

一种可递归和并行实现的快速 JPDA 算法¹

程洪玮 周一宇 孙仲康

(国防科技大学电子工程学院 406 教研室 湖南长沙 410073)

摘 要 联合概率数据关联算法是密集杂波环境下一种良好的多目标数据关联跟踪算法。计算量爆炸效应一直是该算法的一个难题。本文提出一种基于分层构造关联事件的快速算法, 层次间的搜索具有递归性并可以独立进行, 因而可以并行实现。文中还将本文的方法跟有关文献方法作了比较, 并且给出相应的计算机仿真实验和结果。

关键词 概率, 数据关联和跟踪, 快速算法

中图分类号 TN953, TP391

1 引言

密集杂波环境下的多目标关联与跟踪是目前国内外研究的热点之一。联合概率数据关联 (Joint Probabilistic Data Association, JPDA)^[1,2] 算法是其中的一种性能良好的优化算法。其缺点是, 当目标的数目增大时, 目标与观测数据之间的关联假设事件的数量呈指数增长, 其计算量也将呈指数增长甚至造成计算量爆炸。为降低计算量, 有不少文献讨论了次优 JPDA 快速算法, 但它们都是以降低关联跟踪性能为代价的。次优算法的代表有 Fitzgerald^[3] 的经验近似公式法和 Roecker^[4] 的简化 JPDA 法。Zhou^[5] 在 James^[6] 方法的基础上提出了一种深度优先搜索法 (Depth-First Search, DFS), 但其后验概率 β_j^t 的计算无规律可言, 该文只好将后验概率的计算分成直接法和近似法进行。

本文将基于关联假设事件的分层构造方法来讨论一种快速 JPDA 算法。由聚矩阵出发, 本文按层次来分析可行关联事件的排列特点, 推导出一种便于递归处理或并行处理的可行关联事件形成方法, 并基于这种关联事件的形成方法而讨论后验概率的快速计算问题。文中还将本文方法与有关文献方法进行了比较, 并作相应的计算机仿真实验, 给出实验结果和结论。

2 JPDA 方法的简单回顾和问题的提出

设 m_k 为当前扫描 (第 k 次扫描) 获得的回波个数, T_k 为目标个数。在 JPDA 算法中, 第 j 个回波与第 t 个目标之间的关联概率 β_j^t 的计算通常从关联聚矩阵 Ω 的构造开始。由 m_k 个回波和 T_k 个目标构成的关联聚矩阵 Ω 是一个 $m_k \times (T_k + 1)$ 的行列式^[1,2]:

$$\Omega = [\omega_{jt}] = \left(\begin{array}{c|cccc} & \overbrace{0 \quad 1 \quad 2 \quad \cdots \quad T_k}^t & & & \\ \hline 1 & \omega_{11} & \omega_{12} & \cdots & \omega_{1T_k} \\ 1 & \omega_{21} & \omega_{22} & \cdots & \omega_{2T_k} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \omega_{m_1} & \omega_{m_2} & \cdots & \omega_{mT_k} \end{array} \right) \left. \begin{array}{l} 1 \\ 2 \\ \vdots \\ m_k \end{array} \right\} j, \quad (1)$$

¹ 1997-12-08 收到, 1998-11-08 定稿
国防预研基金资助课题

式中 $\omega_{j0} = 1 (j = 1, \dots, m_k)$, 其含义是指所有的回波都可能来自杂波。其他元素 $\omega_{jt} = 1 (t \neq 0)$ 指第 j 回波落入了目标 t 的选通波门, $\omega_{jt} = 0 (t \neq 0)$ 指第 j 回波没有落入第 t 目标的选通波门。可行关联假设事件也将依据聚矩阵 Ω 来生成。目标与回波之间的关联问题实质上是一个组合问题, 从而关联假设事件的数量将随目标和回波数量的增大而呈指数增长。如何降低 JPDA 算法的计算量一直是人们关心的主要问题。文献 [6] 推导了极端情况 (即聚矩阵为全 1 元素) 下的可行关联事件数目, 这也是穷举搜索所需的搜索次数。当 $m_k = 10$, $T_k = 10$ 时, 可行关联事件的数目达 234662231 个, 可知其计算量是难以忍受的。

