Spatial Models for Indoor Context-Aware Navigation Systems: A Survey

Imad Afyouni, Cyril Ray, Christophe Claramunt

June 10, 2011

Abstract

This paper surveys the multitude of indoor spatial models developed for many fields of study ranging from mobile robot mapping to indoor location-based services (LBS) and most recently to context-aware navigation services applied to indoor environments. Recently, several studies have been presented to evaluate spatial models in the context of mobile robot localisation and mapping [12], [74] as well as the one of ubiquitous computing [7]. This paper provides an extended analysis of indoor space modelling for context-aware navigation systems. The main criteria to be considered in the design of indoor spatial models are firstly introduced from a context-aware systems perspective. A taxonomy of models is then presented and assessed based on these criteria with the aim of providing a flexible spatial data structure, thus strengthening the entire context-aware system being developed.

1 Introduction

The notion of context has been recently recognised as a key element in the development of context-aware information systems. Context varies according to application constraints, but also integrates the way users act in the environment as well as the interfaces they use to interact. So far, no clear boundaries have been defined to classify context dimensions [59]. Nonetheless, two generic concepts have been introduced as super-classes that encapsulate these dimensions. The *context of use* comprises all user-centred dimensions of context which mainly include: (1) the user space (i.e., the user profile and its physical/cognitive capacity), (2) the user platform that provides direct interaction with the system, and (3) the environmental context which refers to the parameters that influence the user (e.g., location, temperature, light, etc.). The *context of execution* models the dynamic behaviour of an information system and encompasses two main dimensions: (1) the infrastructure dimension that detects changes in the topological organization of system components and (2) the system dimension that measures resource utilization (e.g., memory, processor, and network) (cf. [59] for more details).
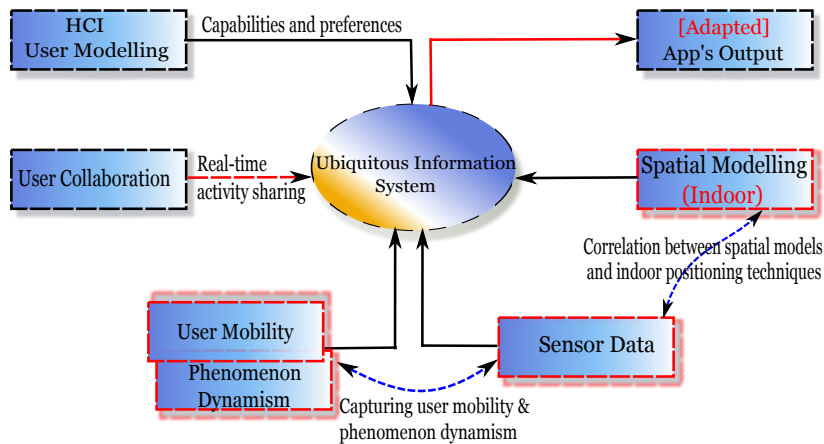
Figure 1: Key elements for the design of ubiquitous information systems

Emerging and continuing advances in ubiquitous systems and localisation techniques have brought novel opportunities to develop context-aware indoor navigation services. Diverse kinds of services can be provided by enabling real-time integration of the context dimensions into the query processing model. Examples of such services include: human way-finding and navigation in built environments, evacuation routes for people stuck in a building in case of an emergency, real-time collaborative activities, etc. (see section 2 for an extended review).

It follows that the location information is an unavoidable context dimension to be considered. Whether the location information should be handled as any other context information, or be managed differently is a key issue to address. In fact, a better understanding of the location information and the relationships that might exist among spatial entities will enrich context-awareness. Hence, spatial modelling appears as a key element that contributes to the design of context-aware information systems (Fig. 1). Much work has been done on user modelling for developing adaptable systems [30]. Other works have addressed the problem of information gathering from different kinds of deployed sensors [15], and other developed systems that support collaborative activities [18]. However, these systems still require the development of appropriate spatial data structures and data manipulation facilities [7].

Amongst many applications, context-aware indoor navigation services strongly rely on location information to adapt to context changes. A better modelling integration of indoor spaces and context-aware systems still requires the development of dynamic and flexible spatial models that will help to provide appropriate services to mobile users acting in the environment. We believe that a context-dependent spatial model should represent

(i) all features that populate the environment which include rigid entities (e.g., mobile users, sensors, static objects) as well as continuous phenomena (e.g., noise and fire processes), (ii) their spatial properties, and (iii) the behaviours that emerge from them (i.e., how objects can communicate physically interact within the environment).

The aim of this research is to study context-aware services and queries in indoor environments, and particularly: (1) to reconsider current indoor spatial models from a context-aware systems perspective so that a flexible and well-suited model can thereafter be designed, (2) to take into consideration dynamic properties and activity-oriented interactions of moving objects with their physical surroundings, (3) to examine the role played by smart devices and ubiquitous positioning sensors. Once these primary issues are addressed and efficiently integrated into the system, a large range of context-aware services and queries can be developed. Such a model should also support human activities aimed at achieving some objectives. An activity can be defined as a collection of goal-oriented actions an entity can perform [79], [39]. Actions comprise a sequence of location movements, interactions with other neighbouring entities and artefacts, and requests for some services in order to achieve a predefined goal. An activity can also be made of a set of primitive activities or be part of a larger collective activity. Medical activities are such complex activities [5].

In this paper, we provide an extended analysis of different spatial models by discussing context-awareness to further enrich the interaction and communication of the user within the indoor space. The remainder of the paper is organized as follows: section 2 provides a review of typical location-aware services. Section 3 introduces the main criteria to be considered in the design of indoor spatial models. Section 4 describes and evaluates different spatial models based on these criteria.

2 Location-aware, navigation-related services applied to indoor spaces

A wide range of location-aware services can be applied to indoor environments by enabling a continuously accurate localization technique. A main goal of all these services is to provide the user with the ability to interact with his or her physical surroundings in order to achieve some objectives.

Location-aware, user-centred services can generally be distinguished according to data access modes. Indeed, two modes of data access can be discerned, *pull* mode and *push* mode.

- Pull-based location-aware services comprise all user-initiated requests with the aim of pulling some location-dependent information from the service provider.

- Push-based services are initiated by the service provider without having been requested by the user. The service provider uses the location information of consenting or subscribed users to trigger alerts or contextual advertising and push them to the users' devices.

