针对不确定射频识别数据流的改进概率推导方法

聂艳明，李战怀，陈群
（西北工业大学计算机学院，710072，西安）

摘要：针对射频识别（RFID）数据存在漏读和交叉读而导致所提供的位置信息不准确，以及RFID数据与上层应用需求之间存在的信息鸿沟，提出了一种可以处理RFID交叉读问题的改进的RFID数据推导方法.该方法利用动态图模型并辅以历史RFID识读，从不确定RFID数据流上有效捕获对象的当前状态，采用基于熵的方法推导对象的最可能位置与包含，并且利用仿真物流仓库的RFID模拟数据进行算法评价.实验结果显示，该方法在获得准确推导结果的同时，能确保其高效性.对于常见RFID部署，位置推导和包含推导的错误率都可以控制在10%以内，针对超过17万个节点的推导所用时间小于1 s，采用修剪措施后内存使用低于700 MB.

关键词：不确定射频识别数据流；交叉读；动态图模型；基于熵；概率推导
中图分类号：TP311 **文献标志码**：A **文章编号**：0253-987X(2011)12-0045-08

Enhanced Probabilistic Interpretation over Uncertain Radio Frequency Identification Data Streams

NIE Yanming, LI Zhanhuai, CHEN Qun
(School of Computer Science, Northwestern Polytechnical University, Xi′an 710072, China)

Abstract: To address the uncertainty of radio frequency identification (RFID) data due to the existing missed readings and cross readings, and to mitigate the information gap between RFID data and the requirements of upstream applications, an enhanced data interpretation method is proposed, which is able to handle with cross RFID readings over uncertain RFID streams. A dynamic graph model is chosen, supplemented with the history of RFID readings, to efficiently capture the possible locations and containment for tagged objects. Then an entropy-based method is adopted to estimate the most likely location and containment for each object. The method with the synthetic RFID data is also evaluated. The experimental results show the accuracy and efficiency. The error rates for both inferences get within 10% for common RFID deployment. The inference time is no more than 1 s, as the number of objects over 1.7×10^5 and the used memory less than 700 MB by employing likelihood threshold-based trimming.

Keywords: uncertain radio frequency identification data streams; cross readings; graph model; entropy-based; probabilistic interpretation

由于射频识别（RFID）技术本身的局限性以及环境干扰等影响因素，RFID识读存在漏读和交叉读，所提供的位置信息不准确，同时RFID识读语义非常简单，并不直接捕获对象间的关联关系（如包含等），因而针对RFID数据的清洗和推导就显得尤为关键.我们将仅仅基于识读本身来处理RFID数据的不准确性称为针对RFID数据的物理清洗（即数据清洗），将兼顾识读本身以及识读中存在的关联关

收稿日期：2011-05-24. 作者简介：聂艳明（1972—），男，博士生；李战怀（通信作者），男，教授，博士生导师. 基金项目：国家自然科学基金资助项目（60720106001，61033007）.
网络出版时间：2011-10-08 网络出版地址：http://www.cnki.net/kcms/detail/61.1069.T.20111008.008.html

系来消减 RFID 数据的不准确性称为针对 RFID 数据的逻辑清洗(即数据推导),这也是本文的讨论重点.

文献[1]提出了延迟 RFID 数据清洗的方法,并利用重写机制来限制清洗数据量以提高处理效率. 文献[2]提出的 RFID 数据过滤方法在保持原始事件的顺序和内存需求方面尤为有效. ESP(Extensible receptor Stream Processing)[3]通过引入时态与空间粒度的概念对 RFID 数据流进行在线清洗,但在粒度的选取上存在困难. SMURF(Statistical sMoothing for Unreliable RFid datal)[4]则通过将 RFID 数据流看成物理世界中 RFID 标签的统计样本,通过动态调整窗口大小(即时态粒度)来对 RFID 数据流进行平滑处理,但 SMURF 无法捕获标签在多个阅读器间的迁移.

文献[5]通过扩展 SMURF,提出了 3 种基于动态概率路径事件模型的数据填补算法,并通过挖掘已知区域事件的顺序相关性来对后续事件进行填补. 文献[6-7]综合利用对象的时态信息及其时空关联关系,对 RFID 数据进行清洗. 文献[8]提出一种概率模型,将原始移动 RFID 数据流转换为包含位置坐标信息的事件流. 文献[9]基于当前时刻的 RFID 识读推导对象的位置,如同文献[5-8]中的方法一样也无法应对交叉读问题. 文献[9]同时假定特殊阅读器事先已知并可直接确认对象的包含,存在局限性.

本文提出一种改进的针对不确定 RFID 数据的推导方法. 该方法基于时变图模型,并辅以对象的历史位置信息和对象间的历史同位置信息,捕获对象的当前状态,随后采用基于概率熵的方法来分别估计对象最可能的位置与包含. 针对经过上述位置推导后仍为不确定的节点,本文还提出一种有效形成子图的方法对不确定对象作进一步的推导.