可用一个关联矩阵 $\hat{\Omega}$ 来表示一个关联事件 $\varepsilon(\hat{\Omega})$, 矩阵 $\hat{\Omega}$ 中的元素用 $\hat{\omega}_{jt}$ 表示。 $\hat{\omega}_{jt} = 1$ 表示回波 j 跟杂波 ($t = 0$) 或者目标 $t (t \neq 0)$ 关联。形成可行关联事件之后, 可用文献 [5] 的相应公式计算可行关联事件的条件概率, 其简化形式为 [5]

$$P(\varepsilon(\hat{\Omega})/Z) = \frac{1}{c} (P_0)^{\min(T_k, m_k) - m_a} \prod_{j: \omega_{jt}=1} P_{jt}. \quad j = 1, 2, \dots, m_k; \quad t = 1, 2, \dots, T_k, \quad (2)$$

式中 Z 为到当前时刻为止的回波观测全集, c 为归一化常数, m_a 为该关联事件中能够跟回波相关联的目标数, 且

$$P_{jt} = \begin{cases} N(\tilde{Z}_j^t; 0, S^t) P_D, & \omega_{jt} = 1, \\ 0, & \text{其他;} \end{cases} \quad (3)$$

$$P_{0t} = \lambda(1 - P_D) = P_0. \quad (4)$$

(3)、(4) 式中, λ 为杂波密度, P_D 为目标探测概率, $N(\tilde{Z}_j^t; 0, S^t)$ 为具有零均值和协方差矩阵为 S^t 的正态分布函数。(2) 式中的 c 可以通过和式 $\sum_{\varepsilon(\hat{\Omega})} P(\varepsilon(\hat{\Omega})/Z)$ 求出。为简便, 下面略去 c 符号。目标 t 跟回波 j 之间的关联后验概率 β_j^t 可由 (2) 式的条件概率来计算:

$$\left. \begin{aligned} \beta_j^t &= \sum_{\varepsilon(\hat{\Omega})} P(\varepsilon(\hat{\Omega})/Z) \omega_{jt}, \\ \beta_0^t &= 1 - \sum_{j=1}^{m_k} \beta_j^t, \end{aligned} \right\} \quad j = 1, 2, \dots, m_k; \quad t = 1, 2, \dots, T_k. \quad (5)$$

不妨举一简例来说明问题。设当前扫描得到三个回波, 此前已跟踪两个目标, 见图 1。按 (1) 式的聚矩阵构造方法, 得此时的聚矩阵如 (6) 式。

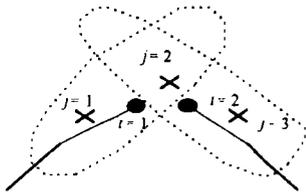


图 1 聚矩阵及关联事件形成举例

$$\Omega = [\omega_{jt}] = \left(\begin{array}{ccc|c} \overbrace{0 \quad 1 \quad 2}^t & & & \\ \hline 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 2 \\ 1 & 0 & 1 & 3 \end{array} \right) j. \quad (6)$$

由此聚矩阵按上述两个条件可穷举搜索得 8 个可行关联事件 $\varepsilon(\hat{\Omega}_n)$, $n = 0, 1, \dots, 7$ 。其相应的

关联矩阵为

$$\hat{\Omega}_0 = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \quad \hat{\Omega}_1 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \quad \hat{\Omega}_2 = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}, \quad \hat{\Omega}_3 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

$$\hat{\Omega}_4 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \quad \hat{\Omega}_5 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad \hat{\Omega}_6 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}, \quad \hat{\Omega}_7 = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

各关联事件的条件概率可按 (2) 式计算, 然后可求出后验概率 β_j^i , 如

$$\beta_2^2 = P(\varepsilon(\hat{\Omega}_2)/Z) + P(\varepsilon(\hat{\Omega}_6)/Z).$$

3 关联事件的构造及快速算法

最显而易见的关联事件是所有目标都不能与当前回波关联这一假设事件, 即上述 $\varepsilon(\hat{\Omega}_0)$. 再之, 如果 $Q_{\max} = \min(m_k, T_k)$, 那么, 在本次扫描的情形下, 每个可行事件中最多可有 Q_{\max} 个回波能够跟目标关联. 由此启发, 本文将可行关联事件进行归类处理, 把一个事件中能跟回波关联的最大目标数作为一个层次, 将所有可行关联事件分成若干层次, 然后按层次分析目标能跟回波关联的情况.

