

Manage the Data from Indoor Spaces: Models, Indexes & Query Processing

Huan Li

Database Laboratory, Zhejiang University

lihuancs@zju.edu.cn

March 24, 2016

Overview

1. Outlines
2. Indoor Space Models & Applications
3. Indoor Data Cleansing
4. Indoor Movement Analysis
5. Appendix

1. Outlines
2. Indoor Space Models & Applications
3. Indoor Data Cleansing
4. Indoor Movement Analysis
5. Appendix

1. Outlines
2. Indoor Space Models & Applications
3. Indoor Data Cleansing
4. Indoor Movement Analysis
5. Appendix

About This Work...

Graph Model Based Indoor Tracking. [1]

C. S. Jensen, H. Lu, and B. Yang.

- Published in year 2009, *MDM* conference.
- A pioneering work that introduces base graph model to indoor data management.
- Detailed tracking algorithms are designed for RFID-based positioning.
- Easy to understand, with comprehensive concepts.

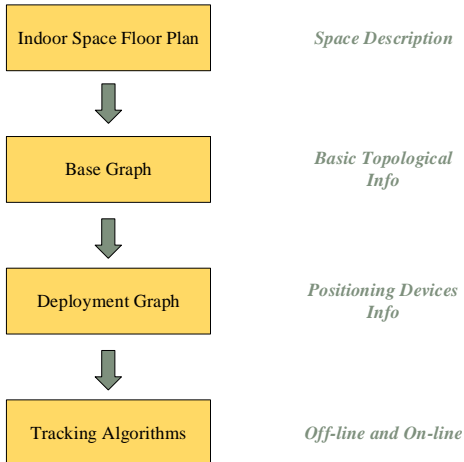
2.1 Graph Model Based Indoor Tracking

Motivation

- We are spending most of our time in indoor spaces
 - Office building, University, Shopping Centers, etc.
- We cannot use GPS-based tracking indoor movements
 - Indoor navigation and route guidance (museum)
 - Flow analysis
 - how do people use the indoor space → important in pricing of advertisement space in store rental
- We can use other technology...
 - Wi-Fi, Infrared, Bluetooth or RFID
 - This paper is focusing on RFID, since it is now mature and effortless
 - RFID tags are cheap and RFID reader are expensive

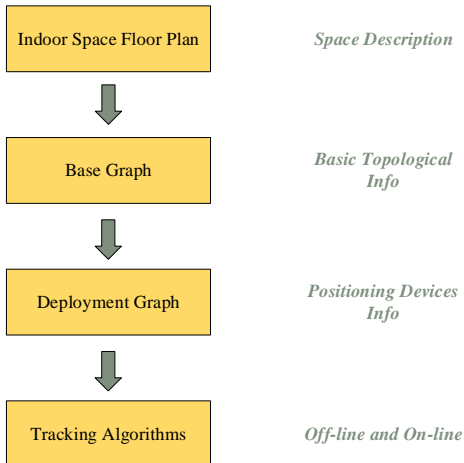
2.1 Graph Model Based Indoor Tracking

Idea



2.1 Graph Model Based Indoor Tracking

Idea

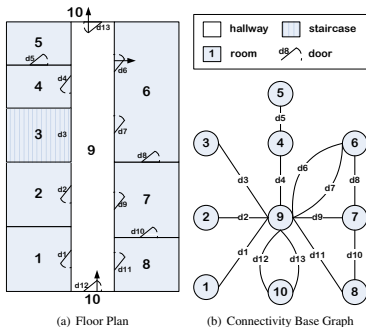


Goal: Improve indoor tracking accuracy from a data management perspective, to capture where a particular object can be at a particular time.

2.1 Graph Model Based Indoor Tracking

Base Graph Model

By capturing the essential connectivity and accessibility, **Base Graph** describes the topology of a floor plan of a possibly complex indoor space.



Connectivity Base Graph

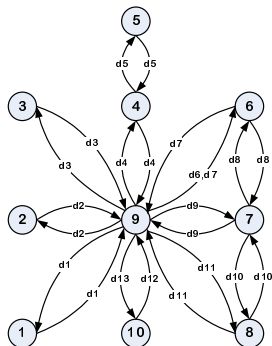
a labeled, undirected graph.

- $G_{conn} = \{V, E_d, \Sigma_{door}\}$
- V : each separate partition is represented as a vertex
- E_d : each door is captured as an edge
- Σ_{door} : a set of edge labels that represent connections

2.1 Graph Model Based Indoor Tracking

Base Graph Model

Accessibility Graph is constructed to represent the movement permitted by doors or connections.



Accessibility Graph

a labeled, directed graph.

- $G_{accs} = \{V, E, \Sigma_{door}, l_e\}$
- V : the set of vertices
- E : the set of directed edges, i.e.,
 $E = \{\langle v_i, v_j \rangle | v_i, v_j \in V \wedge v_i \neq v_j\}$
- l_e : a function that maps edges to subsets of the set of doors, i.e.,
 $l_e : E \rightarrow 2^{\Sigma_{door}}$

Base Graph Model

In addition to the topological information of a floor plan, its geometrical information should also be captured.

Base Graph Model

In addition to the topological information of a floor plan, its geometrical information should also be captured.

The *Building Partitions Mapping* is defined as:

Base Graph Model

In addition to the topological information of a floor plan, its geometrical information should also be captured.

The *Building Partitions Mapping* is defined as:

$$BuildingPartitions : V \rightarrow Ploygons \quad (1)$$

Base Graph Model

In addition to the topological information of a floor plan, its geometrical information should also be captured.

The *Building Partitions Mapping* is defined as:

$$\textit{BuildingPartitions} : V \rightarrow \textit{Ploygons} \quad (1)$$

The *Doors Mapping* is defined as:

Base Graph Model

In addition to the topological information of a floor plan, its geometrical information should also be captured.

The *Building Partitions Mapping* is defined as:

$$BuildingPartitions : V \rightarrow Ploygons \quad (1)$$

The *Doors Mapping* is defined as:

$$Doors : \Sigma_{door} \rightarrow Line\ Segments \quad (2)$$

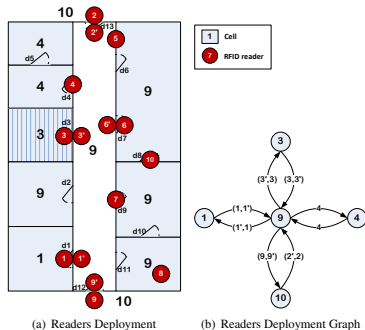
RFID Deployment Graph Model

- RFID based proximity analysis
 - RFID readers deployment may cover only part of the space, or it may be capable of only detecting some movements in the space.
 - assume that all RFID readers have disjoint activation ranges.
- Types of RFID readers
 - **Partitioning Readers** partition the indoor space into cells in the sense that an object cannot move from one cell to another without being observed.
 - **Presence Readers** simply observe the presence(and non-presence) of tags in their activation ranges.