1 问题陈述

1.1 相关概念

(1)物理世界及其状态与观测. 一个物理世界包括一个贴标对象集 O、一个预先设定的位置集 L 和一个全序的离散时域 T. 位置集 L 中的位置可以是逻辑区域(如仓库入口)或是物理坐标,本文暂只考虑逻辑位置. 在某个时刻 t,贴标对象所在位置及其之间的包含等关联关系,称为物理世界在 t 时刻的状态. 某个时刻的 RFID 识读,则称为对物理世界的一个观测.

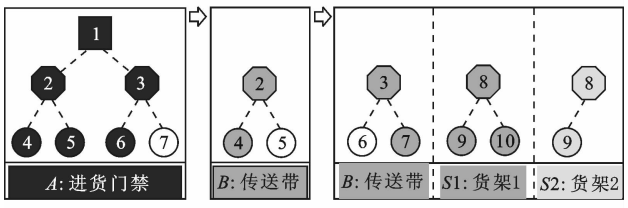
(2)RFID 数据中的漏读与交叉读. 位于阅读器识读范围中的贴标对象没有成功地被该阅读器检测到,此即 RFID 数据中的漏读,通常使用识读率来描述漏读,指 RFID 标签能够被位置最近的阅读器检测到的概率. 由于部署的 RFID 阅读器的识读范围存在一定程度的区域重叠,贴标对象可能会被邻近的阅读器检测到,此即 RFID 数据中的交叉读,使用交叉率来描述,指 RFID 标签能被其邻近阅读器识读到的概率.

(3)RFID 识读中的时态信息与空间信息. 动态产生的 RFID 数据中包含关于贴标对象随时间而改变的状态数据,如对象的位置等,此即 RFID 识读中的时态信息. 贴标对象之间也存在诸如同位置、包含等多种关联关系,此即 RFID 识读中的空间信息.

RFID 数据推导问题就是基于已有 RFID 识读中并不精确的时态信息和空间信息,对物理世界的当前状态进行尽可能准确的估计,即报告对象的最可能位置和最可能包含. 对 RFID 数据流的实时推导主要是基于最近 k 个时刻内对象的历史位置信息,并辅以对象间的同位置信息,推导对象在当前时刻最可能的位置和包含.

1.2 应用场景示例

一个小型物流仓库示例如图 1 所示. RFID 标签编码标准[10]规定了对象的 3 个包装层级(即单品、包装箱、托盘,分别对应到图 1 中的方形、八边形、圆形),将其编码于对象标签的 ID 中. 为了表示方便,每个对象被着色为 t 时刻观测位置的颜色(可以存在多个观测位置),而白色的节点表示 t 时刻该对象未被任何位置观测到. 虚线边表示对象间的包含关系是不确定的. 如果对象存在已确认的父包含,则相应的父连接边改由实线表示.



(a)$t=1$ (b)$t=2$ (c)$t=3$

图 1 小型物流仓库示例

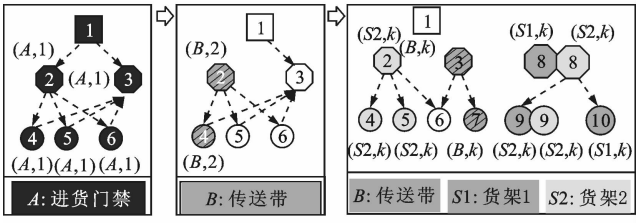
在 $t=1$ 时,进货门禁处的阅读器观测到对象 1~6,由实心节点表示. 对象 7 也处于进货门禁,但未被观测到(即漏读),由空心节点表示. 对象 4、5 和 6 的包装容器可以是箱子 2,也可以是箱子 3,所以存

在不确定性. 在 $t=2$ 时,箱子 2 在传送带上被单独扫描. 在 $t=3$ 时,箱子 3 在传送带上被单独扫描,同时箱子 8 以及单品 9 被货架 1 和 2 同时观测到(即交叉读). 由于 RFID 识读本身的不可靠性,以及诸如包含关系等语义的隐含性,物流仓库中对象的状态是不确定的,这就需要进行数据推导.

2　数据捕获模型及其动态更新

2.1　数据捕获模型

采用图模型 $G=(V, E)$ 表示物理世界(如物流仓库)当前可能所处的状态[9]. 图捕获模型及其演化的示例如图 2 所示,其中节点集合 V 表示贴有 RFID 标签的对象,根据对象的包装层级对节点进行分层,每层对应一个包装层级. 边集 E 表示对象间可能的包含关系,一条有向边 $o_i \rightarrow o_j$ 表示对象 i 包含对象 j. 模型允许节点具有多条子边(如一个包装箱可包含多个单品)和多条父边(如一个单品对应多个可能的包装箱),最终通过概率方法选择最可能的包含.