3.1 分层搜索算法 (Layered Searching Algorithm, LSA)

定义有 L 个目标能与回波关联的情形为第 L 层. $L = 0$ 为第 0 层, 表示没有目标能与回波关联. 显然最大层次为 $Q_{\max} = \min(m_k, T_k)$. $L = 0$ 层只有一个事件, 即 $\varepsilon(\hat{\Omega}_0)$. 注意到, 在关联矩阵 $\hat{\Omega}_0$ 中, 除 $t = 0$ 列的元素非 0 外, 其他元素全为 0. 对 $L = 1$ 层的事件, 图 1 例中有 $\varepsilon(\hat{\Omega}_1)$ 、 $\varepsilon(\hat{\Omega}_4)$ 、 $\varepsilon(\hat{\Omega}_6)$ 、 $\varepsilon(\hat{\Omega}_7)$. 又注意到, 在关联矩阵 $\hat{\Omega}_n (n = 1, 4, 6, 7)$ 中, $t = 0$ 列之外的其他元素中只能有一个元素为 1. $L = 2, 3, \dots, Q_{\max}$ 的情况依此类推. 现在的问题是, 如何分层生成关联事件并使得生成次序有利于降低后验概率的计算量.

Zhou 和 James 的方法是依据聚矩阵的行向量中非 0 值来进行搜索并以此得出快速算法. 本文考察聚矩阵中列向量的情况. 对于 $0 < L \leq Q_{\max}$ 的层次, 每一个关联矩阵中除 $t = 0$ 列外, 其他各列中, 只能在 L 个列 (记为 t_1, t_2, \dots, t_L 列) 中取非 0 值, 取值遵照可行事件形成的两个约束条件, 次序为自上而下. 对照 $L = 1$ 层的情况, 可在除 $t = 0$ 列外的其他列中的 1 个列中取非 0 值, 亦即, 若仅在第 $t_1 = 1$ 列取非 0 值, 可得关联事件为 $\varepsilon(\hat{\Omega}_1)$ 、 $\varepsilon(\hat{\Omega}_4)$; 若仅在第 $t_1 = 2$ 列上取非 0 值, 可得关联事件为 $\varepsilon(\hat{\Omega}_6)$ 、 $\varepsilon(\hat{\Omega}_7)$. 至此, 已经没有非 0 值元素可取. 这样得出的四个关联事件正好就是 $L = 1$ 层的全部可行事件. 注意到, 在 $L = 1$ 层的搜索处理时, 列序号 (即目标序号) 的变化范围是从 $t_1 = 1$ 到 $t_1 = 2$, 即从 1 到 Q_{\max} .

再来考察 $L = 2$ 层的搜索. 类似地只能在 $t = 0$ 列之外的其他 2 个列 (t_1, t_2 列) 中取非 0 值. 由 (6) 式, 在 $t_1 = 1$ 列中自上而下先取 $\omega_{11} = 1$ 值, 再在 $t_2 = 2$ 列中自上而下分别取 $\omega_{22} = 1$ 和 $\omega_{32} = 1$, 组合成 $\omega_{11}\omega_{22}$ 和 $\omega_{11}\omega_{32}$ 从而产生两个 $L = 2$ 层的关联事件, 即 $\varepsilon(\hat{\Omega}_2)$ 、 $\varepsilon(\hat{\Omega}_3)$. 该处理过程如图 2. 再在 $t_1 = 1$ 列中自上而下取第二个非 0 值, 即 $\omega_{21} = 1$, 类似地, 在 $t_2 = 2$ 列中自上而下搜索, 只能取 $\omega_{32} = 1$, 组合成 $\omega_{21}\omega_{32}$ 而得一个 $L = 2$ 层的关联事件 $\varepsilon(\hat{\Omega}_5)$. 至此, 全部层次的关联事件搜索结束. 这里将本法的搜索与 Zhou 和 James 方法比较, 按照 Zhou 和 James 的方法对上例的聚矩阵进行搜索, 搜索得出的关联事件的次序为:

$\varepsilon(\hat{\Omega}_0) \rightarrow \varepsilon(\hat{\Omega}_1) \rightarrow \varepsilon(\hat{\Omega}_2) \rightarrow \varepsilon(\hat{\Omega}_3) \rightarrow \varepsilon(\hat{\Omega}_4) \rightarrow \varepsilon(\hat{\Omega}_5) \rightarrow \varepsilon(\hat{\Omega}_6) \rightarrow \varepsilon(\hat{\Omega}_7)$. 用本文方法, 次序为:
 $\varepsilon(\hat{\Omega}_0) \rightarrow \varepsilon(\hat{\Omega}_1) \rightarrow \varepsilon(\hat{\Omega}_4) \rightarrow \varepsilon(\hat{\Omega}_6) \rightarrow \varepsilon(\hat{\Omega}_7) \rightarrow \varepsilon(\hat{\Omega}_2) \rightarrow \varepsilon(\hat{\Omega}_3) \rightarrow \varepsilon(\hat{\Omega}_5)$.

$$\Omega = \left(\begin{array}{ccc|c} \overbrace{0 \ 1 \ 2}^t & & & \\ \hline 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 2 \\ 1 & 0 & 1 & 3 \end{array} \right) \xrightarrow{j} \hat{\Omega}_2 = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}; \quad \Omega = \left(\begin{array}{ccc|c} \overbrace{0 \ 1 \ 2}^t & & & \\ \hline 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 2 \\ 1 & 0 & 1 & 3 \end{array} \right) \xrightarrow{j} \hat{\Omega}_3 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

图 2 $L=2$ 层相邻的两次搜索

在 $L=2$ 层的搜索中注意到, 2 个列的列号次序取值没有变化, 即 t_1 取值为从 1 到 1, t_2 取值为从 2 到 2. 回想 $L=1$ 层搜索时的列号取值次序为 t_1 取值从 1 到 Q_{\max} . 还需指出, 在 $L=2$ 层的搜索中, 若 t_1 列和 t_2 列的序号取值保持不变, 则相邻两次搜索的中间结果 $\omega_{j_1 t_1} \omega_{j_2 t_2}$ 仅有下标 j_2 发生变化.

将上述搜索方法推广应用到 $Q_{\max} \geq 3$ 的情况中, 经本文考察可归纳出下列结论:

(1) 设第 L 层的搜索中间结果为 $\omega_{j_1 t_1} \omega_{j_2 t_2} \cdots \omega_{j_L t_L}$, 则列号 $t_i (i=1, 2, \cdots, L)$ 的取值范围为: $i \leq t_i \leq Q_{\max} - L + i$. 如上例中第一层 t_1 的取值为从 1 到 $Q_{\max} - 1 + 1 = Q_{\max} = 2$; 第二层中间结果 $\omega_{j_1 t_1} \omega_{j_2 t_2}$ 中, t_1 取值为从 1 到 $Q_{\max} - 2 + 1 = 1$, t_2 取值为从 2 到 $Q_{\max} - 2 + 2 = Q_{\max} = 2$.