Location-dependant queries are typical examples of pull-based services [81], [32], and constitute a fundamental group of services needed in any context-aware navigation system. Queries are location-dependent, meaning that any change over time of the locations of objects that are involved in the query may significantly affect the answer. If, for example, a user wants to find out his or her friends within 100 meters while navigating a shopping centre, this answer will depend on the user's current position as well as on the location of the nearest friends. Many different types of queries can be derived from this category; we put our main focus on types of queries that are navigation-related (an extended survey on location-based queries has been presented in [32]).

- *Navigation queries* encompass all queries that directly help users by providing them with navigational information. Examples of such queries are: (i) asking for some direction or location information, (ii) discovering the *optimal* path to the nearest points of interests (e.g., place, feature, etc.), (iii) planning a path to the destination, (iv) retrieving information about objects of interests, etc.

- *Range queries* are used to find and retrieve information about objects or places within a user-specified range or area [78], [80]. They provide an important navigation support by continuously updating relevant details according to users' movements. Ranges may be characterized by a circular or rectangular-shaped window in which objects of interest are captured. In addition, range queries may be static or dynamic according to whether or not the initial query point is in a static position. Similarly, a range query can be applied on static or dynamic data depending on whether target objects are moving or not. [80] has proposed an approach to support range queries in indoor environments by constructing a virtual cell-based network used to expand segments from the initial point until objects that meet the specified conditions are found and stored for the final answer.

- *(k) nearest neighbour (NN) queries* search for the (k) closest qualified objects to the moving user with respect to his or her current position [70], [81].

The efficient management of static and dynamic data is a key issue for processing these kinds of continuous queries since the result of a query is only valid for particular locations of the involved objects. Therefore, continuously updating queries implies a communication overhead and additional

processing cost at the server side. A concept of validity region was proposed by [81] to lessen this problem by determining a safe area around the initial user position in which the result of the query is always valid. Many variants of these queries are summarized in [32].

[40], [58], [46] have reviewed numerous other location-based, real-time services that deliver rich information and allow for communication. For instance, location-based instant messaging in which outgoing messages are not just stamped by the local time-zone of the sender but also by the sender's current location. In the same manner, a user can instead manually specify a certain location specification so that other users whose specifications meet those given by the sender will receive the message. Other kinds of services are based on *geocast* messaging [58], [6], and include location-based advertisements and alerts. Geocast messaging can be described as a location-based multicast where messages are delivered to users located in a specific area instead of those subscribed to a given group.

In particular, location-based alerts are not necessarily time-related and are generally used in case of an emergency to warn people to go around the dangerous zone. It might also be useful, for example, to remind a user navigating a shopping centre to buy some food or fruit stock when he or she is located next to a supermarket. On the other hand, location-based advertisements [1] (also called proximity-triggered advertisements) generally target nearby consumers to provide them with information about stores' offers, discount coupons, etc. This kind of push-based services should provide users with a subscription-based mechanism that allows for relevant and non-intrusive advertising.

Research studies on location-based queries and services have hitherto discussed and barely dealt with the location information as the only dimension of context. They have almost ignored other dimensions such as *user context* (e.g., profile, preferences, disabilities, etc.) and *environmental context* (e.g., time, temperature, statistics about data sources, etc.) that may completely affect the answer to a query even if the locations of the query issuer and other involved objects have not been changed. [54] has recently highlighted these issues and discussed many challenges to be considered in order to carry out practical context-aware queries and services. Some of these challenges include (i) designing a user model, (ii) supporting multi-objective and perhaps contradictory queries, and (iii) analysing efficiency and scalability when dealing with context-aware continuous queries.

3   Criteria for a well-suited context-dependent spatial model

As mentioned earlier, context-aware navigation systems require the use and integration of an appropriate indoor spatial model that meets the requirements and satisfies application and structural constraints. Indoor spatial

models have been studied and developed in many areas ranging from mobile robot mapping to GIS to context-aware computing. In this section, we propose to classify the main criteria that characterize a spatial model into two categories: service-oriented criteria and design-related criteria. The first group includes criteria for real-time services, namely, (i) *localisation*, (ii) *navigation*, (iii) *location-aware communication*, and (iv) *activity-oriented interaction*, and a criterion for delayed services named (v) *spatial analysis*. The second group examines performance-related issues and contains three criteria: (i) *construction and maintenance*, (ii) *efficiency and scalability*, and (iii) *flexibility*. We believe that these are generic and application-independent requirements which will later be used in this paper to evaluate existing approaches and then to design our own model.

3.1 Service-oriented criteria

A context-dependent spatial model should firstly be able to locate objects of interest accurately and with additional semantic description so that other more advanced services can be provided. It also needs to support navigational services a user is likely to ask. Communication between entities based on their locations needs further to be supplied. In addition, a user navigating an indoor space should be able to interact with his or her physical surroundings. Other deferred services can also be offered by the model thus enabling better understanding of usersŠ behaviours. These requirements are thoroughly discussed in this section.

3.1.1 Localisation

The process of determining the approximate position of a mobile object is called *"localisation"*. Recently, intense scientific studies have been performed in order to design and build positioning systems in GPS-less indoor environments. Indeed, indoor spaces are constrained by the architectural components such as doors, floors, corridors, and walls. Thus, the task of achieving accurate localisation is rather difficult, despite its practical relevance in some situations when designing context-aware applications.

The location information delivered by a positioning system may be in different formats according to two concepts of representation:

Geometric information that gives quantitative description about moving objects in the form of tuples composed of either cartesian or latitude-longitude-altitude coordinates. These coordinates are always delivered according to a given reference coordinate system. In particular, GPS is specially an absolute positioning system used mostly outdoors and provides geometric coordinates with respect to the WSG84, that is, the global reference system used to determine position in any place on the earth. Whereas,

the Active Bat system [76] and MIT Cricket [60] provide absolute positions with respect to local reference coordinate systems developed based on a set of local landmarks (e.g., beacons) that are distributed in the indoor space. Spatial relationships like *"overlap"*, *"inside"*, *"intersect"*, and *"disjoint"* can be inferred by determining the distance between two geometrically located entities. Hence, it enables the interpretation of a set of coordinates and the deduction of relative spatial relations between entities.

Symbolic information that provides qualitative human-readable descriptions about moving objects based on symbolic points of interest (e.g., room or floor identifier, building name, etc.). For instance, the Active Badge system [75] handles coordinates that represent the symbolic identifiers of the fixed IR sensors used for positioning. In contrast to geometric coordinates, symbolic coordinates allow to explicitly model topological relations (e.g., spatial containment and spatial connectedness) between entities in the environment. This can be done by means of symbolic spatial models such as set-based or graph-based models (details about the different spatial representations will be given in the next section). Consequently, symbolic coordinates enable spatial and semantic reasoning at an abstract level, thus favouring interaction between neighbourhoods and within the space.