(a)$t=1$　　(b)$t=2$　　(c)$t=k(k>2)$

图 2　图模型及其动态更新

图模型保存有历史统计信息. 首先,与文献[9]中仅仅利用当前时间的观测位置不同,本文为每个节点保存了所对应对象在最近 K 个时间周期的观测位置信息 L_K. 对象在某个时间周期上被所有位置的观测情况由一个位向量表示,每位对应到一个观测位置. 如对象被某个位置观测到,就将该位置对应为位置 1,否则置 0. 显然,该图模型可以处理 RFID 交叉读问题. 另外,每个节点还记录其对应的已确认父包含 P_C. 图 2c 中节点 8 和 9 分别对应有两个图形,表示同时被两个位置观测到,但在图模型中只是更新节点所对应的位向量. 其次,每条边维护一个位向量 $\boldsymbol{C}_S$ 来记录邻接节点对应的两个对象在最近 S 个时间周期是否被相同位置观测到. 当一条边的两个邻接对象在同一个位置同时被观测到时,位向量中对应的位就被设定为 1,否则为 0.

2.2　图模型的实时更新

根据当前时间周期的识读集 $R^{(t)}$,对前一个时间周期的图进行更新处理,得到一个新图. 由于 RFID 技术本身的物理特性以及部署环境的限制,RFID 阅读器一次能够识读的 RFID 标签的数量有限,因而一次所产生的 RFID 识读数量也不大. 而且,在图模型更新以及数据推导过程中,使用到的仅为当前时间周期的 RFID 识读. 对于应用场景规模非常庞大的情形,可以考虑采用分布式并行的处理方法. 图模型更新算法的伪代码如图 3 所示,其步骤说明如下.

(1)新建节点. 如果对象是第一次被观测到,则新创建一个图节点,如图 3 中的 1～5 行.

(2)添加边. 如果相邻层两个未邻接的节点具有相同位置的观测,则在该两个节点之间添加一条边,如图 2c 中节点 3-7、8-9、8-10 之间的边,见图 3 中 15 行和 26 行.

输入:前一个时间周期的图G, 当前时间周期上的识读集R_t
输出:更新的图G

```
1.    For each 识读R_t中的每个对象o
2.        v ← 对象o在图G中对应的节点
3.        如果v不存在, 创建新节点v并加入到图G中
4.        更新节点v的L_K
5.    End For
6.    For each 识读R_t中存在对象所对应的包装层级L
7.      For each 包装层级L中的节点v
8.        V_{L+1} ← 识读R_t中包装层级为L+1的节点集
9.        V_p ← 节点v的父节点集
10.       For each (V_{L+1} ∪ V_p) 中的节点v_{+1}
11.         e_{+1} ← 节点v和v_{+1}之间的连接边
12.         If v和v_{+1} 两者的观测位置互不交叉
13.           如果存在e_{+1}, 则删除e_{+1}
14.         Else
15.           如果e_{+1}不存在, 则创建新连接边e_{+1}
16.           将e_{+1}的C_S中的相应位设置为1
17.         End If
18.       End For
19.       V_{L-1} ← 识读R_t中包装层级为L-1的节点集
20.     V_c ← 节点v的子节点集
21.       For each (V_{L-1} ∪ V_c) 中的节点v_{-1}
22.         e_{-1} ← 节点v和v_{-1}之间的连接边
23.         If v和v_{-1} 两者的观测位置互不交叉
24.           如果存在连接边e_{-1}, 则删除e_{-1}
25.         Else
26.           如果e_{-1}不存在, 则创建新连接边e_{-1}
27.           将e_{-1}的C_S中的相应位设置为1
28.         End If
29.       End For
30.     End For
31.   End For
```

图 3　RFID 流数据驱动的图模型更新算法

(3)删除边. 如两个邻接节点具有互不交叉的位置观测,则该边可删除. 如图 2c 中节点 3-4、3-5 之间

的边,见图3中13行和24行.

(4)更新统计信息.对于已观测节点,记录观测位置与时间,并丢弃过时信息,同时更新邻接节点的同位置信息,见图3中16行和27行.

图模型更新的复杂度分析.对于时间周期 t 的识读集 R_t,创建或更新节点的代价上界为 $|R_t|$.对于来自每个阅读器的识读集 $R_t^{(k)}$(k 为阅读器编号),针对邻接节点为相同观测位置新增边的代价小于 $|R_t^{(k)}/2|^2$.待更新边的集合 G_{t-1} 在 O_t 至少为其中一个邻接节点为观测节点的边上的投影,用 $\pi_{R_1,R_2,\cdots,R_k}(G)$ 表示,更新代价为 $2|\pi_{R_1,R_2,\cdots,R_k}(G)|$.因此,一个时间周期图数据更新的复杂度为 $O\left(\sum_k |R_t^{(k)}|^2+|\pi_{R_1,R_2,\cdots,R_k}(G)|\right)$,这意味着图更新代价不大于输入图在其中一个邻接节点为观测节点的边上投影集合的大小,再加上来自每个阅读器识读的个数的平方和.同时,由于RFID防冲撞协议以及实际部署,单个时间周期内一个RFID阅读器可识读的标签数量有限,为确保识读效果,实际部署的标签数量会更少些.

