International Journal of Multimedia and Ubiquitous Engineering
Vol.9, No.4 (2014), pp.157-170
http://dx.doi.org/10.14257/ijmue.2014.9.4.17

A Multi-Granularity Grid-Based Graph Model for Indoor Space

Na Wang[1], Peiquan Jin[1,2], Yan Xiong[1] and Lihua Yue[1,2]

[1]*School of Computer Science and Technology,*
University of Science and Technology of China
[2]*Key Laboratory of Electromagnetic Space Information,*
Chinese Academy of Sciences

jpq@ustc.edu.cn

Abstract

With the development of the Internet of Things (IoT) and indoor positioning technologies, how to manage the rapidly increasing indoor moving object data is a research topic. The fundamental issue is to consider the representation of complicated indoor space as well as the moving object. In this paper, we aim to propose a complete indoor space model that can support various indoor applications. The proposed model divides each indoor element into multi-granularity grids and introduces grids as indoor primitive geometry for the representation of connectivity between indoor elements and location of moving objects. The advantage of this approach is that it can simultaneously represent the connective relationship and geometric information of indoor elements, and give more accurate description of indoor distance and direction information. Several queries of indoor applications under the shopping mall scenario are employed to illustrate the completeness and robustness of our model.

Keywords: *Indoor space; Moving objects; Model; Grids*

1. Introduction

With the development of indoor positioning technologies and various portable devices, it has brought a demand of diverse indoor location based services. For example, the shopping mall intends to provide navigation services and personalized recommendation for customers; the coal manager needs to track the movement of coal miners under the mine in case of emergency. Therefore, how to manage the indoor moving object data and answer various indoor queries has been a recent research topic [1, 2].

Previous researches mainly focus on modeling moving objects in outdoor space and especially on spatiotemporal changes [3]. However, the movement of humans in indoor environment is constrained by doors, walls and obstacles, and then the calculation of indoor distance needs to consider the connectivity of indoor elements. Besides, indoor positioning usually provides relative locations which need to consult the indoor geometry. Due to the specialty of indoor space, various kinds of indoor space data model have been presented in several domains, such as cognitive navigation services with object feature model [4, 5], and the indoor space design and visualization using geometric models [6-11], and topological studies of indoor elements with symbolic models [12-17]. However, these model representations usually focus on their own domains, and cannot be directly applied to other applications fields. The other disadvantage of existing models is the calculation of indoor distance which is based on door-to-door Dijkstra distance out consideration of indoor obstacles. Generally speaking, little

ISSN: 1975-0080 IJMUE

attention has been paid to construct an indoor space model which can represent indoor semantics as complete as possible to support diverse indoor location based queries.

In this paper, we analyze relationships between indoor entities, and propose a grid-based indoor primitive geometry as intermediary for representation of indoor semantics. Then we present the formalization of the grid-based graph model called InMGG_Model, which considers both topological and geometric indoor features, and support several fine-granularity indoor applications.

2. Related work

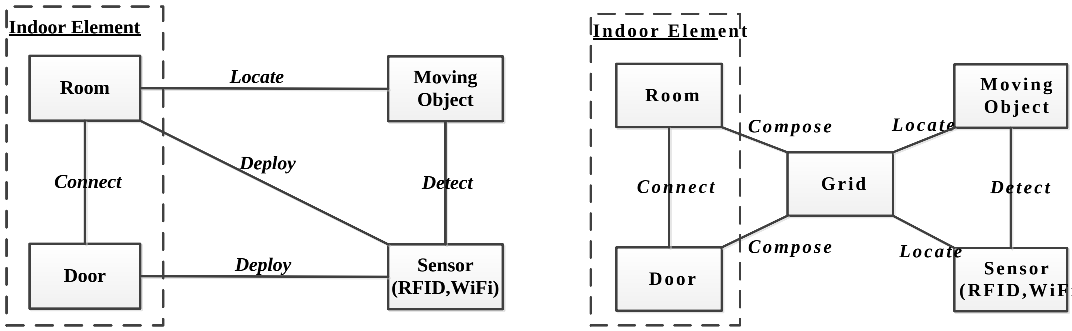
Current methods of indoor space modelling can be classified into three categories, object feature models, geometric models and symbolic models. Among them, object feature models mainly describe attributes and relationships of entities, which includes the UML-based IndoorML [4] and the entity-relationship based ONALIN model [5]. However, they cannot answer indoor distance or direction related query due to the lack of indoor geometrical features. The geometric model focuses on the geometrical expression of indoor space. In the literature [6], the authors propose to divide indoor space into regular rectangles. In [7-9], there are also some irregular division methods. The 2D-3D hybrid model proposed in [10] records geometric bounds of indoor elements. The prismatic model in [11] utilize prism to describe rooms. As these models lack the description of indoor connectivity, it cannot support the fine-granularity indoor queries such as indoor navigation.