Nowadays, hybrid positioning approaches appear as promising solutions providing reliable, continuous, and accurate location information [63]. These approaches discuss the fact that location information is often acquired by different sensor types. Therefore, an appropriate multi-sensor data fusion technique, and potentially an algorithm for map matching, needs to be used in the filtering process. Consequently, the spatial model should be capable of representing the approved coordinate system by which the location information is expressed and, when necessary, transforming that location information in order to get all sensor data in a common format.

3.1.2   Context-aware navigation

The art of navigation through the environment could be defined as a scheduled and goal-oriented movement made by humans or robots [55]. While navigating, a user may be technically assisted by sensory devices embedded in the environment that provide relevant information to maintain orientation, and other suggestions to encourage him or her to interact within the environment.

As an extension of the traditional navigation tasks described in section [2], context-awareness needs to be integrated thus offering other opportunities well-suited to our research goal. A context-aware navigation task is carried out in two phases [13]: the static phase, which is generally known as *"path planning"*, encompasses a multi-criteria path selection process that

generates an unbroken path from the current location to the destination. This process is context-sensitive as it aggregates multiple criteria (e.g., user preferences and capabilities, distance, time, etc.) passed as function parameters to evaluate the cost value of each step and then to adopt the most appropriate and the safest path, that is the *optimal* path, that allows, for instance, to reach destination while avoiding threats (e.g., gas leak). The dynamic phase lies in an effective dynamic framework needed to monitor the user's progression in order to avoid deviations from the planned path. This framework comprises a next-step selection algorithm that keeps continuous track of the user's position and tries to recover from deviations by providing additional information. Moreover, the framework should adapt the predetermined path if it detects any significant event, such as gas leak or fire diffusion, that may affect the user's movement. In such a situation, a path to the nearest emergency exit should be recomputed.

3.1.3 Location-aware communication

In contrast to telecommunications which enable communication between computing devices based on IP or MAC addresses, communication among objects in smart environments is achieved based mainly on their location. Communication models in distributed systems can be classified according to who initiates information exchange and how information is then forwarded to the specified receiver [3]. So, we can distinguish between direct (i.e., point-to-point) and indirect communication. As an example of indirect communication, a provider can disseminate information about events to invoke *remote callback methods* from potential subscribers. Similarly, a consumer can request information from a known (i.e., direct message exchange) or anonymous (i.e., indirect message exchange) provider. Moreover, collaborative activities can take place between communicating objects working together to achieve common goals.

As an illustration, the author in [65] has introduced a dynamic, symbolic location model that supports location-aware communication among rigid entities (i.e., sensors and users) in smart spaces. Rigid entities and spaces are represented as components in a hierarchy of symbolic labels (i.e., user-room-floor-building hierarchy) based on the containment relationship. Each component in the hierarchy can act as a service provider, a service consumer or both. An example of the dynamicity of the model might be the case where users navigate between subspaces (e.g., rooms). In this case, a component migration mechanism is used to update the hierarchical structure of components and thus keeps track of users' movements. Two types of communication are then established between these components depending on their location: *vertical communication* that supports the publish/subscribe approach; this means that events are published by parent components to address their children. Child components that are interested in those events

need to be subscribed so that they can receive runtime notifications. *Horizontal communication* encompasses communication between same-level components of the hierarchy (i.e., components that belong to the same parent in the hierarchy).

3.1.4 Activity-oriented interaction

Activity-based computing is an important paradigm of ubiquitous computing that aims at allowing humans to create personal and collaborative activities to provide them with deeper understanding of the surrounding space [18]. Activity theory has focussed attention on the usefulness of spatial models in the design process to enrich consciousness and interaction within the space [57], [38]. It hence becomes extremely important to embed spatial representations that serve as mediator to relay relevant information to humans, which may influence outcomes and human behaviours in the environment. Indeed, the number of spatial artefacts has increased and been distributed into ubiquitous environments. This allows users for both collecting knowledge and understanding physical surroundings, and furthermore for reconfiguring and manipulating spatial artefacts to anticipate or produce changes in the environment. Consequently, embodied interaction defined by [21] can take place by means of spatial reconfigurations that may affect existing activities and initiate new ones.

3.1.5 Spatial analysis

In *a-posteriori* simulations, an essential issue lies in identifying an appropriate spatial representation with respect to the phenomenon or behaviour being explored. So far, the analysis of spatial configurations has been studied to help measure human-environment interaction. The granularity parameter is a mandatory criterion that enables the exploration and analysis of spatial data. A fine granularity can certainly reflect the indoor space delicately, and is especially needed when simulating a physical process such as the diffusion of liquid or gaseous materials, or when considering human mobility impacts to get accurate understanding of the environment.

In contrast, topological analyses often require a coarser granularity to support adjacency and connectivity relations between spatial units (e.g., room), and thus deriving a more abstract view of the topology of a complex indoor space. Several typical scenarios are introduced in [51]. These scenarios present some application-dependent constraints, so each of them needs an appropriate level of granularity. Some examples of scenarios applied on a fine-grained spatial model are: first, the route analysis scenario that aims at finding the shortest path between two given locations or all pairs shortest paths applied on a given floor architecture and with a given

data structure. Second, the diffusion analysis defined as a dynamic process where the spatio-temporal evolution and extent of an object or phenomenon within an indoor space are explored. The third scenario involves the usage of centrality measures to characterize the architecture and the spatial distribution of rooms in a built environment (details about these scenarios can be found elsewhere [51]).

3.2 Design-related criteria

In this group, performance-related criteria are presented with the aim of providing an efficient and scalable system design. The spatial model underneath should be flexible so that a wide range of applications can be applied. An offline requirement regarding the modelling effort also needs to be considered.

3.2.1 Efficiency and scalability

A context-aware indoor navigation system requires to efficiently execute users' queries such as shortest paths between two given locations, k-nearest neighbour objects or all accessible locations starting from their current position [52]. The time complexity of query processing is given by the amount of CPU time spent per sampling unit. In addition, when considering contextual information (e.g., human factor, time, etc.), efficiency is no longer restricted to simply calculating the geometric distances, but some other functionalities are required leading to a significant increase in workload. Therefore, spatial data models should be optimized to cover a wide range of operations with maximum efficiency in storage capacity, time, cost, etc.