(2) 在进行第 L 层的搜索时, 若列编号 t_i 不变且跟回波相应的下标 $j_i (i=1, 2, \cdots, L-1)$ 也不变时, 则相邻两次搜索的中间结果中仅有最后的元素 $\omega_{j_L t_L}$ 发生变化, 即从 $\omega_{j_1 t_1} \omega_{j_2 t_2} \cdots \omega_{j_{L-1} t_{L-1}} \omega_{j_L t_L}$ 变为 $\omega_{j_1 t_1} \omega_{j_2 t_2} \cdots \omega_{j_{L-1} t_{L-1}} \omega_{j'_L t_L}$, 注意下划线部分的不同. 如上例第二层搜索时, $t_1 = 1$, $t_2 = 2$ 不变下 $\omega_{11} \omega_{22}$ 变为 $\omega_{11} \omega_{32}$.

(3) 上述两结论适用于任何 $L > 1$ 层次的搜索. 必须指出, 由于各层次的搜索是独立进行的并且具有相同的逻辑结构, 因而各层次的搜索可递归或并行实现.

事实上, 高明的程序员可将上述搜索过程用更加紧凑且可递归调用的形式来实现. 由于这一方法具有潜在的并行机制, 也可以通过软件、硬件来分层并行实现. 必须指出, 在上述搜索程序中, 需要存储的中间结果仅为 $\omega_{j_1 t_1} \omega_{j_2 t_2} \cdots \omega_{j_L t_L}$. 另外, 聚矩阵 Ω 也需占用一些内存. 这一特点与 Zhou 和 James 的方法不同, 他们的方法中需存储由聚矩阵生成的各行向量及其调用情况. 对一个系统资源有限的传感器如雷达、红外跟踪搜索仪而言, 节省内存、软件紧凑、程序的可递归性以及可并行实现, 都正是工程设计人员所想要达到的目标.

3.2 后验概率 β_j^t 的快速计算

后验概率 β_j^t 的快速计算将基于上述搜索过程的结论 2 进行. 在第 L 层, (2) 式可重写为

$$P(\varepsilon(\hat{\Omega}/Z)) = \frac{1}{c} (P_0)^{Q_{\max}-L} \prod_{j:w_{jt}=1} P_{jt}. \quad j = j_1, j_2, \cdots, j_L; \quad t = t_1, t_2, \cdots, t_L. \quad (7)$$

结合本文的关联事件生成方式, 则对于同一 L 层, 因子 $(P_0)^{Q_{\max}-L}$ 仅需要计算一次. L 阶连乘因子 $\prod_{j:w_{jt}=1} P_{jt}$ 的计算在 JPDA 方法中很费时, 但在本法中, 该因子将依据搜索中间结果 $\omega_{j_1 t_1} \omega_{j_2 t_2} \cdots \omega_{j_L t_L}$ 进行. 在形成聚矩阵时, 与聚矩阵中非 0 元素一一对应的正态概率分布函数 P_{jt} 也可预先计算好. 由结论 (2), 如果列编号 $t_i (i=1, 2, \cdots, L)$ 保持不变, 则将与 $\omega_{j_1 t_1} \omega_{j_2 t_2} \cdots \omega_{j_{L-1} t_{L-1}}$ 相应的 $L-1$ 阶连乘因子 $\prod_{j:w_{jt}=1} P_{jt} (j = j_1, \cdots, j_{L-1}; \quad t =$

t_1, \dots, t_{L-1}) 的计算结果暂存起来, 待本层的下一个关联事件产生后, 只需将 $\omega_{j_L t_L}$ 相应的 $P_{j_L t_L}$ 值与暂存的 $L-1$ 阶连乘因子 $\prod_{j: w_{j,t}=1} P_{jt} (j = j_1, j_2, \dots, j_{L-1}; t = t_1, t_2, \dots, t_{L-1})$ 相乘即可; 若列编号 t_i 有变化, 则 L 阶连乘因子 $\prod_{j: w_{j,t}=1} P_{jt} (j = j_1, j_2, \dots, j_L; t = t_1, t_2, \dots, t_L)$ 中所有 L 个因子都需要更新且需计算一次连乘, 共需作 $L-1$ 次乘法, 用所得的新的 $L-1$ 阶连乘因子 $\prod_{j: w_{j,t}=1} P_{jt} (j = j_1, \dots, j_{L-1}; t = t_1, \dots, t_{L-1})$ 替代暂存在内存中的原 $L-1$ 阶连乘因子。由于相邻两次搜索所得到的两个事件之间的计算结果有继承性, 因而大大降低了计算量。所有的 $P(\varepsilon(\hat{\Omega}/Z)$ 计算完后, 可依据 (5) 式计算所有的后验概率 β_j^t 。