The symbolic model uses a unique symbol to represent each indoor element, and describes topological relationships between these symbols. Becker proposed the set-based model [12] to represent the inclusion between indoor elements, which is convenient for range query. The topology-based semantic model [13] represents indoor elements as a single set of objects for indoor analysis. A lattice-based semantic model was presented in [14], which use lattice structure to represent indoor space and mainly support indoor navigation. The most popular graph-based model [15-17] utilizes an undirected graph to represent the structural properties of indoor environment where humans behave, where nodes represent rooms [15] or doors [16] or both [17]. However, it neglects geometric features of indoor entity, and the symbolic representation only supports a rough calculation of indoor distance. Besides, it also cannot support navigation visualization in robotics domain or indoor emergency diffusion analysis.

3. Framework of the Multi-Granularity Grid Graph Model

Indoor space data model needs to describe three kinds of indoor entities, including indoor elements, sensor deployment and moving object data. Figure 1(a) is the entity-relationship graph of traditional symbolic model. *Indoor element* includes *Room* and *Door*, and *Room* is connected with *Door*; *Moving Objects* are detected by *Sensor*; *Sensor* is deployed in *Room* or *Door*; *Moving Object* is located in *Room*. As indoor element is abstracted as symbol, there is no geometric data of indoor element, which leads to symbolic moving object location and inaccurate indoor distance calculation.

To address this problem, we propose to add entity *Grid* as indoor space primitive geometry (as shown in Figure 1(b)). *Room* and *Door* are composed with *Grids*, which can represent geometry information. *Sensor* and *Moving Objects* are located in *Grids*, which are more accurate than symbolic locations.



(a) Symbolic model **(b) Proposed model**

Figure 1. Entity-relationship diagram of proposed model

3.1. Indoor grids partition

In the literature [18], Li *et al.* proposed to consider the whole 2D indoor space as the model input, and partition it into uniform continuous grids. However, the geometrical feature of each indoor element is varied and a fixed level of granularity cannot adapt every indoor element. In this paper, we consider each indoor element as the model input, and confirm its unique partitioning way according to the geometric feature. It not only can avoid the storage and computational complexity led by uniformly fine-grained grids, but also sufficiently retain the indoor geometric features to ensure the precision of indoor distance calculation.

When humans encounter obstacles or turning points in indoor space, they need to adjust their path and find a new direction to go ahead. In order to reflect this behavior pattern in our representation model, we generate original grids according to these features. The partitioning algorithm **ConstructMultiGranuGrid** (as shown in Alogrithm.1) consists of three main steps (as illustrated in Figure 2):

- Generate original grids: determine the minimum enclosing rectangle of indoor element (line 1 in Algorithm.1) and partition it into fine-grained grid matrix (line 2 to 7 in Algorithm.1);

  — For each indoor obstacles, prolong the horizontal and vertical bounds;

  — For each indoor turning point, prolong the perpendicular bounds at its location;

- Determine obstacle grids (line 8 to 10 in Algorithm.1): distinguish obstacle grids in grid matrix;

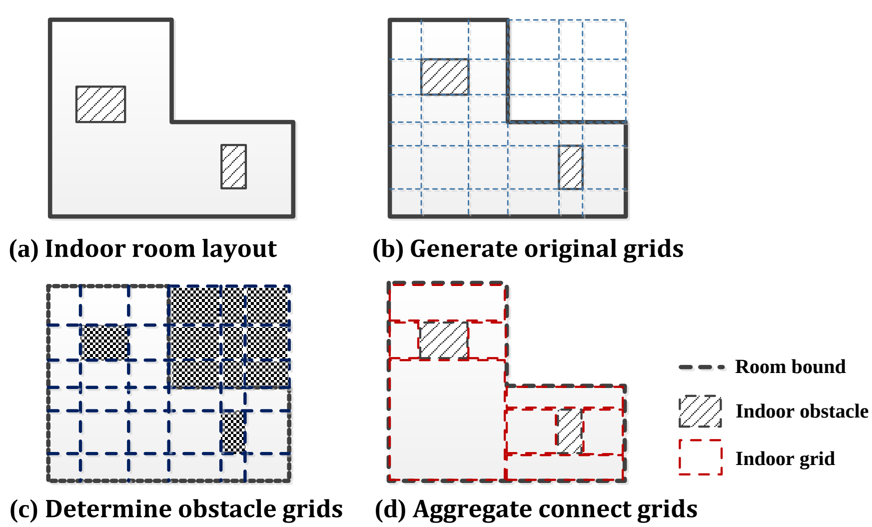
- Aggregate connect grids (line 11 to 15 in Algorithm.1): Utilize the greedy algorithm **AggMaxSubMartix** (as shown in Algorithm.2) to fetch the maximum full connect sub grid matrix in each loop (line 1 to 12 in Algorithm.2), and aggregate it into a larger grid (line 13 in Algorithm.2).