2.1 Graph Model Based Indoor Tracking

RFID Deployment Graph Model

Vertices represent cells. A directed edge indicates that one can move from one cell to another without entering other cells, which is detected by a corresponding partitioning reader.



RFID Deployment Graph

a labeled, directed graph.

- $G_{RFID} = \{C, E_r, \Sigma_{reader}, l_e\}$
- C : the set of the vertices
- E_r : An edge is an ordered pair $\langle c_i, c_j \rangle$ of distinct vertices from C
- l_e maps an edge to a partitioning reader (pair), i.e.,

$$E_r \rightarrow 2^{\Sigma_{reader}} \cup 2^{\Sigma_{reader} \times \Sigma_{reader}}$$

2.1 Graph Model Based Indoor Tracking

RFID Deployment Graph Construction

Each cell created by partitioning readers corresponds to one or more base graph partitions.

2.1 Graph Model Based Indoor Tracking

RFID Deployment Graph Construction

Each cell created by partitioning readers corresponds to one or more base graph partitions.

$$Cells : V \rightarrow C \quad (3)$$

2.1 Graph Model Based Indoor Tracking

RFID Deployment Graph Construction

Each cell created by partitioning readers corresponds to one or more base graph partitions.

$$Cells : V \rightarrow C \quad (3)$$

For each RFID reader, record its accurate deployment location and activation range.

2.1 Graph Model Based Indoor Tracking

RFID Deployment Graph Construction

Each cell created by partitioning readers corresponds to one or more base graph partitions.

$$Cells : V \rightarrow C \quad (3)$$

For each RFID reader, record its accurate deployment location and activation range.

$$\begin{aligned} Mapping\ 1 : \Sigma_{reader} \rightarrow \{ & (loc, range, flag) \mid loc \in R^2 \wedge \\ & range \in (0, d_{max}] \wedge flag \in \{PAR, PRE\} \} \end{aligned} \quad (4)$$

2.1 Graph Model Based Indoor Tracking

RFID Deployment Graph Construction

Each cell created by partitioning readers corresponds to one or more base graph partitions.

$$Cells : V \rightarrow C \quad (3)$$

For each RFID reader, record its accurate deployment location and activation range.

$$Mapping\ 1 : \Sigma_{reader} \rightarrow \{(loc, range, flag) \mid loc \in R^2 \wedge range \in (0, d_{max}] \wedge flag \in \{PAR, PRE\}\} \quad (4)$$

A mapping of readers to the cells that their activation ranges intersect is introduced as:

2.1 Graph Model Based Indoor Tracking

RFID Deployment Graph Construction

Each cell created by partitioning readers corresponds to one or more base graph partitions.

$$Cells : V \rightarrow C \quad (3)$$

For each RFID reader, record its accurate deployment location and activation range.

$$Mapping\ 1 : \Sigma_{reader} \rightarrow \{(loc, range, flag) \mid loc \in R^2 \wedge range \in (0, d_{max}] \wedge flag \in \{PAR, PRE\}\} \quad (4)$$

A mapping of readers to the cells that their activation ranges intersect is introduced as:

$$Mapping\ 2 : \Sigma_{reader} \rightarrow 2^C \quad (5)$$

2.1 Graph Model Based Indoor Tracking

RFID Deployment Graph Construction

Algorithm 1 RFIDGraphConstruction (Readers R , ConnectivityBaseGraph G_{conn} , AccessibilityGraph G_{accs})

```

1: Readers  $R' \leftarrow \emptyset$ ;  $DR(\langle \Sigma_{door} k', Readers RSet \rangle) \leftarrow \emptyset$ ;
   Connected Component  $CCs \leftarrow \emptyset$ ;  $int\ m \leftarrow 0$ ;
2:  $G_{RFID}(C, E_r, \Sigma_{reader} l_e) \leftarrow (\emptyset, \emptyset, R, NULL)$ ;
3: for each edge  $d_a$  in  $G_{conn}.E_d$  do
4:   for each reader  $r_b$  in  $R$  do
5:     if Circle(Mapping1( $r_b$ ).loc, Mapping1( $r_b$ ).range) covers
       Doors( $d_a.k$ ) then
6:       insert  $r_b$  into  $R'$ ;
7:   if  $|R'| > 0$  then
8:     delete  $d_a$  from  $G_{conn}.E_d$ ; insert  $(d_a.k, R')$  into  $DR$ ;
        $R' \leftarrow \emptyset$ ;
9: store all connected components of  $G_{conn}$  in  $CCs$ ;
10: for each connected component  $cc_c$  in  $CCs$  do
11:   create a new vertex  $c_m$  and add it to  $G_{RFID}.C$ ;  $m++$ ;
12:   for each vertex  $v_x$  in the vertices of  $cc_c$  do
13:     add the mapping  $(v_x \rightarrow c_m)$  to  $Cells$ ;
14: for each  $dr_n$  in  $DR$  do
15:   for each  $e_l$  in  $G_{accs}.l_e^{-1}(dr_n.k')$  do
16:      $c_p \leftarrow Cells(e_l.v_i)$ ;  $c_q \leftarrow Cells(e_l.v_j)$ ;
17:     if  $c_p \neq c_q$  then
18:       if  $\langle c_p, c_q \rangle$  is not in  $G_{RFID}.E_r$  then
19:         add  $\langle c_p, c_q \rangle$  to  $G_{RFID}.E_r$ ;
20:       add the mapping  $(\langle c_p, c_q \rangle \rightarrow$ 
         readersequence( $dr_n.RSet$ )) to  $G_{RFID}.l_e$ ;