Furthermore, dynamic updates of the spatial data structure stored on mobile devices should be periodically performed in order to maintain the accuracy and quality of the stored data. For instance, one challenge facing spatial model designers is the cost of updating the location of every moving object continuously in the database. Existing approaches try to overcome this problem by performing periodic, deferred (i.e., demand-based) or immediate (i.e., event-driven) update queries to keep an incremental view maintenance of the database (see [33] for more details). Another approach is presented in [37], which focuses on real-time map updates on mobile devices. This proposal aimed at providing a compromise solution between two conflicting constraints: the cost of communication between the server and mobile devices, and the limited hardware capacity of these devices. This framework considers the use of a spatial model with different levels of detail. Hence, it first handles updates at the basic level on the server side, and then manages the update propagation process through the other description levels of the spatial model. Further, it transfers update messages that contain updated spatial objects to subscribed clients thus replacing original

objects with matching identifiers.

The scalability requirement means the ability of a spatial model to scale up to a large indoor environment while keeping tolerable resource consumption, which mainly affects the memory storage capacity and the behaviour of processing workloads. In context-aware applications, heterogeneous mobile devices as well as an enormous number of embedded sensors will have to be handled in an efficient and scalable way. In this situation, there is a need to establish a mechanism that eliminates superfluous information to reduce processing and communication costs, so that the system can handle a higher number of queries in real-time.

3.2.2 Flexibility

A flexible spatial model should support a wide range of applications that can be developed at different levels of abstraction, as well as different kinds of positioning sensors that might coexist to achieve better accuracy. In brief, may a simple unmixed spatial model assist and deal with a very diverse intelligent navigation tasks efficiently?

3.2.3 Design and maintenance

Modelling effort is another aspect to consider when designing spatial models. This is generally expressed by how much space and time it takes to construct the model from floor plan data. Additionally, some authors [68] have discussed methods for automated construction that minimize manual intervention by designers during the modelling phase.

4 A taxonomy of indoor spatial models

An indoor space can be informally defined as a built environment where people usually behave [50] (e.g., houses, commercial malls). We believe that spatial modelling is a mandatory element for the delivery of navigation-oriented context-aware systems applied to indoor spaces. Indeed, spatial-awareness requires an appropriate data structure that is capable of representing the locations of objects either situated or actively acting within the environment.

Regarding the modelling approaches, two main classes can be inferred: symbolic and geometric spatial models. As mentioned in the previous section, both geometric and symbolic spatial models are used by positioning systems depending on their nature. So, geometric-based approaches (otherwise referred to as metric or coordinate-based approaches) consider that locations are represented as points, lines, areas, or volumes. Each object is then quantitatively described with either cartesian or latitude-longitude-altitude coordinates within a given coordinate system. Symbolic-based approaches pro-
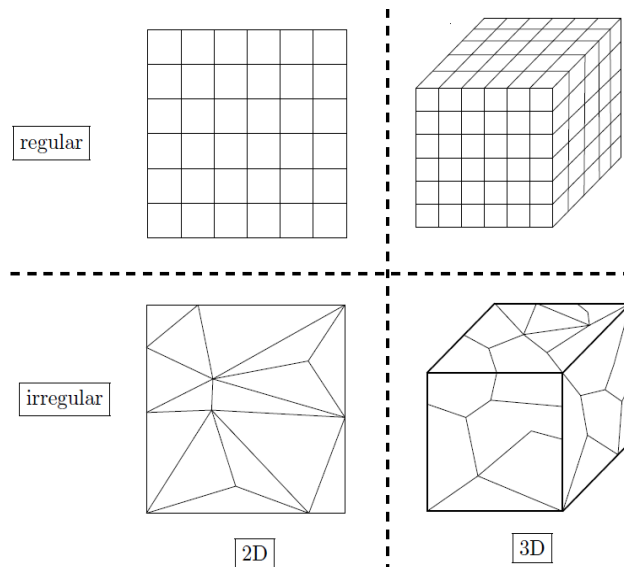
Figure 2: Examples of regular and irregular tessellations in two and three dimensions

vide qualitative human-readable descriptions about objects based on symbolic points of interest (e.g., room or floor identifier, building name, etc.).

In the following section, we shall assess the strengths and weaknesses of both classes of models by describing and evaluating different spatial representations while having in mind the above-mentioned criteria which are necessary for strengthening the spatial data structure and thus the entire context-aware system being developed.

4.1 Geometric-based approaches

Geometric spatial models mainly comprise cell-based and boundary-based geometrical representations. They can efficiently integrate metric properties i.e., the distance between two points or the angle between two vectors. This allows for specifying location and direction information precisely and for computing distance accurately.

4.1.1 Cell-based models

The *cell-based* approach tessellates the physical space into a finite number of non-overlapping regions thus building a partition that covers the entire space. This approach provides an explicit modelling ability to capture adjacencies between neighbouring cells. Two main types of tessellations can be distinguished [66], [44], [53]: regular tessellations decompose the space into
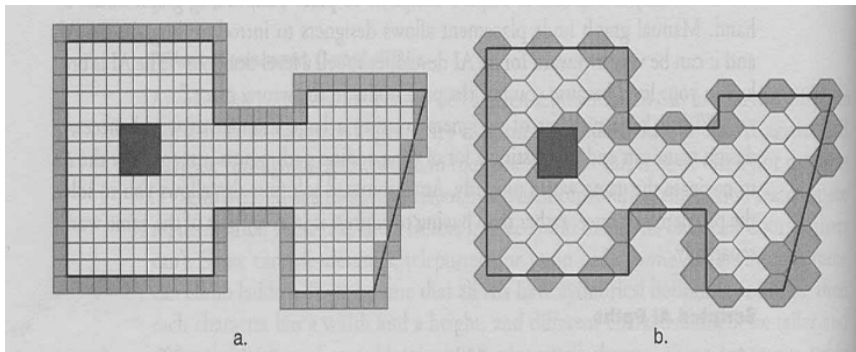
Figure 3: Grid representations based on (a) square and (b) hexagonal cells

cells that have the exact same shape and size (e.g., primarily, square- and hexagonal-shaped cells (Fig. 3)). Irregular tessellations aim at providing adaptive decomposition of the space that is suitable to exactly represent the complexity of the environment being studied (e.g., to accurately represent obstacles). Cells forming the irregular partition of the space can be with different shapes and sizes (i.e., arbitrary polygons in 2D or polyhedra in 3D).