Algorithms.1: ConstructMultiGranuGrid (R)
Input: Indoor room layout R
Output: PGS, the set of multi-granularity grids of R
Begin
1 Determine the bounding rectangle of R to be divided
2 **For** each obstacle ot in R **Do** //make grids
3   Extend the boundary horizontal and vertical lines of the bounding rectangle of ot
4 **End For**

```
**For** each turning point tp of R **Do**
    Extend a horizontal line and a vertical line that pass through tp
**End For**
**For** each grid G of M*N grid matrix GM generated in step 1 to 6 **Do**
    Determine G is a connect or obstacle grid //distinguish obstacle grids
**End For**
**While** M*N grid matrix has connect grids **Do** //aggregate the maximum sub-matrix with all connect grids into a large grid PG
    PG ← AggMaxSubMartix(GM)
    PGS ← PGS ∪ {PG}
    The grids in the maximum sub-matrix are set to be obstacle for next aggregation
**End While**
**Return** PGS
**End**
```

As shown in the aggregating algorithm **AggMaxSubMartix**, in order to fetch the full connect sub grid matrix with the maximum number of indoor grids in each while loop, firstly map the M*N grid matrix into the M*N 0-1 binary matrix (line 1 in Algorithm.2), in which 0 represents the obstacle grid and 1 represents the non-obstacle. Then we can devote to the solution of the maximum full-1 sub matrix in 0-1 matrix instead of finding the maximum sub grid matrix (line 2 to 12 in Algorithm.2).

Given the 0-1 binary matrix B_{M*N}, firstly traverse each row $i \in [1..M]$ from top to down (line 2 in Algorithm.2): for each 1-elemnt e existed in column $j \in [1..N]$ of this row, count up the number of 1-element in the above of e (including e) as $c[j]$ (line 3 to 5 in Algorithm.2); for each non-zero element $c[j]$ of array $c[1..N]$, find the bound index of non-zero and non-smaller element on both left and right sides of $c[j]$ as $l[j]$ and $r[j]$ (line 6 to 8 in Algorithm.2); it can be seen that there exists a full-1 sub matrix $(i - c[j] + 1, i, l[j], r[j])$ above element e, in which row $i \in [i - c[j] + 1..i]$, and column $j \in [l[j]..r[j]]$ (line 9 in Algorithm.2). Thus, we record the maximum full-1 sub matrix in the previous step of the whole loop as (mt, mb, mr, ml), which is the maximum full-1 sub matrix in the binary 0-1 matrix B_{M*N} (line 12 in Algorithm.2). Figure 3 illustrates the progress of fixing the maximum full-1 sub matrix of a 4-order binary matrix.



(a) Indoor room layout (b) Generate original grids (c) Determine obstacle grids (d) Aggregate connect grids

Figure 2. An example of constructing multi-granularity grids partition

Algorithm 2: AggMaxSubMartix(GM)
Input: GM, a M*N dimension grid matrix
Output: PG, the large grid aggregated from maximum sub-matrix of GM
Begin
1 Correspond GM into 0-1 matrix BGM which represent obstacle/connect status of grids
2 **For** each row i of BGM **Do**
3 **For** each 1-element e in column j of row i **Do**
4 Add up the count of consecutive 1-element in the same column above e as $c[j]$
5 **End for**
6 **For** each nonzero element $c[j]$ in array $c[1..\mathrm{N}]$ **Do**
7 Find the first nonzero and smaller element $c[k_1]$ on the left of $c[j]$, $l[j] \leftarrow k_1 + 1$
8 Find the first nonzero and smaller element $c[k_2]$ on the right of $c[j]$, $r[j] \leftarrow k_2 - 1$
9 Calculate the number of grids in sub-matrix $(i - c[j] + 1, i, l[j], r[j])$ with the formula $(r[j] - l[j] + 1) * c[j]$
10 **End for**
11 **End for**
12 Record the maximum sub-matrix (mt, mb, mr, ml) in step 9
13 Aggregate all grids in (mt, mb, mr, ml) and construct a large grid GM
14 **Return** GM
End