```

2.1 Graph Model Based Indoor Tracking

RFID Deployment Graph Construction

Algorithm 1 RFIDGraphConstruction (Readers R , ConnectivityBaseGraph G_{conn} , AccessibilityGraph G_{accs})

```

1: Readers  $R' \leftarrow \emptyset$ ;  $DR(\langle \Sigma_{door} k', Readers RSet \rangle) \leftarrow \emptyset$ ;
   Connected Component  $CCs \leftarrow \emptyset$ ;  $\text{int } m \leftarrow 0$ ;
2:  $G_{RFID}(C, E_r, \Sigma_{reader} l_e) \leftarrow (\emptyset, \emptyset, R, NULL)$ ;
3: for each edge  $d_a$  in  $G_{conn}.E_d$  do
4:   for each reader  $r_b$  in  $R$  do
5:     if Circle(Mapping1( $r_b$ ).loc, Mapping1( $r_b$ ).range) covers
       Doors( $d_a.k$ ) then
6:       insert  $r_b$  into  $R'$ ;
7:   if  $|R'| > 0$  then
8:     delete  $d_a$  from  $G_{conn}.E_d$ ; insert  $(d_a.k, R')$  into  $DR$ ;
        $R' \leftarrow \emptyset$ ;
9: store all connected components of  $G_{conn}$  in  $CCs$ ;
10: for each connected component  $cc_c$  in  $CCs$  do
11:   create a new vertex  $c_m$  and add it to  $G_{RFID}.C$ ;  $m++$ ;
12:   for each vertex  $v_x$  in the vertices of  $cc_c$  do
13:     add the mapping  $(v_x \rightarrow c_m)$  to  $Cells$ ;
14: for each  $dr_n$  in  $DR$  do
15:   for each  $e_l$  in  $G_{accs}.l_e^{-1}(dr_n.k')$  do
16:      $c_p \leftarrow Cells(e_l.v_i)$ ;  $c_q \leftarrow Cells(e_l.v_j)$ ;
17:     if  $c_p \neq c_q$  then
18:       if  $\langle c_p, c_q \rangle$  is not in  $G_{RFID}.E_r$  then
19:         add  $\langle c_p, c_q \rangle$  to  $G_{RFID}.E_r$ ;
20:     add the mapping  $(\langle c_p, c_q \rangle \rightarrow$ 
       readersequence( $dr_n.RSet$ )) to  $G_{RFID}.l_e$ ;

```

① Input: the reader set R , the connectivity base graph G_{conn} , the accessibility graph G_{accs}

2.1 Graph Model Based Indoor Tracking

RFID Deployment Graph Construction

Algorithm 1 RFIDGraphConstruction (Readers R , ConnectivityBaseGraph G_{conn} , AccessibilityGraph G_{accs})

```

1: Readers  $R' \leftarrow \emptyset$ ;  $DR(\langle \Sigma_{door} k', Readers RSet \rangle) \leftarrow \emptyset$ ;
   Connected Component  $CCs \leftarrow \emptyset$ ;  $int\ m \leftarrow 0$ ;
2:  $G_{RFID}(C, E_r, \Sigma_{reader} l_e) \leftarrow (\emptyset, \emptyset, R, NULL)$ ;
3: for each edge  $d_a$  in  $G_{conn}.E_d$  do
4:   for each reader  $r_b$  in  $R$  do
5:     if Circle(Mapping1( $r_b$ ).loc, Mapping1( $r_b$ ).range) covers
       Doors( $d_a.k$ ) then
6:       insert  $r_b$  into  $R'$ ;
7:   if  $|R'| > 0$  then
8:     delete  $d_a$  from  $G_{conn}.E_d$ ; insert  $(d_a.k, R')$  into  $DR$ ;
        $R' \leftarrow \emptyset$ ;
9: store all connected components of  $G_{conn}$  in  $CCs$ ;
10: for each connected component  $cc_c$  in  $CCs$  do
11:   create a new vertex  $c_m$  and add it to  $G_{RFID}.C$ ;  $m++$ ;
12:   for each vertex  $v_x$  in the vertices of  $cc_c$  do
13:     add the mapping  $(v_x \rightarrow c_m)$  to  $Cells$ ;
14: for each  $dr_n$  in  $DR$  do
15:   for each  $e_l$  in  $G_{accs}.l_e^{-1}(dr_n.k')$  do
16:      $c_p \leftarrow Cells(e_l.v_i)$ ;  $c_q \leftarrow Cells(e_l.v_j)$ ;
17:     if  $c_p \neq c_q$  then
18:       if  $\langle c_p, c_q \rangle$  is not in  $G_{RFID}.E_r$  then
19:         add  $\langle c_p, c_q \rangle$  to  $G_{RFID}.E_r$ ;
20:     add the mapping  $(\langle c_p, c_q \rangle \rightarrow$ 
       readersequence( $dr_n.RSet$ )) to  $G_{RFID}.l_e$ ;
```

- ① Input: the reader set R , the connectivity base graph G_{conn} , the accessibility graph G_{accs}
- ② Lines 1–2: Initialize G_{RFID} , DR , CCs

2.1 Graph Model Based Indoor Tracking

RFID Deployment Graph Construction

Algorithm 1 RFIDGraphConstruction (Readers R , ConnectivityBaseGraph G_{conn} , AccessibilityGraph G_{accs})

```

1: Readers  $R' \leftarrow \emptyset$ ;  $DR(\langle \Sigma_{door} k', Readers RSet \rangle) \leftarrow \emptyset$ ;
   ConnectedComponent  $CCs \leftarrow \emptyset$ ;  $int\ m \leftarrow 0$ ;
2:  $G_{RFID}(C, E_r, \Sigma_{reader} l_e) \leftarrow (\emptyset, \emptyset, R, NULL)$ ;
3: for each edge  $d_a$  in  $G_{conn}.E_d$  do
4:   for each reader  $r_b$  in  $R$  do
5:     if Circle(Mapping1( $r_b$ ).loc, Mapping1( $r_b$ ).range) covers
       Doors( $d_a.k$ ) then
6:       insert  $r_b$  into  $R'$ ;
7:   if  $|R'| > 0$  then
8:     delete  $d_a$  from  $G_{conn}.E_d$ ; insert  $(d_a.k, R')$  into  $DR$ ;
        $R' \leftarrow \emptyset$ ;
9: store all connected components of  $G_{conn}$  in  $CCs$ ;
10: for each connected component  $cc_c$  in  $CCs$  do
11:   create a new vertex  $c_m$  and add it to  $G_{RFID}.C$ ;  $m++$ ;
12:   for each vertex  $v_x$  in the vertices of  $cc_c$  do
13:     add the mapping  $(v_x \rightarrow c_m)$  to  $Cells$ ;
14: for each  $dr_n$  in  $DR$  do
15:   for each  $e_l$  in  $G_{accs}.l_e^{-1}(dr_n.k')$  do
16:      $c_p \leftarrow Cells(e_l.v_i)$ ;  $c_q \leftarrow Cells(e_l.v_j)$ ;
17:     if  $c_p \neq c_q$  then
18:       if  $\langle c_p, c_q \rangle$  is not in  $G_{RFID}.E_r$  then
19:         add  $\langle c_p, c_q \rangle$  to  $G_{RFID}.E_r$ ;
20:     add the mapping  $(\langle c_p, c_q \rangle \rightarrow$ 
       readersequence( $dr_n.RSet$ )) to  $G_{RFID}.l_e$ ;