3 数据推导方法

在数据捕获阶段得到的更新图的基础上,利用基于概率分布的熵的方法来推导目标对象的位置和包含.对图 G 中的每个节点,根据该节点在所有可能位置上概率分布的熵来推导其最可能的位置,根据该节点的所有可能包含的概率分布的熵来推导其最可能的包含.

3.1 基于熵的概率推导方法

信息论中的熵是关于不确定性的一种度量,小的熵意味着随机变量更为确定.当熵很小时,就可认为该随机变量是确定的,并可将返回概率最大的对应值作为该随机变量的值.具有 n 个可能取值 $\{x_1,x_2,\cdots,x_n\}$ 的离散随机变量 X 的熵为

$$H(X)=-\sum_{i=1}^{n}p(x_i)\log_b p(x_i) \tag{1}$$

式中:$p(x_i)$ 为 X 取值 x_i 的概率质量函数.基于熵的定义,熵对随机变量 X 可能取值的个数是敏感的.例如,具有10个相同可能取值的随机变量的熵会远远高于具有2个相同可能取值的随机变量的熵.为了减轻这种影响,利用具有 n 个可能取值的最大熵对拥有 n 个可能取值的随机变量的熵进行归一化处理,亦即

$$H(X)=\left(-\sum_{i=1}^{n}p(x_i)\log_b p(x_i)\right)/\left(-\frac{1}{n}\log_b\left(\frac{1}{n}\right)\right) \tag{2}$$

3.2 包含关系推导

包含推导分为包含关系确认和一般包含关系推导.包含关系确认是确定在对象的所有可能包含中是否存在一个显著的包含.如果没有,则对其进行一般包含关系推导,以获得关于该对象包含关系的最可能估计,如图4所示.

输入:待进行包含关系推导的节点 v
输出:节点 v 的最可能包含 $e_{\max}$

```
1.  E_p ← 节点v的父连接边集
2.  For each E_p中的父连接边e_i
3.      w̄_{e_i} ← (Σ_{j=0}^{W} C_S[j])/Σ_{j=0}^{W} 1 //W为窗口大小
4.  End For
5.  h ← 节点v的父连接边的分布概率的熵
6.  For each E_p中的父连接边e_p
7.      p̄_{ei} ← w̄_{ei}对进行归一化处理
8.      h += p̄_{ej} lb p̄_{ej}
9.  End For
10. h /= (1/n)lb(1/n)//n为父连接边的个数
11. e_max ← E_p中概率p̄_e最大的连接边
12. If h < δ_h// δ_h为设定的阈值
13.   Return e_max
14. End If
15. For each E_p中的父连接边e_i
16.     w_{e_i} (Σ_{j=0}^{S} (C_S[j]/j^a))/Σ_{j=0}^{S} (1/j^a))
17. End For
18. For each E_p中的父连接边e_i
19.     p_{e_i} = ((1-β)m(e_i)+β_{w_{e_i}})/Z
20. End For
21. e_max ← E_p中概率p_e最大的连接边
22. p_{e_max} ← E_p中最大的概率p_e
23. Δ ← p_{e_max}与E_p中其他父边概率p_e的均方差
24. If Δ < δ_Δ //δ_Δ为设定的阈值
25.   Return e_max
26. End If
27. Return null
```

图4 节点包含关系推导算法

3.2.1 包含关系确认 文献[9]中假定特殊阅读器(如传送带阅读器等)事先已知,并且可以直接确认对象的包含,本文则通过找到一个特定的识读序列来识别出对象的真正包含.如传送带上的阅读器一次只识别一个箱子以及其所包含的单品,而且由于传送带和其他阅读器相距很远,因而该箱子及其所包含的单品不会被其他位置观测到.如果该趋势持续几个时间周期,就可确认该传送带上的箱子(唯一)就是这些单品的容器.给定节点 v,包含关系确认方法如下.

(1)计算权重.为节点的每条父连接边计算权重

$\overline{w}_{e_i}$(见图 4 中 2～4 行)

$$\overline{w}_{e_i} = \Big(\sum_{j=0}^{W} \boldsymbol{C}_S[j]\Big) / \sum_{j=0}^{W} 1 \tag{3}$$

式中:$\boldsymbol{C}_S[j]$为同位置向量的第 j 个设置位;W 为窗口大小. 由于太大的窗口会引入更多噪声而降低了边的权重的作用,使用较小的 W(本文取 $W=4$)较为理想.

(2)计算概率. 对节点的每条父边的权重 $\overline{w}_{e_i}$ 做归一化处理,得到 $\overline{p}_{e_i}$.