4.1.1.1   Regular tessellations

The most successful regular tessellation is unequivocally the grid-based models, where cells are square-shaped (also referred to as pixels) in two dimensions and cube-shaped (otherwise known as voxels) in three dimensional spaces. Indeed, these techniques are trivially implemented in a computer and can simply represent navigable and impassable regions in the space by associating different cell states (e.g., cells can be either free or occupied in binary-grid models). More particularly, the well-known *occupancy-grid* approach provides a probabilistic-based regular tessellation representation, which was firstly introduced by Elfes and Moravec [56], [23], to address mobile robot mapping issues. Occupancy grids offer simple yet valuable techniques for enhancing robot spatial awareness. In an occupancy grid, a high probability value is assigned to cells within accessible space, while a low probability is assigned to cells occupied partially or completely by objects or obstacles.

Grid-based approaches are capable of generating spatial models with plentiful geometric details. The extent and the level of granularity are two mandatory parameters that have to be determined a priori for the derivation of the grid. So, the accuracy of the resulting grid will depend on the cell size. Hence, a first trade-off arises between preserving high accuracy information retrieval and its significant impact on memory and time consumption

(a) Real space with arbitrary obstacles

(b) Grid-based representation

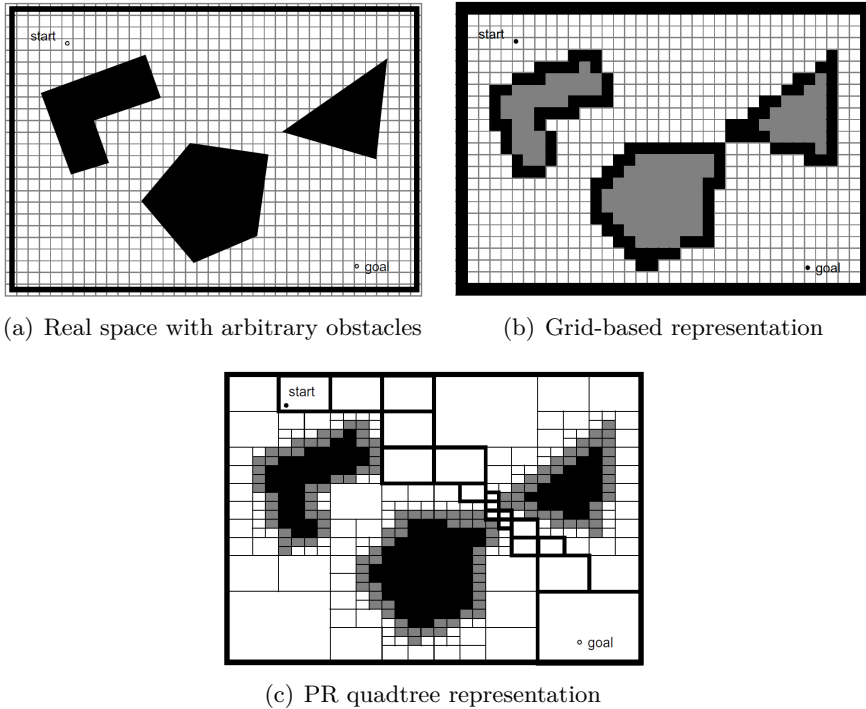(c) PR quadtree representation

Figure 4: Classic (b) and hierarchical grid representations (c) [66]

specially when dealing with large environments.

Obviously, dealing with a huge number of cells may exponentially increase processing time of fundamental queries (e.g., shortest path queries or real-time updates of the date structure), thus leading to performance and scalability problems. In addition, regular tessellation techniques do not represent objects with arbitrary shapes precisely within the space (e.g., polygonal obstacles). Hence, object boundaries will be toothed, and it is possible for narrow pathways to be missed in the modelling process especially in space areas burdened by spatial objects.

The aforementioned inefficiency has motivated the use of alternative hierarchically-organized grid-based structures known as *quadtree* for two dimensional spaces. A quadtree is a tree structure where each node has exactly four child nodes, if any. Different variants of these data structures have been introduced and are surveyed by Hanan Samet [64]. So far, the *region quadtree* (also referred to as *PR quadtree*) structure is commonly used for spatial modelling and is based on a set of points or more generally on a set of spatial objects to recursively tessellate the space into four equally-sized squares while trying to merge all free regions and to split other partially occupied regions. The algorithm breaks off when a sufficiently precise resolution is reached. With a limited number of objects, this technique can be
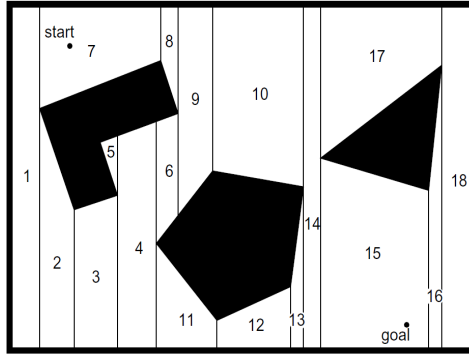
14

Figure 5: Example of trapezoidal tessellation [66]

gainful since it allows saving space when few details are required. This also allows recovering missed pathways by repeatedly creating smaller squares to capture more details when necessary. The key disadvantage of this approach lies in its lack of flexibility especially when dealing with a highly dynamic environment. Indeed, dynamic environments encompass moving users, sensors, and immutable and transient obstacles that can be moved or distributed diversely in the space. This means that whenever the distribution of these objects changes, a significant update may affect the whole quadtree in real time. Moreover, localisation in free areas where the cell-size can be excessively large is significantly disturbed and may not be sufficiently accurate for navigational purposes.

4.1.1.2 Irregular tessellations

Given a set of polygonal obstacles described as line segments, two main techniques have been proposed to irregularly tessellate the space: *free-space tessellations* that exactly represent obstacles in the space and decompose free areas into convex polygonal cells (e.g., triangles, trapezoids) [20], [53]. *Voronoi tessellation* subdivides the space into a set of unconventional cells called *"voronoi cells"* that are the locus of points closer to a given convex object than any other objects [4], [17]. Irregular tessellation techniques offer several advantages over regular techniques including smooth modelling of arbitrary objects, recognition of narrow valid paths, and the fact that it can be markedly compact in unburdened environments. However, most of these techniques have shown no interest in keeping highly accurate positions of mobile objects within the cells.

