Psi$，并使用算法1计算 $\Psi_{(j)}$。在这种情况下，假设代理2可以访问 $\mathcal{Y}_1$，从而能够计算融合最优 $\Psi_{(j)}$。
- $(\mathcal{Y}_1, \mathcal{Y}_\Psi) + \Psi_{(j)}$ 采用算法2：一种降维配置，其中代理1从代理2接收 $\mathcal{Y}_\Psi$，并使用算法2提出的优化策略计算 $\Psi_{(j)}$。

同时还计算 q_e 的标准差。

注4。由于代理1需要 $\Psi_{(1)}$, ..., $\Psi_{(N)}$，才能将 $\mathcal{Y}_\Psi$ 中的估计值与其局部估计融合，因此代理2在传输 $\mathcal{Y}_\Psi$时也必须包含 $\Psi_{(1)}$, ..., $\Psi_{(N)}$。$\Psi_{(j)}$的编码功能见 [12]，MATLAB®代码可在 https://gitlab.com/robinforsling/dtt/ 获取。

B. 结果

数值评估结果如图6所示，其中 q_e 相对于 c绘制。对于每个 c值，均评估 M =1000次蒙特卡洛运行次数。对于前文所述的每种情况，量 q_e 均在 $\mathcal{Y}_1$ 和 $\mathcal{Y}_2$ 的相同实现中计算。图中的阴影区域表示 1-σ 置信区间。

在完整估计情况下，对于所有 c值，均保持完美关联。采用算法2的方法明显优于为实现最优融合性能而计算 $\Psi_{(j)}$ 的方法。

VI. 结束语

针对降维环境下的多目标跟踪，提出了关联问题。在该问题中，航迹

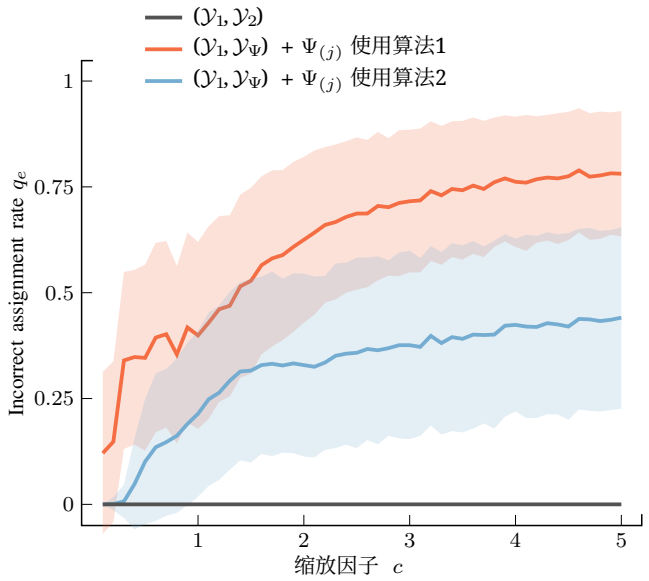


图6。数值评估结果。对于 $c \in [0.1, 5.0]$ 的不同取值，在三种情况下分别将错误分配率 q_e 计算为样本均值。阴影区域表示 q_e 的标准差。

智能体需要传出的估计值，会根据接收降维估计的智能体中的关联质量进行降维。我们对这一隐含问题进行了理论分析，结果表明该问题具有多样性且复杂，并且不存在通用解。有人提出了一种优化策略，用于在保持关联性能的同时计算降维估计。通过数值评估验证了该优化策略，结果显示，所提方法优于一种基于最优融合性能进行降维的方法。

可能的未来扩展包括针对 $m > 1$ 情况对算法2进行推广，以及研究一种更为通用的配置，其中智能体拥有不同的航迹集合。另一种可能性是考虑这样一种场景：接收降维估计的智能体对其他智能体的局部估计掌握部分信息。同时包含融合性能和关联性能的联合问题表述也值得研究。