```

- ① Input: the reader set R , the connectivity base graph G_{conn} , the accessibility graph G_{accs}
- ② Lines 1–2: Initialize G_{RFID} , DR , CCs
- ③ Lines 3–8: the relationship of which door is covered by which readers is captured in DR

2.1 Graph Model Based Indoor Tracking

RFID Deployment Graph Construction

Algorithm 1 RFIDGraphConstruction (Readers R , ConnectivityBaseGraph G_{conn} , AccessibilityGraph G_{accs})

```

1: Readers  $R' \leftarrow \emptyset$ ;  $DR(\langle \Sigma_{door} k', Readers RSet \rangle) \leftarrow \emptyset$ ;
   ConnectedComponent  $CCs \leftarrow \emptyset$ ;  $int\ m \leftarrow 0$ ;
2:  $G_{RFID}(C, E_r, \Sigma_{reader} l_e) \leftarrow (\emptyset, \emptyset, R, NULL)$ ;
3: for each edge  $d_a$  in  $G_{conn}.E_d$  do
4:   for each reader  $r_b$  in  $R$  do
5:     if Circle(Mapping1( $r_b$ ).loc, Mapping1( $r_b$ ).range) covers
       Doors( $d_a.k$ ) then
6:       insert  $r_b$  into  $R'$ ;
7:   if  $|R'| > 0$  then
8:     delete  $d_a$  from  $G_{conn}.E_d$ ; insert  $(d_a.k, R')$  into  $DR$ ;
        $R' \leftarrow \emptyset$ ;
9: store all connected components of  $G_{conn}$  in  $CCs$ ;
10: for each connected component  $cc_c$  in  $CCs$  do
11:   create a new vertex  $c_m$  and add it to  $G_{RFID}.C$ ;  $m++$ ;
12:   for each vertex  $v_x$  in the vertices of  $cc_c$  do
13:     add the mapping  $(v_x \rightarrow c_m)$  to  $Cells$ ;
14: for each  $dr_n$  in  $DR$  do
15:   for each  $e_l$  in  $G_{accs}.l_e^{-1}(dr_n.k')$  do
16:      $c_p \leftarrow Cells(e_l.v_i)$ ;  $c_q \leftarrow Cells(e_l.v_j)$ ;
17:     if  $c_p \neq c_q$  then
18:       if  $\langle c_p, c_q \rangle$  is not in  $G_{RFID}.E_r$  then
19:         add  $\langle c_p, c_q \rangle$  to  $G_{RFID}.E_r$ ;
20:     add the mapping  $(\langle c_p, c_q \rangle \rightarrow$ 
       readersequence( $dr_n.RSet$ )) to  $G_{RFID}.l_e$ ;

```

- ① Input: the reader set R , the connectivity base graph G_{conn} , the accessibility graph G_{accs}
- ② Lines 1–2: Initialize G_{RFID} , DR , CCs
- ③ Lines 3–8: the relationship of which door is covered by which readers is captured in DR
- ④ Lines 9–13: a deployment graph vertex is created for each CC , mapping $Cells$ is also stored

2.1 Graph Model Based Indoor Tracking

RFID Deployment Graph Construction

Algorithm 1 RFIDGraphConstruction (Readers R , ConnectivityBaseGraph G_{conn} , AccessibilityGraph G_{accs})

```

1: Readers  $R' \leftarrow \emptyset$ ;  $DR((\sum_{door} k', Readers\ RSet)) \leftarrow \emptyset$ ;
   ConnectedComponent  $CCs \leftarrow \emptyset$ ;  $int\ m \leftarrow 0$ ;
2:  $G_{RFID}(C, E_r, \sum_{reader} l_e) \leftarrow (\emptyset, \emptyset, R, NULL)$ ;
3: for each edge  $d_a$  in  $G_{conn}.E_d$  do
4:   for each reader  $r_b$  in  $R$  do
5:     if Circle(Mapping1( $r_b$ ).loc, Mapping1( $r_b$ ).range) covers
       Doors( $d_a.k$ ) then
6:       insert  $r_b$  into  $R'$ ;
7:   if  $|R'| > 0$  then
8:     delete  $d_a$  from  $G_{conn}.E_d$ ; insert  $(d_a.k, R')$  into  $DR$ ;
        $R' \leftarrow \emptyset$ ;
9: store all connected components of  $G_{conn}$  in  $CCs$ ;
10: for each connected component  $cc_c$  in  $CCs$  do
11:   create a new vertex  $c_m$  and add it to  $G_{RFID}.C$ ;  $m++$ ;
12:   for each vertex  $v_x$  in the vertices of  $cc_c$  do
13:     add the mapping  $(v_x \rightarrow c_m)$  to  $Cells$ ;
14: for each  $dr_n$  in  $DR$  do
15:   for each  $e_l$  in  $G_{accs}.l_e^{-1}(dr_n.k')$  do
16:      $c_p \leftarrow Cells(e_l.v_i)$ ;  $c_q \leftarrow Cells(e_l.v_j)$ ;
17:     if  $c_p \neq c_q$  then
18:       if  $\langle c_p, c_q \rangle$  is not in  $G_{RFID}.E_r$  then
19:         add  $\langle c_p, c_q \rangle$  to  $G_{RFID}.E_r$ ;
20:     add the mapping  $(\langle c_p, c_q \rangle \rightarrow$ 
       readersequence( $dr_n.RSet$ )) to  $G_{RFID}.l_e$ ;

```

- ① Input: the reader set R , the connectivity base graph G_{conn} , the accessibility graph G_{accs}
- ② Lines 1–2: Initialize G_{RFID} , DR , CCs
- ③ Lines 3–8: the relationship of which door is covered by which readers is captured in DR
- ④ Lines 9–13: a deployment graph vertex is created for each CC , mapping $Cells$ is also stored
- ⑤ Lines 14–20: for each door in DR , determine if its edges' head and tail are mapped to different cells. If so, add an edge to deployment graph. Function *readersequence* returns the possible reader sequence for that edge