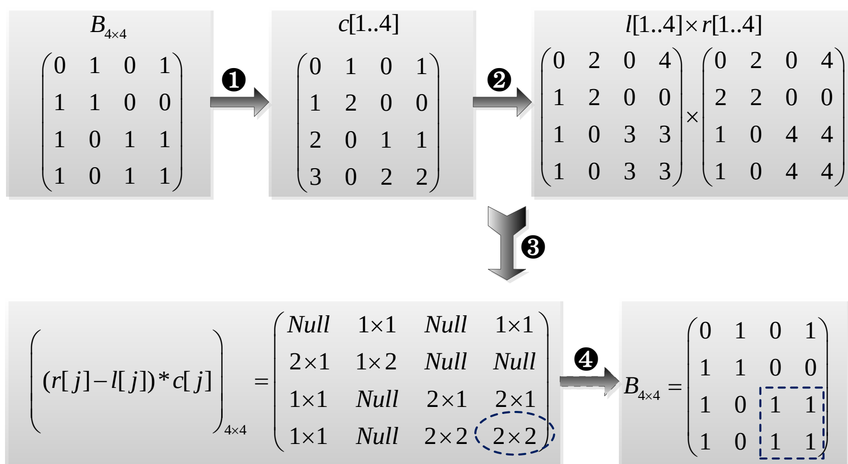


Figure 3. An example of all 1-element maximum sub-matrix solution

3.2. Indoor primitive geometry

In the literature [18], Li *et al.* proposed to consider the whole 2D indoor space as the model input, and partition it into uniform continuous grids. However, the geometrical feature of each indoor element is varied and a fixed level of granularity cannot adapt every indoor element. In this paper, we consider each indoor element as the model input, and confirm its unique partitioning way according to the geometric feature. It not only can avoid the storage and computational complexity led by uniformly fine-grained grids, but also sufficiently retain the indoor geometric features to ensure the precision of indoor distance calculation.

As the indoor element such as room and door is composed of grids, we firstly give the formalization of the indoor primitive geometry for our proposed model.

Definition 1. An indoor grid IG is defined as a quadruple:

$$IG = (Gtype, Rid, Did, Greg)$$

Where $Gtype$ indicates the grid type such as obstacle grid or non-obstacle grid or door grid; if it's a grid inside room, Rid is the room id and $Greg$ is grid range; else if it's a door grid, Did is the door id and $Greg$ is grid location. The value type of $\boldsymbol{IG}$ is $ingrid$.

We can easily determine whether two grids are **adjacent** or **disjoint** through their spatial topological relationship. Besides, if two adjacent grids were not blocked by walls or obstacles, these two grids are **connect**.

Definition 2. An indoor path IP is defined as a list of connect indoor grids:

$$IP =< g_1, g_2, \dots, g_n >$$

Where $\forall i, g_i$ and $g_{i+1} (i = 1,2, .., \mathrm{n} - 1)$ are connect. The value type of IP is $inpath$.

Definition 3. An indoor region IR is defined as a set of adjacent indoor grids:

$$IR = \{g_1, g_2, \dots, g_n\}$$

Where $\forall i, \exists j, g_i$ and $g_j (i, j = 1,2, .., n)$ are adjacent. The value type of IR is $inregion$.

The spatial relationship operations of indoor geometry primitives are (as shown in Figure 4):

$ingrid \times ingrid$	$\rightarrow bool$	**adjacent**
$ingrid \times ingrid$	$\rightarrow bool$	**connect**
$ingrid \times ingrid$	$\rightarrow bool$	**disjoint**
$inpath \times inpath$	$\rightarrow bool$	**intersect**
$inpath \times inregion$	$\rightarrow bool$	
$inregion \times inregion$	$\rightarrow bool$	**overlap**
$inpath \times ingrid$	$\rightarrow bool$	**contain**
$inregion \times ingrid$	$\rightarrow bool$	
$ingrid \times ingrid$	$\rightarrow inregion$	**union**
$inregion \times inregion$	$\rightarrow ingrids$	**intersection**

The **adjacent**, **disjoint**, **intersect**, **overlap** and **contain** are spatial topological relationship operations similar to outdoor space; the **union** and **intersection** are geometric relationship calculations of indoor grids.

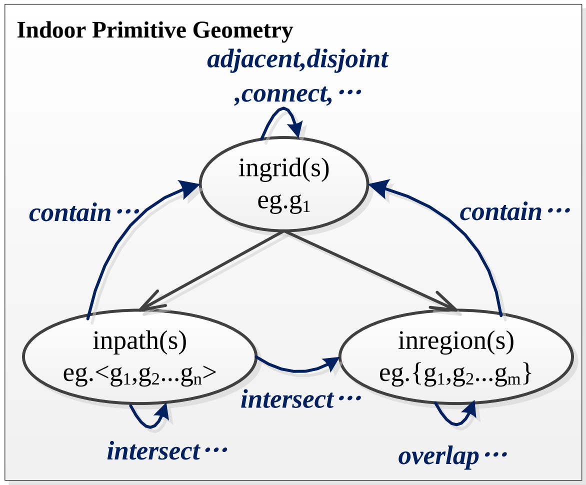


Figure 4. Relationships of indoor primitive geometry

There are also pure spatial attribute operations of indoor geometry primitives:

point	→*ingrid*	**p2g**
ingrid	→*point*	**center**
inpath	→*real*	**length**
inregion	→*real*	**area**

Given a 3D indoor point, **p2g** can relate to its resident grid; **center** return the center point of indoor grid; **length** return the length of indoor path; **area** return the area of indoor region. The center point can be utilized to calculate the indoor distance and oriented direction of two connect grids.