Free-space tessellations Two kinds of decompositions shall be described in this category: *trapezoidal-* and *triangulation-based* tessellations. Both models are constructed based on the endpoints of the line segments

15

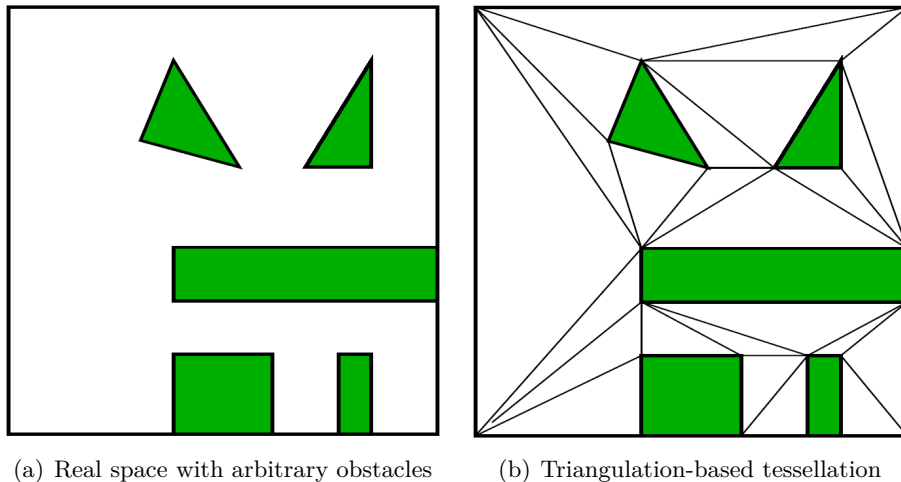(a) Real space with arbitrary obstacles (b) Triangulation-based tessellation

Figure 6: Classic triangulation tessellation technique [20]

that compose the boundaries of the obstacles. In addition, navigation is carried out by finding the intermediate cells to find the destination. Trapezoidal decomposition is built by projecting a vertical line from each end-point trough the free space until it hits another barrier, thus forming trapezoidal cells of different thicknesses. As shown in figure 5, the whole space can be represented with a little number of cells thus providing a light model. However, resulting cells tend to be long and thick, which is not suitable for localization.

On the other hand, many methods have been developed to triangulate the space. Triangulation-based tessellation can be generated by injecting edges between boundaries' endpoints without any edge crossings until no more edges can be inserted (Fig. 6). In such a model, edges that represent the boundaries are referred to as *constrained* edges because they are simply impenetrable borders, whereas all other injected edges are called *unconstrained* edges [20]. This allows for effective path finding by hopping between triangles. Still, this technique may result in very thin triangles.

Another technique known as *Delaunay triangulation* has tried to overcome the shortcomings of the last mentioned technique. A Delaunay triangulation (**DT**) of the set of endpoints is a set of triangles that compose the space such that no endpoint is inside the *circumscribed circle* (i.e., so the circle which passes through exactly three endpoints and does not contain any other endpoint) of any triangle of the **DT** (Fig. 7). It is worth noting that Delaunay triangulation may also be derived directly from the voronoi diagram described later. However, these two methods suffer from suboptimal or multiple paths between a source and a destination inside the chosen channel.
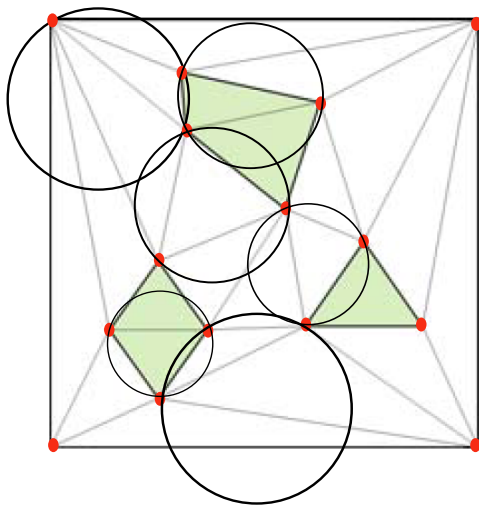
16

Figure 7: Delaunay triangulation

Voronoi tessellation The *Voronoi Diagram $\mathcal{VD}$* of a set of k points S (also known as sites) arbitrary given in $\mathbb{R}^n$ is another attractive approach that represents the built environment as a network of one-dimensional curves which concisely exhibit pathways across these points thus forming an irregular tessellation of the space [4]. Voronoi cells are generated by recursively applying a function that computes the distance between each point in the space and the selected point $p \in S$. Therefore, a voronoi cell C_p of a point p comprises the set of points $x \in \mathbb{R}^n$ that are closer to p than any other point in S. More formally defined in [45],
$C_p = \{x \in \mathbb{R}^n \mid \|x - p\| \leq \|x - q\|, \forall q \in S\}$.

For two points p and q in the plane, we denote the bisector of p and q as the perpendicular bisector of the segment pq. This bisector splits the plane into two half-planes. Thus, the voronoi diagram is observed as the intersection of $k - 1$ half-planes providing k convex bounded/unbounded cells. Each point on the outer boundary of a cell (also referred to as *voronoi edge*) is equidistant from exactly n points in $\mathbb{R}^n$, and each meet point (also called *voronoi vertex*) is equidistant from $n + 1$ points. For instance, each meet point in $\mathbb{R}^2$ is the centre of a circle referred to as the maximal inscribed circle which passes through three sites but encloses no site (see [44] for more details). The simplest way to compute the voronoi diagram in $\mathbb{R}^2$ is as follows: for each site, compute the common intersection of the other half-planes which takes $O(k \log k)$ time per voronoi cell, and $O(k^2 \log k)$ for the whole diagram. Fortune's algorithm [26] is an optimal algorithm and computes the VD in $O(k \log k)$ time.

The voronoi diagram can also be applied to convex objects such as line segments or obstacles rather than points. This extended approach, called the *Generalised Voronoi Diagram* ($\mathcal{GVD}$), uses a function to measure the distance from a point in the space to the closest point on the geometric object. Choset et al. [17], [16] have formally discussed how to incrementally construct the GVD by operating with line-of-sight sensors so that they can derive the *Generalised Voronoi Graph* (GVG) used for motion planning and described in section [4.2]. In this approach, the authors consider an indoor space IS as the union of bounded free-areas FS and a set of situated obstacles $O = O_1, \ldots, O_n$; that is, $IS = FS \bigcup (\cup_{i=1 \ldots n} O_i)$.

Two types of distance functions have been introduced: one measures the minimum distance between a point x and a convex obstacle O_i by identifying the nearest point to x in O_i; that is, $d_i(x) = \min_{y \in O_i} \|x - y\|$. The other computes the distance between a point x and a line-of-sight detectable obstacle O_i; it returns $d_i(x)$ if O_i is line-of-sight detectable and infinity ∞ otherwise.