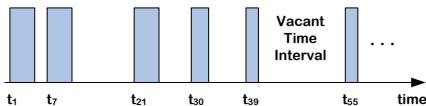
2.1 Graph Model Based Indoor Tracking

RFID-based Indoor Tracking

Raw Trajectories: Sequences of RFID Tag Observation

Table 1. Raw Trajectory of tag_1

readerID	tagID	t	readerID	tagID	t
reader ₉	tag ₁	t ₁	reader ₁₀	tag ₁	t ₄₀
reader ₉	tag ₁	t ₂	reader ₈	tag ₁	t ₅₅
reader _{9'}	tag ₁	t ₃	reader ₈	tag ₁	t ₅₆
reader _{9'}	tag ₁	t ₄	reader _{3'}	tag ₁	t ₇₉
reader _{1'}	tag ₁	t ₇	reader _{3'}	tag ₁	t ₈₀
reader _{1'}	tag ₁	t ₈	reader ₄	tag ₁	t ₈₅
reader ₁	tag ₁	t ₉	reader ₄	tag ₁	t ₈₆
reader ₁	tag ₁	t ₁₀	reader ₄	tag ₁	t ₈₇
reader ₁	tag ₁	t ₂₁	reader ₄	tag ₁	t ₁₀₀
reader ₁	tag ₁	t ₂₂	reader ₄	tag ₁	t ₁₀₁
reader _{1'}	tag ₁	t ₂₃	reader ₄	tag ₁	t ₁₀₂
reader _{1'}	tag ₁	t ₂₄	reader _{6'}	tag ₁	t ₁₀₉
reader ₇	tag ₁	t ₃₀	reader _{6'}	tag ₁	t ₁₁₀
reader ₇	tag ₁	t ₃₁	reader ₆	tag ₁	t ₁₁₁
reader ₇	tag ₁	t ₃₂	reader ₆	tag ₁	t ₁₁₂
reader ₁₀	tag ₁	t ₃₉	reader ₅	tag ₁	t ₁₂₅



1. each reader detects and reports tags with a sampling rate
2. formatted as $\langle readerID, tagID, t \rangle$

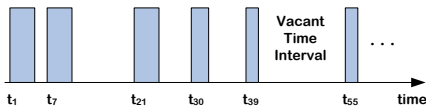
2.1 Graph Model Based Indoor Tracking

RFID-based Indoor Tracking

Raw Trajectories: Sequences of RFID Tag Observation

Table 1. Raw Trajectory of tag_1

readerID	tagID	t	readerID	tagID	t
reader ₉	tag ₁	t ₁	reader ₁₀	tag ₁	t ₄₀
reader ₉	tag ₁	t ₂	reader ₈	tag ₁	t ₅₅
reader _{9'}	tag ₁	t ₃	reader ₈	tag ₁	t ₅₆
reader _{9'}	tag ₁	t ₄	reader _{3'}	tag ₁	t ₇₉
reader _{1'}	tag ₁	t ₇	reader _{3'}	tag ₁	t ₈₀
reader _{1'}	tag ₁	t ₈	reader ₄	tag ₁	t ₈₅
reader ₁	tag ₁	t ₉	reader ₄	tag ₁	t ₈₆
reader ₁	tag ₁	t ₁₀	reader ₄	tag ₁	t ₈₇
reader ₁	tag ₁	t ₂₁	reader ₄	tag ₁	t ₁₀₀
reader ₁	tag ₁	t ₂₂	reader ₄	tag ₁	t ₁₀₁
reader _{1'}	tag ₁	t ₂₃	reader ₄	tag ₁	t ₁₀₂
reader _{1'}	tag ₁	t ₂₄	reader _{6'}	tag ₁	t ₁₀₉
reader ₇	tag ₁	t ₃₀	reader _{6'}	tag ₁	t ₁₁₀
reader ₇	tag ₁	t ₃₁	reader ₆	tag ₁	t ₁₁₁
reader ₇	tag ₁	t ₃₂	reader ₆	tag ₁	t ₁₁₂
reader ₁₀	tag ₁	t ₃₉	reader ₅	tag ₁	t ₁₂₅



● *vacant time intervals*: unable to observe the moving objects

1. each reader detects and reports tags with a sampling rate
2. formatted as $\langle readerID, tagID, t \rangle$

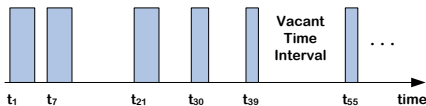
2.1 Graph Model Based Indoor Tracking

RFID-based Indoor Tracking

Raw Trajectories: Sequences of RFID Tag Observation

Table 1. Raw Trajectory of tag_1

readerID	tagID	t	readerID	tagID	t
reader ₉	tag ₁	t ₁	reader ₁₀	tag ₁	t ₄₀
reader ₉	tag ₁	t ₂	reader ₈	tag ₁	t ₅₅
reader _{9'}	tag ₁	t ₃	reader ₈	tag ₁	t ₅₆
reader _{9'}	tag ₁	t ₄	reader _{3'}	tag ₁	t ₇₉
reader _{1'}	tag ₁	t ₇	reader _{3'}	tag ₁	t ₈₀
reader _{1'}	tag ₁	t ₈	reader ₄	tag ₁	t ₈₅
reader ₁	tag ₁	t ₉	reader ₄	tag ₁	t ₈₆
reader ₁	tag ₁	t ₁₀	reader ₄	tag ₁	t ₈₇
reader ₁	tag ₁	t ₂₁	reader ₄	tag ₁	t ₁₀₀
reader ₁	tag ₁	t ₂₂	reader ₄	tag ₁	t ₁₀₁
reader _{1'}	tag ₁	t ₂₃	reader ₄	tag ₁	t ₁₀₂
reader _{1'}	tag ₁	t ₂₄	reader _{6'}	tag ₁	t ₁₀₉
reader ₇	tag ₁	t ₃₀	reader _{6'}	tag ₁	t ₁₁₀
reader ₇	tag ₁	t ₃₁	reader ₆	tag ₁	t ₁₁₁
reader ₇	tag ₁	t ₃₂	reader ₆	tag ₁	t ₁₁₂
reader ₁₀	tag ₁	t ₃₉	reader ₅	tag ₁	t ₁₂₅



- *vacant time intervals*: unable to observe the moving objects
- to search RFID deployment graph to infer the possible regions of moving object

1. each reader detects and reports tags with a sampling rate
2. formatted as $\langle readerID, tagID, t \rangle$