Definition 4. Given two indoor grids (g_1, g_2), the indoor oriented direction between them *Indir* is a triple:

$$Indir = (d_x, d_y, d_h)$$

Where d_x, d_y, d_h is the deviation between center points of g_1 and g_2. The value type of $Indir$ is $indir$.

The classic orientation description can be easily deduced from the *indir* value with the positive and negative nature of the deviation components. Thus we provide operations **getX**, **getY** and **getZ** to get deviations of indoor direction.

indir	→*float*	**getX,getY,getZ**

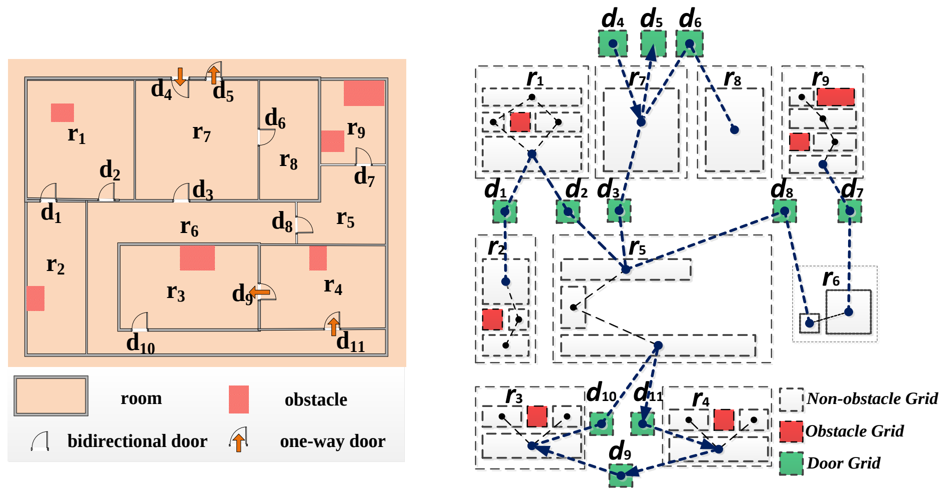
The **gindir** operation can return the indoor oriented direction between two grids; **gindirs** can return the indoor direction sequence of an indoor path.

ingrid×*ingrid*	→*indir*	**gindir**
inpath	→$(indir)^+$	**gindirs**

3.3. Modeling of indoor space

In this section, we discuss the proposed indoor multi-granularity grid graph model (InMGG_Model). Figure 5(b) illustrates the InMGG_Model of indoor space layout in Figure

5(a). Each room is partitioned into multi-granularity grids, and we maintain links between non-obstacle grids as a local connect graph. Each door is also abstracted as a virtual grid, which has only location and no size, and we also maintain its link to two grids in adjacent rooms as a global connect graph. The global connect graph is a directed graph, and the direction of its grid link indicates the access direction of the door element.



(a) Indoor space layout **(b) InMGG_Model representation**

Figure 5. An illustration of indoor space InMGG_Model representation

The InMGG_Model utilizes indoor grid as primitive geometry, and gives a complete expression of indoor connective relationship and geometric information as listed in Figure 1:

- The geometric shape information *Compose* and the geometric location information *Locate* of indoor elements and moving objects are illustrated by the geometry information of indoor partitioned grids;
- The *Connect* relationship between indoor room and door element is represented by connections between indoor geometry primitive of grid connect graph;
- The indoor distance can be fixed with indoor grid connect graph, which avoids the indoor obstacles and turning points, and gives more accurate description of indoor space distance information and indoor oriented direction.

Definition 5. The InMGG_Model representation of an indoor space is a quadruple:

$$InMGG = (GS, RS, DS, GCG)$$

Where GS is the set of indoor grids, RS is the set of room entities; DS is the set of door entities; the global connect graph GCG is a set of grid links. The value type of $InMGG$ is $inmgg$.

Definition 6. The grid link GL between two connect grids (g_1, g_2) can be defined as:

$$GL: g_1 \times g_2 \rightarrow Lwt \times Accdir$$

Which maps two connect grids to its distance weight Lwt with real type and its access direction $Accdir$ with enumeration type, which indicates one-way or bidirectional.

Definition 7. An indoor room entity $Room$ is defined as a quintuple:

$$Room = (Rid, Rname, Rkind, Rreg, LCG)$$

Where Rid is the room ident; $Rname$ and $Rkind$ is the semantic information of room with string type; the range of room $Rreg$ is an indoor region; the local connect graph LCG is a set of grid links.

Definition 8. An indoor door entity $Door$ is defined as a triple:

$$Door = (Did, Dtype, Dloc)$$

Where Did is the door ident; $Dtype$ is the semantic type of door with string type, such as the open door of the cafe bar; the location of door $Dloc$ is an indoor grid.