(3)计算熵. 计算节点的所有父边的概率分布的熵 h_i(见图 4 中 6～14 行).

$$h_i = \Big(-\sum_{j=1}^{n} \overline{p}_{e_j} \mathrm{lb}\overline{p}_{e_j}\Big) / \Big(-\frac{1}{n}\mathrm{lb}\frac{1}{n}\Big) \tag{4}$$

当节点的所有父边的概率分布的熵很小时,就可认返回概率最大的边作为该节点的确认边. 同时,更新变量 P_C,并删除该节点的其他所有父连接边,以节省系统的内存开销.

3.2.2　一般包含关系推导　如果在当前时间周期节点没有确认的包含,则对该节点进行一般包含关系推导,并选择概率值最大的父边作为其最可能的容器. 由于利用到了包括父连接边的同位置向量(即 $\boldsymbol{C}_S$)和表示节点之前已经确认的包含(即 P_C)等历史统计信息,该节点于当前时刻所丢失的 RFID 识读对于节点包含关系推导结果的影响就大大减小. 节点包含关系推导包括以下两个步骤.

(1)计算权重. 为每条父边计算权重 w_{e_i},计算方法[9]如下(见图 4 中 15～17 行)

$$w_{e_i} = \Big(\sum_{j=0}^{S} \frac{\boldsymbol{C}_S[j]}{j^{\alpha}}\Big) / \sum_{j=0}^{S} \frac{1}{j^{\alpha}} \tag{5}$$

式中:S 为位向量 $\boldsymbol{C}_S$ 的窗口大小(本文中 $S=32$);参数 α 控制不同时刻历史位置信息的权重,服从 Z 分布,$\alpha>0$ 给越新的历史赋予越高的权重,而 $\alpha=0$ 为历史信息中的各个位赋予相同的权重.

(2)计算概率. 计算节点 v 所有可能父连接边的概率分布. 如果父连接边中具有的最大概率和其他概率的均方差小于一个设定阈值,则返回该具有最大概率值的父边作为节点 v 的最可能父包含;否则,节点 v 的父包含是不确定的(见图 4 中的 18～27 行). 通过权衡连接边的相对权重和该连接边是否为已确认父边,计算每条边的概率,参数 β 则用来平衡这两个因素. 连接边 e_i 的概率 p_{e_i} 的计算方法[9]如下

$$p_{e_i} = ((1-\beta)m(e_i) + \beta w_{e_i})/Z \tag{6}$$

如果 e_i 为确认边,内存函数 $m(e_i)$取值 1,否则为 0. 参数 Z 为归一化因子.

3.3　位置推导

节点位置推导的关键就是在继续停留于原位置、移动到一个新的位置和不在任何位置而消失这 3 种可能情形之间进行权衡. 节点推导对节点 v 的所有可能位置构建概率分布,并根据最近 K 个时间周期的位置统计信息和邻居节点的位置统计信息,推导该节点当前最可能的位置. 节点的位置推导算法的伪代码如图 5 所示.

3.3.1　基于 RFID 数据的时态信息的位置推导　基于节点的历史位置信息 L_K 计算其所有可能位置上的概率分布,方法如下.

(1)计算权重. 对于节点 v 的位置统计信息中所有位置,逐个计算节点 v 仍可能处于位置 l_i 的权重 $w_{l_i}(v)$(见图 5 中 1～6 行). 节点在 T_O 处于位置 l_i 的权重的计算方法如下

$$w^t_{l_i}(v) = 1/(T_N - T_O + 1)^{\theta} \tag{7}$$

式中:T_N 为当前时间周期;T_O 为节点 v 所代表的物品在位置 l_i 被识别到的时间周期;参数 θ 控制节点 v 的最近位置 l_i 的衰减速度.

(2)计算概率. 对相同位置上的权重进行累加,并对所有可能位置的权重进行归一化处理,得到节点 v 可能所处位置的概率分布 $p_{l_i}(v)$.

(3)计算熵. 如同包含关系推导,基于节点所处位置的概率分布来计算熵 h_i. 如果该节点在某个位置上较为为显著(即位置概率分布的熵小于设定的阈值,并且该位置概率大于设定阈值)时,则可推导出该节点的最可能位置. 如果该节点所有位置的概率小于一个很小的阈值时,则可以推导出该节点的位置为未知(即丢失). 否则,该节点的位置为不确定,并推迟对该节点的位置推导到后续的“基于 RFID 数据的空间的位置推导”中进行,见图 5 中 7～19 行.

3.3.2　基于 RFID 数据的空间信息的位置推导　借助邻居节点的历史位置信息 L_K,对不确定节点 v_u 进行位置推导. 方法如下.

(1)构建局部子图. 首先向上遍历节点 v 的所有父节点,找到可能性最大的父节点 v_p. 如果 v_p 不是根节点,类推遍历父节点 v_p 的所有父节点,直到遍历到的父节点为根节点 v_r 为止. 然后,向下遍历根节点 v_r 的所有子节点,如果子节点 v_c 的最大可能父节点就是 v_r,则将该子节点 v_c 加入到子图中. 如此类推,向下递归遍历该子节点的所有子节点,直到加入到子图中的子节点为叶子节点为止,见图 5 中

20～26行. 为节点 I_5 构建的局部子图如图6所示，由双点划线所包围的节点及其之间概率最大的连接边(标有向下箭头)所构成.