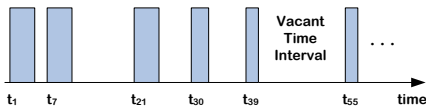
2.1 Graph Model Based Indoor Tracking

RFID-based Indoor Tracking

Raw Trajectories: Sequences of RFID Tag Observation

Table 1. Raw Trajectory of tag_1

readerID	tagID	t	readerID	tagID	t
reader ₉	tag ₁	t ₁	reader ₁₀	tag ₁	t ₄₀
reader ₉	tag ₁	t ₂	reader ₈	tag ₁	t ₅₅
reader _{9'}	tag ₁	t ₃	reader ₈	tag ₁	t ₅₆
reader _{9'}	tag ₁	t ₄	reader _{3'}	tag ₁	t ₇₉
reader _{1'}	tag ₁	t ₇	reader _{3'}	tag ₁	t ₈₀
reader _{1'}	tag ₁	t ₈	reader ₄	tag ₁	t ₈₅
reader ₁	tag ₁	t ₉	reader ₄	tag ₁	t ₈₆
reader ₁	tag ₁	t ₁₀	reader ₄	tag ₁	t ₈₇
reader ₁	tag ₁	t ₂₁	reader ₄	tag ₁	t ₁₀₀
reader ₁	tag ₁	t ₂₂	reader ₄	tag ₁	t ₁₀₁
reader _{1'}	tag ₁	t ₂₃	reader ₄	tag ₁	t ₁₀₂
reader _{1'}	tag ₁	t ₂₄	reader _{6'}	tag ₁	t ₁₀₉
reader ₇	tag ₁	t ₃₀	reader _{6'}	tag ₁	t ₁₁₀
reader ₇	tag ₁	t ₃₁	reader ₆	tag ₁	t ₁₁₁
reader ₇	tag ₁	t ₃₂	reader ₆	tag ₁	t ₁₁₂
reader ₁₀	tag ₁	t ₃₉	reader ₅	tag ₁	t ₁₂₅



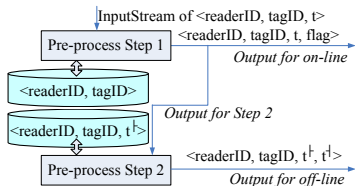
- *vacant time intervals*: unable to observe the moving objects
- to search RFID deployment graph to infer the possible regions of moving object
- to apply maximum speed position interpolation to further shrink the possible regions [2, 3]

1. each reader detects and reports tags with a sampling rate
2. formatted as $\langle readerID, tagID, t \rangle$

2.1 Graph Model Based Indoor Tracking

RFID Readings Pre-processing

Step-1's output is used in on-line tracking, while Step-2's is used in off-line tracking



- $Flag \in \{START, END\}$
- $START \rightarrow$ enters the range
- $END \rightarrow$ leaves the range

2.1 Graph Model Based Indoor Tracking

Off-line Tracking (Refinement Step 1)

Table 2. Off-Line Trajectory of tag_1

readingID	readerID	tagID	t^-	t^+
$reading_1$	$reader_9$	tag_1	t_1	t_2
$reading_2$	$reader_9'$	tag_1	t_3	t_4
$reading_3$	$reader_1'$	tag_1	t_7	t_8
$reading_4$	$reader_1$	tag_1	t_9	t_{10}
$reading_5$	$reader_1$	tag_1	t_{21}	t_{22}
$reading_6$	$reader_1'$	tag_1	t_{23}	t_{24}
$reading_7$	$reader_7$	tag_1	t_{30}	t_{32}
$reading_8$	$reader_{10}$	tag_1	t_{39}	t_{40}
$reading_9$	$reader_8$	tag_1	t_{55}	t_{56}
$reading_{10}$	$reader_{3'}$	tag_1	t_{79}	t_{80}
$reading_{11}$	$reader_4$	tag_1	t_{85}	t_{87}
$reading_{12}$	$reader_4$	tag_1	t_{100}	t_{102}
$reading_{13}$	$reader_{6'}$	tag_1	t_{109}	t_{110}
$reading_{14}$	$reader_6$	tag_1	t_{111}	t_{112}
$reading_{15}$	$reader_5$	tag_1	t_{125}	t_{125}

Table 3. Trajectory After Step 1

tagID	$[t^-, t^+]$	Graph Element	Reader ₁	Reader ₂
tag_1	$[t_1, t_4]$	$e(10, 9)$	$reader_9$	$reader_9'$
tag_1	$[t_7, t_{10}]$	$e(9, 1)$	$reader_1'$	$reader_1$
tag_1	$[t_{21}, t_{24}]$	$e(1, 9)$	$reader_1$	$reader_1'$
tag_1	$[t_{30}, t_{32}]$	c_9	$reader_7$	$reader_7$
tag_1	$[t_{39}, t_{40}]$	c_9	$reader_{10}$	$reader_{10}$
tag_1	$[t_{55}, t_{56}]$	c_9	$reader_8$	$reader_8$
tag_1	$[t_{79}, t_{80}]$	c_9	$reader_{3'}$	$reader_{3'}$
tag_1	$[t_{85}, t_{87}]$	$e(4, 9), e(9, 4)$	$reader_4$	$reader_4$
tag_1	$[t_{100}, t_{102}]$	$e(4, 9), e(9, 4)$	$reader_4$	$reader_4$
tag_1	$[t_{109}, t_{112}]$	c_9	$reader_{6'}$	$reader_6$
tag_1	$[t_{125}, t_{125}]$	c_9	$reader_5$	$reader_5$

- Step 1 transforms an RFID reading sequence to corresponding vertices or edges in deployment graph
- If two consecutive reading sequences are *contiguous*, they should stem from a partitioning pair, which map to an edge
- Otherwise, should come from either a single *PRE* or a *PAR* reader
- A *PAR* is replaced by the set of corresponding edges according to $G_{RFID}.l_e^{-1}$
- A *PRE* always corresponds to one or several cells according to *Mapping 2*

2.1 Graph Model Based Indoor Tracking

Off-line Tracking (Refinement Step 2)

Table 3. Trajectory After Step 1

tagID	$[t^-, t^+]$	Graph Element	Reader ₁	Reader ₂
tag ₁	$[t_1, t_4]$	$e(10, 9)$	reader ₉	reader _{9'}
tag ₁	$[t_7, t_{10}]$	$e(9, 1)$	reader _{1'}	reader ₁
tag ₁	$[t_{21}, t_{24}]$	$e(1, 9)$	reader ₁	reader _{1'}
tag ₁	$[t_{30}, t_{32}]$	c_9	reader ₇	reader ₇
tag ₁	$[t_{39}, t_{40}]$	c_9	reader ₁₀	reader ₁₀
tag ₁	$[t_{55}, t_{56}]$	c_9	reader ₈	reader ₈
tag ₁	$[t_{79}, t_{80}]$	c_9	reader _{3'}	reader _{3'}
tag ₁	$[t_{85}, t_{87}]$	$e(4, 9), e(9, 4)$	reader ₄	reader ₄
tag ₁	$[t_{100}, t_{102}]$	$e(4, 9), e(9, 4)$	reader ₄	reader ₄
tag ₁	$[t_{109}, t_{112}]$	c_9	reader _{6'}	reader ₆
tag ₁	$[t_{125}, t_{125}]$	c_9	reader ₅	reader ₅