This leads to an extended definition of the so-called *generalised voronoi cell* S_i which can be viewed as a surface comprising the set of points closest to an obstacle O_i; that is: $S_i = \{x \in FS \mid d_i(x) \leq d_j(x) \; \forall i \neq j\}$. The set of points equidistant to two obstacles O_i, O_j constructs an outer boundary of the cells S_i, S_j containing the obstacles, and is defined as: $S_{ij} = \{x \in IS \setminus (S_i \cup S_j) \mid d_i(x) - d_j(x) = 0\}$.

When more than two obstacles intervened, the resulting line segment shall be the intersection between all involved boundaries. For instance, S_{ijk} refers to an outer boundary that is made of the set of points equidistant to three obstacles O_i, O_j, O_k such that: $S_{ijk} = S_{ij} \cap S_{jk} \cap S_{ik}$. The union of all these outer boundaries generates the generalised voronoi diagram (See [4], [17] for a detailed description of the approach).

GVD represents a fundamental data structure for spatial modelling and have been widely used in many research areas such as robotic motion planning, computational geometry, computer graphics, GIS, etc. With a restricted number of objects, Voronoi tessellation as other irregular tessellation approaches can be more compact than grid-based approaches. However, the construction of the GVD is rather sophisticated and thus takes considerable time. Additionally, localisation is often mentioned as a real problem since voronoi diagrams force mobile users' positions to be shifted along voronoi edges which may yield to suboptimal trajectories. As a result, GVDs cannot directly fulfil our requirements with respect to context-aware navigational services, but lend themselves perfectly to extract the GVGs that, per se, can be used as operational route-based graphs suitable for many uses.

Model	Localisation	Navigation	Communication	Interaction	Spatial analysis	Modelling effort	Efficiency	Flexibility
Grid-based [56], [23]	geometric, regular tessellation, accurate location information	no symbolic instructions, narrow pathways may be missed	yes	require semantic annotations attached to points of interest	accurate analysis	easy to build and maintain	consumes high memory and processor time, poor in large space	no
Quadtree [64]	geometric, poor in free space	no symbolic instructions, could result in suboptimal path, no missed pathways	yes	require semantic annotations	cannot provide accurate analysis in free space	Medium	Efficient with a limited number of objects, poor in highly dynamic environments	not flexible in dynamic environments
Free-space tessellation [20], [53]	geometric, irregular tessellation, not suitable for localization	suffers from sub-optimal or multiple path	basic types of communication	objects are represented exactly but no interaction with them	No accurate analysis	Medium	only basic operations	no
Generalised voronoi diagram [4], [17]	geometric, irregular tessellation, Not very accurate	suboptimal trajectories	basic types of communication	Objects are detected within cells, no semantic annotations	No accurate analysis	Sophisticated, takes considerable time	only basic operations	no
Boundary-based [14], [19]	geometric, high accuracy is possible	not very suitable	no	no	basic types of analysis	Model matching can be costly	only basic operations	no

Table 1: An assessment of geometric-based approaches based on our pre-defined criteria

4.1.2 Boundary-based models

Boundary-based models represent the obstacles' boundaries of an indoor space with sequences of primitive geometries: points, lines, curves, etc. In most cases, a boundary-based map is constructed by concatenating a set of lines segments extracted from sensor data and representing obstacles as polygons [14], [19], [71]. The resulting extracted objects are then matched to a geometric map. In contrast to cell-based approaches, boundary-based geometric models are continuous and can be significantly more compact. However, model matching can be computationally very expensive. Indeed, the line extraction sensors need to be extremely accurate so that lines representing walls, for instance, can be properly positioned at the right location and the right angle on the map. That is, a single match operation necessitates many computations. Moreover, cell-based models are more suitable for navigational services, like path planning, or for communication because they lend themselves very well to applying standard techniques and algorithms for searching in the space, finding the most appropriate path, and many other services.

4.2 Symbolic-based approaches

Symbolic approaches have long attempted to model indoor environments using topological-based structures [31], [22], graphs by capturing the connectivity and reachability between cellular units [17], [34], and hierarchies [17], [69], [9]. The main advantage of these approaches is that object location is provided semantically by using human-readable symbols. Symbolic coordinates allow modelling of topological relations between entities in the environment. Depending on their nature, symbolic location models can then reveal relationships between landmarks such as *containment*, *connectedness*, *closeness*, and *overlapping* relationships. Using the containment relationship, for instance, one can derive a hierarchical structure of the model in order to take into account some application constraints and properties. Hence, symbolic location models enable semantic reasoning at an abstract level, thus favouring interaction between neighbourhoods and within the space. However, symbolic models are incapable of supporting a highly accurate indoor positioning system, and are less suitable for determining the euclidean distance between spatial entities. Moreover, the set of symbols always depends on the application domain and needs to be created and managed manually. Managing a very large number of location symbols requires a considerable modelling effort.

In this paper, we propose to classify symbolic location models into two categories: set-based models and graph-based models. The set-based approach gathers object or place identifiers into sets and subsets; these are used to define spatial relations between elements of a set or, for instance,

the intersection between two subsets, etc. [7], [11]. These sets can be further hierarchically organised to form a tree- or lattice-based structure [31], [22], [49]. Graph-based approaches represent a space as a topological graph where nodes symbolize predefined landmarks (e.g., place, gate, sensor range, object, etc.) extracted either manually or automatically from the environment, and edges stand for the connections that make it possible to move through these landmarks [62], [17], [61], [77]. These two categories will be discussed further in the following two sections.

4.2.1 Set-based symbolic models

As mentioned, set-based models identify places or objects of interest within an indoor space, and then gather these symbolic identifiers into sets. We can distinguish two kinds of set-based models: the first group considers creating a set of place identifiers based on the architectural division of the space [7], [48]. The second deals with all architectural entities (e.g., walls, doors, sensors, etc.) that may contribute to building a smart environment [11].