(2)计算概率. 对上述局部子图中的所有节点于相同位置上的权重进行累加，并对所有可能位置的权重进行归一化处理，得到节点 v 可能所处位置的概率分布 $p_{ci}(v)$.

输入:待进行位置推导的节点v

1. For each L_K中观测位置位向量$\boldsymbol{L}$ //L_K为一HashMap，键为时间戳，值$\boldsymbol{L}$为位向量
2. For each $\boldsymbol{L}$中的观测位置l_i
3. $w_{l_i}^t=1/(T_N-T_O+1)^\theta$ //T_N为当前时间周期，T_O为观测时间周期，为控制参数
4. $w_{l_i}+=w_{l_i}^t$ //节点v处于观测位置l_i的权重
5. End For
6. End For
7. $h\leftarrow$节点v的所有可能位置的分布概率的熵
8. For each L_K中观测位置l_i
9. $\overline{p}_{l_i}\leftarrow$累加相同位置上权重$w_{l_i}$，并对所有位置的权重进行归一化处理
10. $h+=\overline{p}_{l_i}\mathrm{lb}\,\overline{p}_l$
11. End For
12. $h/=(1/n)\mathrm{lb}(1/n)$//归一化. n为可能位置的个数
13. $p_{\max}\leftarrow L_K$中最大的概率$\overline{p}_{l_i}$
14. $l_{\max}\leftarrow L_K$中最大的概率所对应的连接边
15. If $h<\delta_h$ //δ_h为设定的阈值
16. Return $l_{\max}$
17. Else If $p_{\max}<\delta_{\min}$
18. Return Unknown //节点v可能已丢失
19. End If
20. $G_{\mathrm{Sub}}\leftarrow$待构建的局部子图
21. $v_r\leftarrow v$
22. While v_r不是根节点 //即节点v_r存在父节点
23. $e_p\leftarrow$节点v_r所有父边中概率最大的那条边
24. $v_r\leftarrow$边e_p的父邻接节点
25. End While
26. call addChildNodes(v_r, G_{Sub})
27. $h\leftarrow$节点v的所有可能位置的分布概率的熵
28. If $h<\delta_h$ //δ_h为设定的阈值
29. Return $l_{\max}$
30. Else If $p_{\max}<\delta_{\min}$
31. Return Unknown //节点v可能已丢失
32. End If
33.
34. 子例程 addChildNodes(v, G_{Sub})
35. G_{Sub}.add(v)
36. For each e_c of v //对于节点v的每一条子边
37. $v_c\leftarrow$边e_c的子邻接节点
38. $e_p\leftarrow$节点v_c所有父边中概率最大的那条边
39. If $e_{ci}==e_p$
40. addChildNodes(v_c, G_{Sub}) //递归调用该例程
41. End If
42. End For

图5 节点位置推导算法

(3)计算熵. 如同在基于RFID数据的时态信息的位置推导，根据计算出的熵来判断该节点当前所处的位置(包括未知)，如果仍不能推导出节点的可能位置，则输出该节点的位置为不确定，见图5中27～32行.

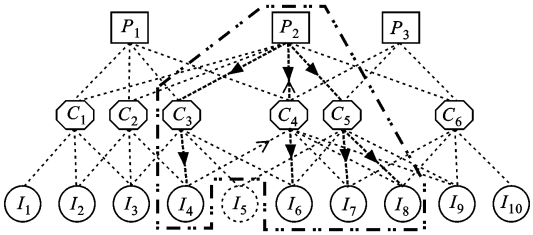


图6 局部子图的构建

3.4 数据推导的复杂度分析

节点包含关系推导的代价为$|E|$，对于基于时态信息的位置推导，其代价小于$|V|$，而对于基于空间信息的位置推导，其主要代价小于$|E|+|V|$(一般情况下满足$|V|<|E|$). 因此，数据推导的复杂度为$O(|E|)$. 为改善推导效率，可在推导过程中对模型进行修剪处理. 首先，如果一个物品已经通过仓库出口并有一段时间未被观测到，就可以从图中删除对应的节点及其所有邻接边. 其次，在对节点进行包含关系推导时，也可以剪除那些概率小于设定阈值的连接边.

4 性能评价

采用Java实现本文所提出的RFID数据捕获模型与数据推导方法，并在一个模拟的物流仓库环境中对该方法的准确性和效率进行评价.

4.1 RFID数据模拟器

为产生实验所需的数据，我们开发了一个在仓库部署RFID的数据模拟器. 在该模拟仓库中：①托盘以一定的速度注入到仓库，并将被仓库入口处的阅读器读到；②到达的托盘随后被解包为包装箱，这些包装箱被放置到入库传送带，并被传送带阅读器逐个扫描；③经过扫描后的包装箱将被放置到安装有阅读器的货架上，并在货架上停留一段时间；④待出货的包装箱被放置到出库传送带，并被传送带阅读器逐个扫描；⑤在安装有阅读器的打包区域，将待出货的包装箱重新打包成托盘；⑥托盘以一定的速度出库，并被仓库出口处的阅读器扫描到. 其中，物品(托盘或包装箱)在各个位置上的停留时间和在位置之间的迁移时间服从指数分布.