Table 4. Trajectory After Steps 2 and 3

tagID	$[t^-, t^+]$	Step 2	Step 3
tag ₁	$[t_5, t_6]$	c_9	$c_9 \cap \Theta(\text{reader}_{9'}, \text{reader}_{1'}, t_5, t_6)$
tag ₁	$[t_{11}, t_{20}]$	c_1	$c_1 \cap \Theta(\text{reader}_1, \text{reader}_1, t_{11}, t_{20})$
tag ₁	$[t_{25}, t_{29}]$	c_9	$c_9 \cap \Theta(\text{reader}_{1'}, \text{reader}_7, t_{25}, t_{29})$
tag ₁	$[t_{33}, t_{38}]$	c_9	$c_9 \cap \Theta(\text{reader}_7, \text{reader}_{10}, t_{33}, t_{38})$
tag ₁	$[t_{41}, t_{54}]$	c_9	$c_9 \cap \Theta(\text{reader}_{10}, \text{readers}, t_{41}, t_{54})$
tag ₁	$[t_{57}, t_{78}]$	c_9	$c_9 \cap \Theta(\text{readers}, \text{reader}_{3'}, t_{57}, t_{78})$
tag ₁	$[t_{81}, t_{84}]$	c_9	$c_9 \cap \Theta(\text{reader}_{3'}, \text{reader}_4, t_{81}, t_{84})$
tag ₁	$[t_{88}, t_{99}]$	c_4, c_9	$(c_4, c_9) \cap \Theta(\text{reader}_4, \text{reader}_4, t_{88}, t_{99})$
tag ₁	$[t_{103}, t_{108}]$	c_9	$c_9 \cap \Theta(\text{reader}_4, \text{reader}_{6'}, t_{103}, t_{108})$
tag ₁	$[t_{113}, t_{114}]$	c_9	$c_9 \cap \Theta(\text{reader}_6, \text{reader}_5, t_{113}, t_{114})$

- The *graph elements* from Step 1 indicates some region(s) within which the object may be in during the vacant time interval
- Check its previous record's tail element and current record's head element, select their intersection as Step 2's candidate

2.1 Graph Model Based Indoor Tracking

Off-line Tracking (Refinement Step 3)

- Calculate the *possible region* Θ according to maximum speed limit

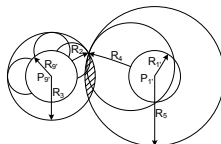
- Circle based possible region

- locations: $P_{9'}$, $P_{1'}$
- activation ranges: $R_{9'}$, $R_{1'}$
- for $t_x \in [t_5, t_6]$,
 $\Delta t_1 = t_x - reading_2.t^{\top}$,
 $\Delta t_2 = reading_3.t^{\top} - t_x$
- $R_3 = R_{9'} + V_{max} * \Delta t_1$,
 $R_5 = R_{1'} + V_{max} * \Delta t_2$

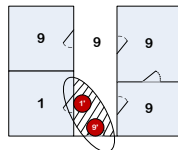
- Ellipse based possible region

- foci: two points belonging to the circle centered at $P_{9'}$, $P_{1'}$
- length of major axis is:

$$2a = V_{max} * (\Delta t_1 + \Delta t_2)$$



(a) Circle based possible region



(b) Ellipse based possible region

2.1 Graph Model Based Indoor Tracking

Off-line Tracking (Refinement Step 3)

- Calculate the *possible region* Θ according to maximum speed limit

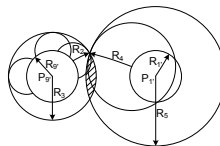
- Circle based possible region

- locations: $P_{g'}$, $P_{1'}$
- activation ranges: $R_{g'}$, $R_{1'}$
- for $t_x \in [t_5, t_6]$,
 $\Delta t_1 = t_x - \text{reading}_2.t^\top$,
 $\Delta t_2 = \text{reading}_3.t^\top - t_x$
- $R_3 = R_{g'} + V_{max} * \Delta t_1$,
 $R_5 = R_{1'} + V_{max} * \Delta t_2$

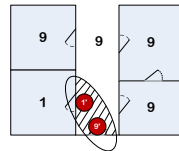
- Ellipse based possible region

- foci: two points belonging to the circle centered at $P_{g'}$, $P_{1'}$
- length of major axis is:

$$2a = V_{max} * (\Delta t_1 + \Delta t_2)$$



(a) Circle based possible region



(b) Ellipse based possible region

$$\left. \begin{array}{l} (\text{reading}_1, \text{reader}_9, \text{tag}_1, t_1, t_2) \\ (\text{reading}_2, \text{reader}_{g'}, \text{tag}_1, t_3, t_4) \\ (\text{reading}_3, \text{reader}_{1'}, \text{tag}_1, t_7, t_8) \\ (\text{reading}_4, \text{reader}_1, \text{tag}_1, t_9, t_{10}) \end{array} \right\} \xrightarrow{\text{Step 1}}$$

2.1 Graph Model Based Indoor Tracking

Off-line Tracking (Refinement Step 3)

- Calculate the *possible region* Θ according to maximum speed limit

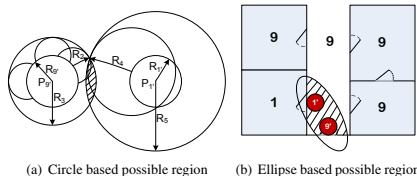
- Circle based possible region

- locations: $P_{g'}$, $P_{1'}$
- activation ranges: $R_{g'}$, $R_{1'}$
- for $t_x \in [t_5, t_6]$,
 $\Delta t_1 = t_x - \text{reading}_2.t^\top$,
 $\Delta t_2 = \text{reading}_3.t^\top - t_x$
- $R_3 = R_{g'} + V_{max} * \Delta t_1$,
 $R_5 = R_{1'} + V_{max} * \Delta t_2$

- Ellipse based possible region

- foci: two points belonging to the circle centered at $P_{g'}$, $P_{1'}$
- length of major axis is:

$$2a = V_{max} * (\Delta t_1 + \Delta t_2)$$



$$\left. \begin{array}{l} (\text{reading}_1, \text{reader}_9, \text{tag}_1, t_1, t_2) \\ (\text{reading}_2, \text{reader}_{g'}, \text{tag}_1, t_3, t_4) \\ (\text{reading}_3, \text{reader}_{1'}, \text{tag}_1, t_7, t_8) \\ (\text{reading}_4, \text{reader}_1, \text{tag}_1, t_9, t_{10}) \end{array} \right\} \xrightarrow{\text{Step 1}}$$

$$\left. \begin{array}{l} (\text{tag}_1, [t_1, t_4], e\langle 10, 9 \rangle, \text{reader}_9, \text{reader}_{g'}) \\ (\text{tag}_1, [t_7, t_{10}], e\langle 9, 1 \rangle, \text{reader}_{1'}, \text{reader}_1) \end{array} \right\} \xrightarrow{\text{Step 2}}$$