Given the InMGG_Model representation, we can judge whether any two grids are connect with **connect** and the minimum indoor path between them with **inroute**.

$ingrid \times ingrid \times inmgg$	$\rightarrow bool$	**connect**
$ingrid \times ingrid \times inmgg$	$\rightarrow inpath$	**inroute**
$ingrid \times ingrid \times inmgg \times inregion$	$\rightarrow inpath$	**cos_inroute**

The indoor distance between two connect grids can be easily calculated using their center points. Then we can calculate the local distance and path matrix for the local connect graph inside room with Dijsktra algorithm offline. Finally, we can fix the source and destination room, and calculate the minimum path between them with the global connect graph whose node and edge both have weights.

The **cos_inroute** operation is to calculate the minimum path inside a constrained indoor region. As the grid partition considers the geometric feature of indoor room, the distance calculation is apparently more accurate than other models.

As indoor moving object is located in indoor grid, then the trajectory of indoor moving object can be represented as a sequence of indoor grids with time stamps. Given the spatio-temporal data type[19]: $mov(\alpha) = (\text{TIME} \times \alpha)^{+} \rightarrow \tau(\alpha)$, and spatio-temporal operations such as *atinstant*, *atperiod*, *minvalue*, *maxvalue* and so on.

Definition 9. An indoor moving object IMO can be defined as a quadruple:

$$IMO = (Oid, Oname, Ospeed, Otraj)$$

Where Oid is object ident; $Oname$ is its sematic name with string type; $Ospeed$ is its speed with $mov(real)$ type; $Otraj$ is its moving trajectory with $mov(ingrid)$ type.

Generic operations for indoor moving grids are, for example:

$mov(ingrid)$	$\rightarrow inpath$	**intraj**
$mov(ingrid) \times \text{inregion}$	$\rightarrow mov(ingrid)$	**instpass**
$mov(ingrid) \times mov(ingrid)$	$\rightarrow mov(bool)$	**instmeet**
$mov(ingrid) \times mov(ingrid)$	$\rightarrow mov(real)$	**instdist**

Where **intraj** is the projection of indoor moving grid to an indoor path; **instpass** return the section of indoor moving grid when it was inside the indoor region; **instmeet** judge whether two indoor moving grids are meet; **instdist** calculate the distance between two indoor grids all the time.

4. Applications of InMGG_Model

In this section, we assume a shopping mall scenario and introduce several queries of indoor applications. It's worth noting that these queries illustrate a large range of possibilities and exemplify the potential of the proposed InMGG_Model which is not limited to a specific application domain. Firstly, we integrate the presented data types into the relational model and have relations:

```
Mall(Mid:number,Mname:string,Mbuilding:inmgg);
Stores(Rid:number,Mid:number,Rname:string,Rreg:inregion);
Doors(Did:number,Mid:number,Dloc:ingrid);
Customers(Cid:number,Cname:string,Ctraj:mov(ingrid))
```

4.1. Indoor navigation

Previous indoor navigation researches focus on how to get the optimal indoor path and neglect the output of the navigation. As InMGG_Model support the description of indoor oriented direction between indoor grids, we can return a directional navigation path for the convenience of blinded persons or robotics. We can ask a query "Tell me how to get from Nike store to KFC in CBD mall?":

```
SELECT inroute(RR1.Rreg,RR2.Rreg,MM.Mbuilding) AS PathValue,
  gindirs(inroute(RR1.Rreg,RR2.Rreg,MM.Mbuilding)) AS DirValue
FROM Mall MM,Stores RR1,Stores RR2
WHERE MM.Mname='CBD' AND RR1.Rname='Nike' AND RR2.Rname='KFC';
```

Besides directional navigation, we also can use the indoor oriented direction as filtering condition, and answer indoor queries with direction constraint such as "How to get to the nearest Cafe on the east of Adidas store in CBD mall?":

```
CREATE VIEW VDist AS
SELECT RR2.Rid AS Rid,RR2.Rname AS Rname,inroute(RR1.Rreg,
  RR2.Rreg,MM.Mbuilding) AS PathValue,length(inroute (RR1.Rreg,
  RR2.Rreg,MM.Mbuilding)) AS DistValue
FROM Mall MM,Stores RR1,Stores RR2
WHERE MM.Mname='CBD' AND RR1.Rname='Adidas' AND RR2.Rkind
  ='Cafe' AND getX (gindir(RR1.Rreg,RR2.Rreg))>0;


SELECT Rid,Rname,PathValue
FROM VDist
WHERE DistValue=(SELECT MIN(DistValue) FROM V_Dist);
```