用于控制模拟数据生成的主要参数如下：托盘注入速率为每4～600 s注入1个托盘，托盘中箱子

的数量为 5～8 个,箱子中单品的数量为 20 个,模拟持续时间为 3～24 h,阅读器的识读率为 0.5～1,阅读器的交叉率为 0～0.8. 基于所产生的数据,数据推导算法以 1 s^{-1} 的频率对对象进行包含关系推导和位置推导.

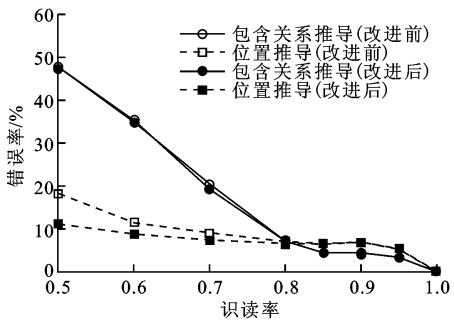
4.2　数据推导的准确性

基于针对包含关系推导和位置推导的实验结果可知,当 $S=32$、$\alpha=0$、$\beta=0.4$ 以及 $\theta=1.25$ 时,推导效果最好. 由于篇幅限制,这里略去相关实验结果图及分析.

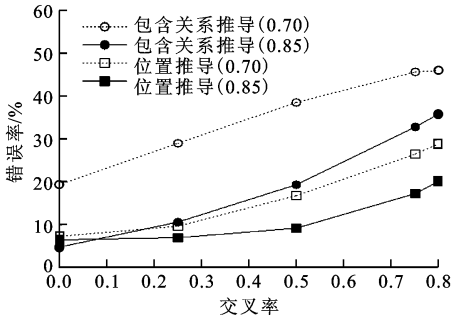
实验 1:不同识读率的影响. 本文提出的推导方法可充分利用最近 K 个时间周期的观测位置信息,即使识读率较低也可获得相当准确的对象的位置推导结果. 由于低识读率会影响包含关系的有效确认,同时还会造成邻接节点之间的同位置信息丢失,对象间的包含关系推导的准确性会随着识读率的降低存在较为明显的下降,如图 7a 所示. 对于不小于 0.8 的识读率,采用该改进方法两种推导错误率均低于 10%.

相比于改进前的方法[9](仅根据当前时刻的 RFID 识读来进行对象的位置推导),该改进后的方法除了支持处理 RFID 交叉读外,在对象位置推导准确性上也有明显提高,当识读率为 50%时推导准确性的改善可达 7%,如图 7a 所示. 由于该方法放松了文献[9]中关于特殊阅读器事先已知并可用于直接确认对象包含的假设,并采用基于对象邻接边的概率熵的包含确认方法,在一定程度上抵消了对象位置推导结果的改善对于对象间包含关系推导的促进作用. 因而,本文所提出的方法与文献[9]的方法在对象间的包含关系推导结果的准确性方面区别不是很明显,如图 7a 所示. 需要注意的是,本实验中的交叉率为 0. 由于文献[9]方法并不能处理 RFID 交叉读问题,对于存在交叉读的场景其推导错误率将更高.

实验 2:不同交叉率的影响. 如图 7b 所示,对于常见 RFID 部署(识读率为[0.8,1],交叉率为[0,0.25]),两种推导的错误率都可以控制在 10%以内. 当交叉率增大时,两种推导的错误率都随之增加. 过多的交叉读,会使得来自于最近阅读器的识读少于来自于交叉阅读器的识读,从而造成更多的位置推导错误,同时会使邻接节点的同位置信息中包含更多的噪声,使得用于包含关系确认的有效识读更少,从而导致了非常高的包含关系推导错误率,见图 7b 中位于顶部的曲线.



(a)变化识读率的影响



(b)变化交叉率的影响

图 7　位置推导和包含关系推导的错误率

4.3　数据推导的效率

从处理速度和内存使用来评价本文所提出方法的性能. 变化托盘的注入速率控制系统的吞吐量,变化托盘中包装箱个数以及模拟时间控制图模型中节点的规模. 本实验采用 2.33 GHz 的英特尔 Xeon 处理器与 8 GB 内存且运行 JVM1.6.0 的 Linux 服务器,最大 Java 分配池为 3 GB.

实验 3:处理速度. 表 1 为不同节点规模的图的处理时间. 随着节点数目的增加,每个时间周期(长度为 1 s)图更新代价和推导代价随之增大,而图更新和推导的代价都小于 1 s,其中推导代价的变化更为显著. 结果表明,该数据推导方法可以有效应对海量 RFID 数据流的处理需求.