2.1 Graph Model Based Indoor Tracking

Off-line Tracking (Refinement Step 3)

- Calculate the *possible region* Θ according to maximum speed limit

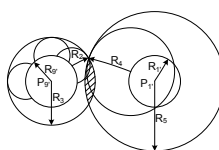
- Circle based possible region

- locations: $P_{g'}$, $P_{1'}$
- activation ranges: $R_{g'}$, $R_{1'}$
- for $t_x \in [t_5, t_6]$,
 $\Delta t_1 = t_x - \text{reading}_2.t^\top$,
 $\Delta t_2 = \text{reading}_3.t^\top - t_x$
- $R_3 = R_{g'} + V_{max} * \Delta t_1$,
 $R_5 = R_{1'} + V_{max} * \Delta t_2$

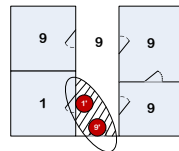
- Ellipse based possible region

- foci: two points belonging to the circle centered at $P_{g'}$, $P_{1'}$
- length of major axis is:

$$2a = V_{max} * (\Delta t_1 + \Delta t_2)$$



(a) Circle based possible region



(b) Ellipse based possible region

$$\left. \begin{array}{l} (\text{reading}_1, \text{reader}_9, \text{tag}_1, t_1, t_2) \\ (\text{reading}_2, \text{reader}_{g'}, \text{tag}_1, t_3, t_4) \\ (\text{reading}_3, \text{reader}_{1'}, \text{tag}_1, t_7, t_8) \\ (\text{reading}_4, \text{reader}_1, \text{tag}_1, t_9, t_{10}) \end{array} \right\} \xrightarrow{\text{Step 1}}$$

$$\left. \begin{array}{l} (\text{tag}_1, [t_1, t_4], e\langle 10, 9 \rangle, \text{reader}_9, \text{reader}_{g'}) \\ (\text{tag}_1, [t_7, t_{10}], e\langle 9, 1 \rangle, \text{reader}_{1'}, \text{reader}_1) \end{array} \right\} \xrightarrow{\text{Step 2}}$$

$$(\text{tag}_1, [t_5, t_6], c_9, \text{reader}_{g'}, \text{reader}_{1'}) \xrightarrow{\text{Step 3}}$$

2.1 Graph Model Based Indoor Tracking

Off-line Tracking (Refinement Step 3)

- Calculate the *possible region* Θ according to maximum speed limit

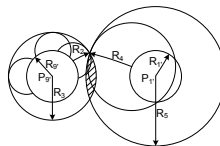
- Circle based possible region

- locations: $P_{g'}$, $P_{1'}$
- activation ranges: $R_{g'}$, $R_{1'}$
- for $t_x \in [t_5, t_6]$,
 $\Delta t_1 = t_x - \text{reading}_2.t^\top$,
 $\Delta t_2 = \text{reading}_3.t^\top - t_x$
- $R_3 = R_{g'} + V_{max} * \Delta t_1$,
 $R_5 = R_{1'} + V_{max} * \Delta t_2$

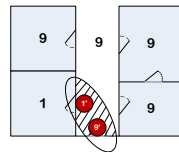
- Ellipse based possible region

- foci: two points belonging to the circle centered at $P_{g'}$, $P_{1'}$
- length of major axis is:

$$2a = V_{max} * (\Delta t_1 + \Delta t_2)$$



(a) Circle based possible region



(b) Ellipse based possible region

$$\left. \begin{array}{l} (\text{reading}_1, \text{reader}_9, \text{tag}_1, t_1, t_2) \\ (\text{reading}_2, \text{reader}_{g'}, \text{tag}_1, t_3, t_4) \\ (\text{reading}_3, \text{reader}_{1'}, \text{tag}_1, t_7, t_8) \\ (\text{reading}_4, \text{reader}_1, \text{tag}_1, t_9, t_{10}) \end{array} \right\} \xrightarrow{\text{Step 1}}$$

$$\left. \begin{array}{l} (\text{tag}_1, [t_1, t_4], e\langle 10, 9 \rangle, \text{reader}_9, \text{reader}_{g'}) \\ (\text{tag}_1, [t_7, t_{10}], e\langle 9, 1 \rangle, \text{reader}_{1'}, \text{reader}_1) \end{array} \right\} \xrightarrow{\text{Step 2}}$$

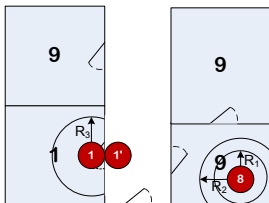
$$(\text{tag}_1, [t_5, t_6], c_9, \text{reader}_{g'}, \text{reader}_{1'}) \xrightarrow{\text{Step 3}}$$

$$(\text{tag}_1, [t_5, t_6], c_9 \cup \Theta(\text{reader}_{g'}, \text{reader}_{1'}, t_5, t_6))$$

2.1 Graph Model Based Indoor Tracking

On-line Tracking

Given $\langle readerID, tagID, t, flag \rangle$. **On-line tracking** is intended to infer the trajectory in the time interval between the last observation and the current time or even in the future.



- $flag = START$, object $tagID$ is in the activation range of $readerID$ at time t .
- $flag = END$, the object is beyond the activation range of $readerID$ and not in the range of any other readers.
 - constrains by a circle determined by the most recent observing reader's range.
 - further refined if an object has recently been detected by a partitioning reader pair.

2.1 Graph Model Based Indoor Tracking

Research Directions

- Extend the deployment graphs to accommodate RFID readers with large and overlapping activation ranges.
- Using multiple deployment graphs for several positioning technologies.
- To enhance on-line tracking. Historical data → association rules → better prediction.

References I

- [1] C. S. Jensen, H. Lu, and B. Yang.
Graph model based indoor tracking.
In *MDM*, pp. 122–131, 2009.
- [2] A. Civilis, C. S. Jensen and S. Pakalnis.
Techniques for efficient road-network-based tracking of moving objects.
In *TKDE*, pp. 698–712, 2005.
- [3] D. Pfoser, and C. S. Jensen.
Capturing the uncertainty of moving-object representations.
In *Advances in Spatial Databases*, pp. 111–131, 1999.

1. Outlines
2. Indoor Space Models & Applications
3. Indoor Data Cleansing
4. Indoor Movement Analysis
5. Appendix

1. Outlines
2. Indoor Space Models & Applications
3. Indoor Data Cleansing
- 4. Indoor Movement Analysis**
5. Appendix

1. Outlines
2. Indoor Space Models & Applications
3. Indoor Data Cleansing
4. Indoor Movement Analysis
5. Appendix

The End. Thanks :)