4.2. Indoor moving object join query

Traditional business analysis usually uses the buying history of customers as the data source. The InMGG_Model offers us a novel opportunity to conduct trajectory-analysis for customers. As we can easily calculate the indoor topological relationship between grids, we can deduce the topological relationship between moving objects and find similar behavior. For example, the moving object join query "Give object pairs which meet together in McCafe of CBD mall during $[t_1, t_2]$ period", and we can process it with distance constraint (when distance is zero), or with topological constraint as below:

```
SELECT CC1.Cid AS Cid1,CC2.Cid AS Cid2
FROM Mall MM,Stores RR,Customers CC1,Customers CC2
WHERE MM.Mname='CBD' AND RR.Rname='McCafe' AND RR.Mid=MM.Mid
  AND CC1.Cid != CC2.Cid AND exists(instmeet(instpass(atperiod
  (CC1.Ctraj,[t1,t2]),RR.Rreg),instpass(atperiod(CC2.Ctraj,[t1,
  t2]),RR.Rreg)),TRUE );
```

4.3. Indoor hotspot analysis

As indoor environment is a constrained space, once there was congestion happened, it would need a long time to recover from it. Therefore, we need to analyze the human crowds behavior to get the indoor hotspot which is visited by most persons, and try to avoid it in the future path planning. Previous research of indoor hotspot analysis used to sort up the count of humans inside room. However, it did not consider the size of indoor space. Now we could refigure it with the geometry information of InMGG_Model, such as a query "Give the room with maximum density of customer flow in CBD mall at t instant":

```
CREATE VIEW VPair AS
SELECT RR.Rid AS Rid,CC.Cid AS Cid
FROM Stores RR,Customers CC
WHERE RR.Rname='CBD' AND contain(RR.Rreg, atinstanct(CC.Ctraj,
  t))='TRUE';

CREATE VIEW AS VCnt AS
SELECT VP.Rid AS Rid,COUNT(*) AS Cnt
FROM VPair VP
GROUP BY VP.Rid;

CREATE VIEW AS VDen AS
SELECT VC.Rid AS Rid, VC.Cnt/area(RR.Rreg) AS Den
FROM VCnt VC,Stores RR
WHERE VC.Rid=RR.Rid;

SELECT Rid FROM VDen
WHERE Dens=(SELECT MAX(Dens) FROM VDen);
```

4.4. Indoor diffusion analysis

Indoor diffusion analysis refers to the simulation of a physical process such as the spread of fire or poisonous gas. As indoor space is abstracted as a set of discrete symbols in traditional symbolic modelling, it cannot be able to simulate the continuous spreading process. Our proposed model can figure it out with indoor grids supporting geometric information of indoor elements.

Assume that the diffusion source was located in (x, y, h) position in CBD mall, what is the diffusion extent after k timestamps with diffusion rate v? As indoor obstacle and wall can be penetrated by diffusion, we should add some linked edges into the grid connect graph of

InMGG_Model and set the distance weight according to its penetrated time, then we can answer the query as:

```
SELECT UNION(GG.Greg)
FROM Mall MM,InGrids GG
WHERE length(inroute(p2g(point(x,y,h)),GG,MM.building)) <=v*t
  AND MM.Mname='CBD';
```

The diffusion analysis is significant in emergent situations, e.g., finding ways-out when indoor fire occurs. In order to calculate a safe way-out, we firstly need to have an estimation of the spreading range. Given the diffusion extent *r*, we can easily calculate a safe way-out for Nike store in CBD mall with **cos_inroute** operation:

```
SELECT cos_inroute(RR1.Rreg,RR2.Rreg,r) AS PathValue
FROM Mall MM,Stores RR1,Stores RR2
WHERE MM.Mname='CBD' AND RR1.Rname='Nike' AND
  RR2.Rname='Exit';
```

5. Conclusions

In this paper, we present an indoor multi-granularity grid graph model for indoor space representation and indoor moving objects. Compared with previous models, our model implements the integration of indoor geometric and topological semantics by introducing multi-granularity grids as indoor primitive geometry. We also give the formalization of data structures and data operations to represent and manipulate the entities in indoor space, which can be implemented on existing data models, such as relational, or object-oriented.

In the future, we will consider the implementation of this data model on main-stream relational databases, and provide a uniform indoor operation interface for applications. We also need to make further optimizations for data operations of this model, such as devising an indoor grid index structure for indoor path calculation.

Acknowledgements

This work is supported by the National Science Foundation of China under the grant no. 61379037 and no. 61073039.

References

[1] C. S. Jensen, H. Lu and B. Yang, Indoor-A New Data Management Frontier, IEEE Data Engineering Bulletin, vol. 33, no. 2, **(2010)**, pp. 12-17.

[2] J. Xu and R. H. Güting, "A Generic Data Model for Moving Objects", GeoInformatica, vol. 17, no. 1, **(2013)**, pp. 125-172.

[3] P. Jin, L. Yue and Y. Gong, "Semantics and Modeling of Spatiotemporal Changes", In Proc. Of CoopIS/DOA/ODBASE (**2003**), LNCS 2888, pp. 924-933.