表 1　图更新与推导的代价

节点数目	更新代价/s	推导代价/s	总代价/s
25 344	0.00 256	0.07 080	0.07 336
54 915	0.00 684	0.15 617	0.16 301
75 275	0.00 967	0.22 159	0.23 126
95 049	0.01 203	0.29 139	0.30 342
135 509	0.01 557	0.43 624	0.45 181
154 893	0.01 656	0.50 930	0.52 586
174 923	0.01 689	0.58 413	0.60 102

实验4:内存使用.数据推导中内存使用由图模型中节点的规模所决定.如图8所示,内存使用随着用于修剪连接边所设定阈值的增大而变小.当阈值取0.5时,图的大小能够保持在700 MB以下,此时节点数目可达175 000.当阈值设定为0.5和0.75时,内存的使用大致呈线性增长.同时我们也观察到,当设定的阈值不小于0.5时,修剪连接边对于位置推导错误率的影响并不显著,这对于内存严重不足的环境是可以接受的.

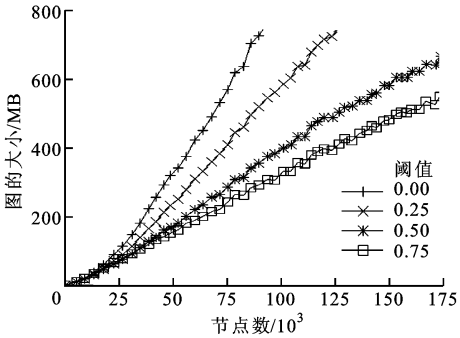


图8 图拥有不同节点数时的内存使用变化

5 结 语

本文提出的RFID数据流的概率推导方法,采用了图模型并辅以历史RFID识读捕获对象的位置与包含,利用基于熵的方法估计对象最有可能的位置与包含.针对经过基于RFID时态信息的位置推导后仍为不确定的节点,本文还采用了一种有效形成子图的方法,并利用空间关联信息来辅助不确定对象的位置推导.实验结果表明,本文所提出的方法可获得较准确的数据推导结果,同时具有良好的执行效率.

参考文献:

[1] RAO J, DORAISWAMY S, THAKKAR H, et al. A deferred cleansing method for RFID data analytics [C]// Proceedings of the 32nd International Conference on Very large Data Bases. Almaden, CA, USA: VLDB Endowment Inc., 2006: 175-186.

[2] BAI Y, WANG F, LIU P. Efficiently filtering RFID data streams [C]// Proceedings of the 1st International VLDB Workshop on Clean Databases. Almaden, CA, USA: VLDB Endowment Inc., 2006: 50-57.

[3] JEFFERY S, ALONSO G, FRANKLIN M, et al. A pipelined framework for online cleaning of sensor data streams[C]// Proceedings of the 22nd International Conference on Data Engineering. Piscataway, NJ, USA: IEEE, 2006: 140.

[4] JEFFERY S, GAROFALAKIS M, FRANKLIN M. Adaptive cleaning for RFID data streams [C]// Proceedings of the 32nd International Conference on Very Large Data Bases. Almaden, CA, USA: VLDB Endowment Inc., 2006: 163-174.

[5] 谷峪,于戈,李晓静,等. 基于动态概率路径事件模型的RFID数据填补算法[J]. 软件学报,2010,21(3):438-451.
GU Yu, YU Ge, LI Xiaojing, et al. RFID data interpolation algorithm based on dynamic probabilistic path-event model[J]. Journal of Software, 2010, 21(3): 438-451.

[6] GU Yu, YU Ge, CHEN Yueguo, et al. Efficient RFID data imputation by analyzing the correlations of monitored objects [C]// Proceedings of the 14th International Conference on Database Systems for Advanced Applications. Berlin, Germany: Springer-Verlag, 2009: 186-200.

[7] 谷峪,于戈,胡小龙,等. 基于监控对象动态聚簇的高效RFID数据清洗模型[J]. 软件学报,2010,21(4):632-643.
GU Yu, YU Ge, HU Xiaolong, et al. Efficient RFID data cleaning model based on dynamic clusters of monitored objects[J]. Journal of Software, 2010, 21(4): 632-643.

[8] TRAN T, SUTTON C, COCCI R, et al. Probabilistic inference over RFID streams in mobile environments [C]// Proceedings of the 25th International Conference on Data Engineering. Piscataway, NJ, USA: IEEE, 2009: 1096-1107.

[9] COCCI R, TRAN T, DIAO Y, et al. Efficient data interpretation and compression over RFID streams [C]// Proceedings of the 24th International Conference on Data Engineering. Piscataway, NJ, USA: IEEE, 2008: 1445-1447.

[10] EPCglobal Inc. EPC Tag data standard version 1.5 [EB/OL]. [2010-08-18]. http://www.gs1.org/gsmp/kc/epcglobal/tds/tds_1_5-standard-20100818.pdf.

(编辑 杜秀杰)