参考文献

[1] B.-N. Vo、M. Mallick、Y. Bar-Shalom、S. Coraluppi、R. Osborne III、R. P. Mahler 和 B.-T. Vo，《多目标跟踪》，John Wiley & Sons, Ltd，2015年，第1–15页。[2] D. B. Reid，“一种多目标跟踪算法”，《IEEE自动控制汇刊》，第24卷，第6期，第843–854页，1979年。[3] C.-Y. Chong、S. Mori 和 D. B. Reid，“多假设跟踪四十年——关键进展综述”，载于《第21届IEEE国际信息融合会议论文集》，英国剑桥，2018年7月，第452–459页。[4] R. P. Mahler，“通过一阶多目标矩实现多目标贝叶斯滤波”，《IEEE航空航天与电子系统汇刊》，第39卷，第4期，第1152–1178页，2003年。[5] K. Granström、L. Svensson、Y. Xia、J. Williams 和 A. F. García-Fernández，“泊松多伯努利混合跟踪器：借助轨迹随机有限集实现连续性”，载于《第21届IEEE国际信息融合会议论文集》，英国剑桥，2018年7月，第1–5页。[6] N. Kimura 和 S. Latifi，“无线传感器网络数据压缩综述”，载于《IEEE信息技术：编码与计算国际会议论文集》，第2卷，美国内华达州拉斯维加斯，2005年4月，第8–13页。[7] R. Forsling、Z. Sjanic、F. Gustafsson 和 G. Hendeby，“通信约束下的一致性分布式航迹融合”，载于《第22届IEEE国际信息融合会议论文集》，加拿大渥太华，2019年7月。[8] K. Zhang、X. R. Li、P. Zhang 和 H. Li，“最优线性估计融合——第VI部分：传感器数据压缩”，载于《第6届IEEE国际信息融合会议论文集》，澳大利亚昆士兰州凯恩斯，2003年7月，第221–228页。[9] J. Fang 和 H. Li，“齐次及特定非齐次场景下分布式估计的最优/近最优降维”，《IEEE信号处理汇刊》，第58卷，第8期，第4339–4353页，2010年。[10] E. J. Msechu 和 G. B. Giannakis，“通过删失和量化实现无线传感器网络估计的以传感器为中心的数据约简”，《IEEE信号处理汇刊》，第60卷，第1期，第400–414页，2012年。[11] R. Forsling、Z. Sjanic、F. Gustafsson 和 G. Hendeby，“利用特征值优化对降维估计进行最优线性融合”，载于《第25届IEEE国际信息融合会议论文集》，瑞典林雪平，2022年7月。[12] R. Forsling、F. Gustafsson、Z. Sjanic 和 G. Hendeby，“降维线性回归中的去中心化状态估计”，2022年10月，预印本，arXiv。

[13] [ ] 2210 06947 在线。可访问：https://arxiv.org/abs/。——，“仅利用局部信息对降维估计进行去中心化数据融合”，载于《IEEE航空航天会议论文集》，美国蒙大拿州大天空镇，2023年3月。[14] X. Tian 和 Y. Bar-Shalom，“异步航迹间融合算法研究”，载于《第13届IEEE国际信息融合会议论文集》，苏格兰爱丁堡，2010年7月。[15] R. E. Burkard、M. Dell'Amico 和 S. Martello，《指派问题》。美国宾夕法尼亚州费城：工业与应用数学学会，2009年。[16] S. S. Blackman 和 R. Popoli，《现代跟踪系统的设计与分析》。美国马萨诸塞州诺伍德：Artech House，1999年。[17] C. R. Rao，《线性统计推断及其应用》，第2版。美国纽约州纽约市：John Wiley & Sons, Ltd，1973年。[18] K. B. Petersen 和 M. S. Pedersen，“矩阵手册”，2012年11月。[19] X. R. Li 和 V. P. Jilkov，“机动目标跟踪综述。第一部分：动态模型”，《IEEE航空航天与电子系统汇刊》，第39卷，第4期，第1333–1364页，2003年。