[4] T. Kolbe, G. Goger and L. Plumer, "CityGML: Interoperable Access to 3D City Models", Proceedings of 1st International Symposium on Geo-information for Disaster Management, **(2005)**, Berlin: Springer, pp. 883-889.

[5] P. Dudas and M. Ghafourian, "ONALIN: Ontology and Algorithm for Indoor Routing", Proceedings of International Conference on Mobile Data Management **(2009)**, Taipei, Taiwan, China, pp. 720-725.

[6] A. Elfes, "Using Occupancy Grids for Mobile Robot Perception and Navigation", Computer, vol. 22, no. 6, **(1989)**, pp. 46-57.

[7] D. Demyen and M. Buro, "Efficient Triangulation-based Path Finding", Proceedings of the 21st National Conference on Artificial Intelligence **(2006)**, Boston, Massachusetts, USA, pp. 942–947.

[8] M. Mekni, "Automated Generation of Geometrically-precise and Semantically-informed Virtual Geographic Environments Populated with Spatially-reasoning Agents", Universal-Publishers, **(2010)**.
[9] J. Wallgrun, "Hierarchical Voronoi graphs: Spatial representation and reasoning for mobile robots", Springer, Heidelberg, Berlin, **(2010).**
[10] H. Kim and C. Jun, "A SDBMS-based 2D-3D Hybrid Model for Indoor Routing", Proceedings of MDM **(2009)**, Taipei, Taiwan, China, pp. 726-730.
[11] F. Penninga, "A Tetrahedronized Irregular Network Based DBMS Approach for 3D Topographic Data Modeling", Progress in Spatial Data Handling **(2006),** Berlin: Springer, pp. 581-598.
[12] C. Becker and F. Durr, "On Location Models for Ubiquitous Computing", Personal Ubiquitous Computing, vol. 9, no. 1, **(2005)**, pp. 20-31.
[13] D. Li and D. Lee, "A Topology-based Semantic Location Model for Indoor Applications", Proceedings of International Conference on Geographic Information Sciences, **(2008)**, Irvine, California, USA, pp. 1-10.
[14] D. Li and D. Lee, "A Lattice-based Semantic Location Model for Indoor Navigation", Proceedings of MDM, **(2008)**, Beijing, China, pp. 17-24.
[15] C. S. Jensen and H. Lu, "Graph Model Based Indoor Tracking", Proceedings of MDM, **(2009)**, Taipei, Taiwan, China, pp. 122-131.
[16] B. Yang and H. Lu, "Probabilistic Threshold k Nearest Neighbor Queries over Moving Objects in Symbolic Indoor Space", Proceedings of EDBT, **(2010)**, Lausanne, Switzerland, pp.335-346.
[17] P. Q. Jin and L. L. Zhang, "Semantics and Modeling of Indoor Moving Objects", International Journal of Multimedia and Ubiquitous Engineering, vol. 7, no. 2, **(2012)**, pp. 153-158.
[18] X. Li, C. Claramunt and C. Ray, "A Grid Graph-based Model for the Analysis of 2D Indoor Spaces", Computer Environment and Urban Systems, vol. 34, no. 6, **(2010)**, pp. 532-540.
[19] M. Erwig and R. H. Güting, "Spatio-temporal data types: An approach to modeling and querying moving objects in databases", GeoInformatica, vol. 3, no. 3, **(1999)**, pp. 269-296.

Authors

Na Wang

She is currently a PhD candidate in the School of Computer Science and Technology of China (USTC). Her research interests are spatiotemporal databases and flash-based databases.

Peiquan Jin

He received his PhD. Degree in computer science from Univer sity of Science and Technology of China (USTC) in 2003. After that he spent two years for his post-doc research in the Department of Electronic Engineering & Information Science, USTC. He is now an associate professor in the School of Computer Science and Technology, USTC. He was a visiting scientist in University of Kaiserslautern in 2009. Currently, he is a senior member of CCF, and a member of IEEE and ACM. In recent years, his research interests focused on spatiotemporal databases, flash-based databases, and Web information retrieval. He has published more than 50 papers both in international journal and conferences.

Yan Xiong

He is currently a full professor in School of Computer Science and Technology, University of Science and Technology of China. His research interests include distributed processing, mobile computation, and information security. He has published over 40 papers in journals and conferences.

Lihua Yue

He has been a full professor in the School of Computer Science and Technology at University of Science and Technology of China (USTC) since 2001. She received her B.S. and M.S. degree in computer science both from USTC. Her research interests include flash-based databases, spatiotemporal databases, information retrieval, and image processing. She has published more than 60 papers in refereed international journals and conferences, such as DEXA, CIKM, MMM and IEEE Transaction on Consumer Electronics. She is a committee of Database Society of the China Computer Federation (CCF DBS) and served as a PC member of many conferences.