3.3 计算量的定量分析

这里只给出求 β_j^t 所需乘法次数, 未计节约的加法次数。由上述三点结论, 第 L 层的乘法次数为

$$\begin{aligned} N_L &= C_{Q_{\max}}^L [m_k \cdots (m_k - L + 2) \cdot (L - 1) + m_k \cdots (m_k - L + 2) \cdot (m_k - L)] \\ &= C_{Q_{\max}}^L (m_k - 1) \frac{m_k!}{(m_k - L + 1)!}. \end{aligned} \quad (8)$$

计算 β_j^t 所需的乘法次数为各层乘法数目之和, 即

$$N = \sum_{L=2}^{Q_{\max}} N_L = \sum_{L=2}^{Q_{\max}} C_{Q_{\max}}^L (m_k - 1) \frac{m_k!}{(m_k - L + 1)!}. \quad (9)$$

将本文的 LSA 方法与 Zhou 的 DFS 方法就所需的乘法次数进行比较, 见表 1。

表 1 算法所需乘法次数的比较, DFS 法 / LSA 法

T_k	m_k	DFS/LSA
3	6	318/240
4	8	5072/4258
5	10	96890/86764
6	12	2218992/2078347

每一目标的预测波门内有 2 ~ 3 个回波的假设是相对合乎实际情况的, 根据这一假设, 观测回波的总数一般都落入下面的闭区间内, 即

$$T_k \leq m_k \leq 2T_k + 1. \quad (10)$$

由这一假设出发, Zhou 方法的计算量中的乘法次数上限为

$$M(T_k, m_k) < 2 * 4^{T_k} - 2 - 3T_k - 3^{T_k}. \quad (11)$$

本文中 LSA 方法的第 L 层乘法次数为

$$N_L = C_{Q_{\max}}^L \overbrace{3 \cdot 3 \cdots 3}^{L-1} \cdot (L - 1) + C_{Q_{\max}}^L \overbrace{3 \cdot 3 \cdots 3}^{L-1} \cdot (3 - 1) = C_{Q_{\max}}^L 3^{L-1} (L + 1), \quad (12)$$

则本文 LSA 方法的计算量中的乘法次数上限为

$$N = \sum_{L=2}^{Q_{\max}} N_L = \sum_{L=2}^{Q_{\max}} C_{Q_{\max}}^L 3^{L-1} (L + 1). \quad (13)$$

将本文 LSA 方法与 Zhou 的 DFS 方法就所需的乘法次数上限进行比较, 见表 2。

表 2 算法所需乘法次数上限的比较

T_k	DSF	LSA
3	90	63
4	417	333
5	1788	1611
6	7443	7497
7	30558	34119

注意到当 $T_k \geq 7$ 时, 本文 LSA 方法的乘法数上限比 DFS 方法的上限略大。但是, 推导 (12) 式时的条件放得比 DFS 还宽松。这里也将本文 LSA 方法的乘法数上限与 James 方法的上限作比较, 见表 3。

表 3 James 的快速 JPDA 方法与本文 LSA 方法所用乘法数目的比较, JPDA/LSA

回波数目 m_k	目标数目 T_k				
	3	4	5	6	7
T_k	51/30	260/131	1045/542	3654/2496	11655/8637
$T_k + 1$	78/37	432/178	1830/795	6624/3450	21603/11601
$T_k + 2$	111/43	672/225	3030/911	11466/3615	38661/14841
$T_k + 3$	150/51	992/245	4780/1027	19020/4365	66626/16695
$T_k + 4$	195/55	1404/265	7235/1143	30360/4653	110922/18549
$T_k + 5$	—	1920/285	10570/1235	46824/5023	178836/20403
$T_k + 6$	—	—	14980/1327	70044/5481	279058/22095
$T_k + 7$	—	—	—	101976/5897	426636/23787
$T_k + 8$	—	—	—	—	634473/25479

表 4 10 个目标的初始位置和速度

目标 t	初始位置 (km)		速度 (km/s)	
	x	y	\dot{x}	\dot{y}
1	3.0	3.0	0.72	0.12
2	3.0	7.8	0.72	-0.12
3	10.0	2.0	0.45	0.60
4	6.0	10.0	0.62	0.35
5	6.0	12.0	0.62	0.35
6	5.0	24.0	0.50	0.22
7	5.0	30.0	0.50	-0.50
8	4.0	34.0	0.70	-0.31
9	5.0	20.5	0.62	0.00
10	4.0	18.0	0.70	0.00

由上面的分析可知, 本文的 LSA 方法具有快速、可递归、省内存、可并行分层实现的特点。对本文方法稍加修改就可以推广到杂波环境下的机动多目标跟踪^[7]。本文限于篇幅, JPDA 算法的其他部分的内容如归一化常数的计算、JPDA 滤波方程组等, 不再一一列举, 读者可从参考文献查到。

4 算法的计算机仿真实验

实验采用线性测量模型, X 、 Y 两方向上的测量误差为 0.15km, 系统扰动系数取为 10^{-4} 。设采样周期为 1s, 监控区域为 $40 \times 40\text{km}^2$ 的范围, 10 个目标的初始位置和速度如表 4 所示。

设传感器对目标的探测概率为 $P_D = 0.9$, 杂波点的数目服从 Poisson 分布, 按照参数为 $\lambda = 2.0$ 的 Poisson 分布函数来产生杂波, 关联波门门限 G^2 选为 17。设置两个实验题目, 一是考察一次样本计算中所用的平均时间, 二是考察 LSA 算法的平均成功率 (即每一百次实验中

不丢失目标的正确跟踪次数)。实验在 PC586-133 上进行, 实验结果见表 5 及表 6, 图 3 ~ 图 6 分别为对 5 个、10 个目标正确跟踪 (80 次扫描) 的例子。

表 5 LSA、DFS 及穷举法进行数据关联所用的平均 CPU 时间

目标数目 T_k	回波数目 m_k	采样次数	LSA 法平均 CPU 时间 (秒)	DFS 法平均 CPU 时间 (秒)	穷举搜索法平均 CPU 时间 (秒)
5	11	16	0.055	0.037	0.775
6	12	21	0.105	0.131	2.189
7	15	18	0.314	0.341	5.585
8	18	15	0.771	0.913	21.821
9	18	15	2.001	2.218	48.333
10	21	8	16.513	20.072	465.383

表 6 跟踪实验的成功率 (扫描次数为 60)

目标数目	实验重复次数	成功次数	成功率 (%)
5	100	100	100
6	100	100	100
7	100	100	100
8	100	99	99
9	80	76	95
10	80	71	88.75

5 结 论

本文讨论了快速递归计算 JPDA 的 LSA 算法, 包括关联假设事件的分层搜索生成以及后验概率的快速计算, 分析和比较了计算量。为了证实该方法的有效性, 做了 Monte Carlo 仿真实验。LSA 方法具有快速、省内存、可递归或并行实现的特点, 这些特点为该方法的实际应用提供了基础, 它特别方便于在工程实践中用硬件来并行实现。以上特点使得本文方法具有很大的实际应用意义。

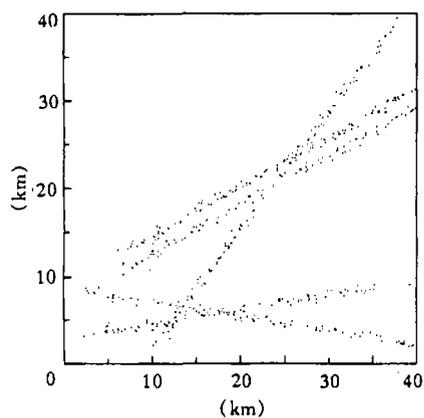


图 3 在预测波门内得到的五个目标的观测报告 (含目标数据和杂波)

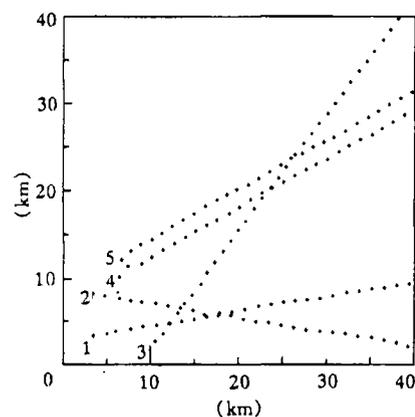


图 4 图 3 的五个目标的跟踪结果

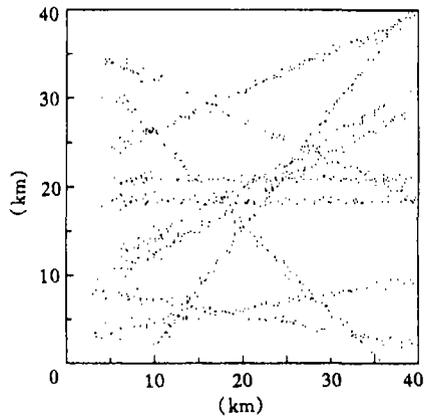


图 5 在预测波门内得到的十个目标的观测报告(含目标数据和杂波)

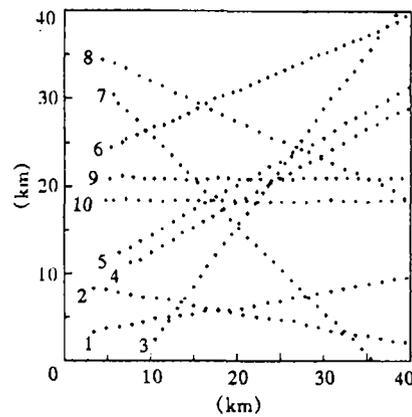


图 6 图 5 的十个目标的跟踪结果

参 考 文 献

- [1] Bar-shalom Y, Blackman T E. Tracking and Data Association. New York: Academic Press, 1988.
- [2] Blackman S S. Multiple Target Tracking with Radar Applications. Delham, MA: Artech House, 1986.
- [3] Fitzgerald R J. Development of practical PDA logic for tracking by microprocessor. Proceedings of American Controls Conference. Seattle, WA: June 1986, 889-898.
- [4] Roecker J A. A class of near optimal JPDA algorithms. IEEE Trans. on AES, 1994, AES-30(2): 504-510.
- [5] Zhou B, Bose N K. Multitarget tracking in clutter: Fast algorithm for data association. IEEE Trans. on AES, 1993, AES-29(2): 352-363.
- [6] James L F, David P C. Fast JPDA multitarget tracking algorithm. Applied Optics, 1989, 28(2): 371-376.
- [7] Houles A, Bar-shalom Y. Multisensor tracking of a maneuvering target in clutter. IEEE Trans. on AES, 1989, AES-25(2): 176-189.

A RECURSIVE AND PARALLEL FAST JPDA ALGORITHM

Cheng Hongwei Zhou Yiyu Sun Zhongkang

(Institute of Electron. Eng., National University of Defence Technology, Changsha 410073)

Abstract Joint Probabilistic Data Association (JPDA) is a very fine optimal multitarget tracking and association algorithm in clutter. However, the calculation explosion effect in computation of association probabilities has been a difficulty. This paper will propose a method based on layered searching construction of association hypothesis events. According to the method, the searching schedule of the association events between two layers can be recursive and with independence, so it can also be implemented in parallel structure. Comparative analysis of the method with relative methods in other references and corresponding computer simulation tests and results are also given in the paper.

Key words Probability, Data association and tracking, Fast algorithm

程洪玮: 男, 1964年生, 博士, 主要研究领域为雷达系统、多传感器数据融合、单/多站有源无源定位跟踪等技术, 近年来在国内外发表论文 20 余篇。