4.2.1.1 Place

Indeed, indoor spaces are constrained by the architectural components such as doors, floors, corridors, and walls. So, the first work has focused on the process of deriving basic semantics directly from the physical space by creating sets and subsets of place identifiers based on the architectural division of the space. A typical example may be to consider all places that belong to a building, such that each floor is contained within at most one building and each room is contained within at most one floor. In this case, a superset is likely to be defined as the set of floor numbers, and at a lower level, a subset related to each floor that aggregate all room numbers has to be created. The model is hierarchically organised by nature, and can be viewed as a tree structure in which location information is represented at different levels of precision. This model can be used to determine containment relationship between places or between to subsets of places. In addition, by modelling sets of neighbouring symbolic identifiers of places, we can reflect the adjacency relation between them as well as some qualitative notions of distance i.e., we can deduce that the distance between two neighbouring rooms is smaller than the distance between two distant rooms (see [7] for more details). However, the main shortcoming of this approach is its inability to model connectedness between places as, for instance, two neighbouring rooms may or may not be interconnected. Also, it is often incapable of determining quantitative distance efficiently. In addition, this tree-based structure is built with respect to the containment relationship and it does not allow for an element to inherit from multiple parents; this means that a
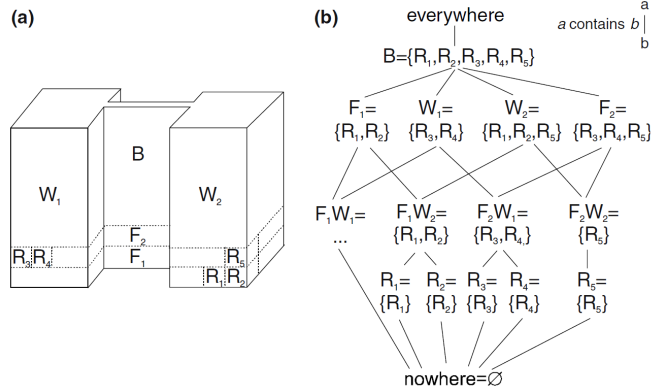
Figure 8: A lattice-based topological model that illustrates overlapping places [22]

place cannot be contained within two overlapping subsets. This is the case where floors are shared between several wings within a building. In such a situation, a room may belong to a floor and a wing at the same time. This problem has been dealt with by using a lattice structure instead of a conventional tree.

As defined in [36], a fine-element lattice is a partially ordered set in which every subset can have a least upper bound and a greatest lower bound. These characteristics of lattices were then applied on a set of places P along with the containment relationship $\leq$ (i.e., $p_1 \leq p_2$ means p_2 spatially contains p_1) in order to construct the model. Two special elements can be distinguished: P as the least upper bound since $p \leq$ P, $\forall p \in$ P, and ϕ as the greatest lower bound because $\phi \leq p$, $\forall p \in$ P. This model overcomes the last mentioned problem about overlapping subsets by allowing for multiple inheritance, and it is very suitable for range queries (e.g., which objects are in a certain place?) as used by [22] to send *geocast* messages. However, this model does not lend itself to other tasks like navigation, because it still does not represent connectedness between places.

A lattice-based semantic model of the space has been presented by [48], which uses the location-exit concept and some mathematical foundations based on the formal concept analysis (FCA) [29]. The exit-location approach models the space with two types of entities: a location described as a bounded geometric area bordered by one or more exits, and an exit as a boundary gate that allows moving in or out of a location [31], [46]. A fundamental element of the lattice structure is called *formal concept*, which is made of a pair (X, Y) where X is a subset of the set of locations and Y is a subset of the set of exits, such that all exits that are shared by the set of locations in X have to be in Y, and all locations that are bounded by the
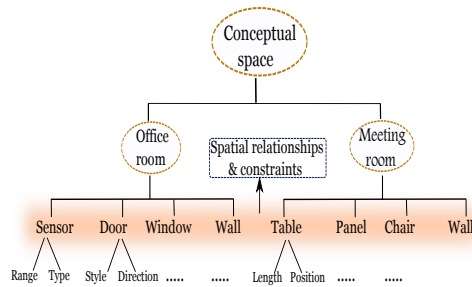
Figure 9: An object-oriented model of an indoor space

set of exits in Y have to be in X. The hierarchical structure of the lattice is then constructed based on these formal concepts and by applying the containment relationship between locations. This model has furthermore been applied to find optimal paths based on the nearest neighbour relation.

4.2.1.2 Object-oriented model

Bhatt and his colleagues [11] proposed an object-oriented model of an indoor environment. They introduced an approach that annotates objects of interest in an indoor space with semantic information according to the properties of the environment. Space was then defined as the combination of three complementary modules: the conceptual module gives terminological information and functional characteristics about architectural entities (e.g., sensor, sliding door, meeting room, computer science laboratory, etc.) regardless of how and where they will be used. The quality module gives qualitative descriptions about spatial dependencies and relationships with other entities. Finally, architectural entities are quantitatively described in the quantity module. The advantage of this approach is that it qualifies the different roles played by the objects in the environment, these roles being spatially qualified. A example of such a conceptual space is illustrated in figure 9.

4.2.2 Graph-based models

Graph-based approaches represent a space as a topological graph where nodes symbolize predefined points of interest (e.g., place, gate, sensor range, object, etc.) extracted either manually or automatically from the environment, and edges stand for the connections that make it possible to move through these landmarks [62], [17], [61], [77]. For instance, a classic topological graph that directly reflects the architecture of a floor plan would be to represent rooms as nodes and doorways as edges; this can simply express connectedness relationship between the architectural entities [27].

In this category, we will discern between two main modelling concepts: *architecture-based* and *architecture-independent* models. In the following section, we make a distinction between these two concepts.

The following three representations rely on graphs where nodes are more or less derived or extracted from the structure of the space. This can be obviously exhibited by the very basic graph models that can be referred to as *place graphs*. *Voronoi-based graphs* are also constructed by extracting meet points and boundary points directly from the space as described earlier. These points were then used as nodes of the generalised voronoi graph. *Fine-grained graphs* preserve indirectly structural properties of the environment since nodes are evenly distributed over the entire space. Some other approaches have adopted graph models that are structurally independent. Nodes are then not directly derived from the architecture, but instead are extracted by means of a sensor deployment strategy within the space.

Place graphs   In their simplest form, place graphs clearly materialize topological properties of the space. In this approach, nodes stand for places such as rooms and/or hallways, and doorways that connect these places appear as edges. Besides the connectivity relationship artlessly presented, other variants of topological relations between structural entities can be inferred such as adjacency and containment properties by annotating nodes and edges. So far, this modelling concept has been widely used since it allows to easily navigate between places, plan routes to destinations, and is well-suited for nearest neighbour queries. Nevertheless, this approach still does not consider interacting objects of the space. It also has a less accurate location information that does not meet specific application requirements. Geometric properties of space disappear, and no real solution is presented, so far, to model a semantic distance function which helps, for instance, in determining the shortest path.

A semantic exit-location based model has been presented by [31]. The aim of this modelling approach is not only to preserve the advantages of the classic place graph model, but also to integrate geometric information such as a distance model to overcome some of the aforementioned problems. The exit-location approach models the space with two types of entities: a location described as a bounded geometric area bordered by one or more exits, and an exit as a boundary gate that allows moving in or out of a location. They have introduced a series of algorithms for, first, automatically extracting location and exit entities from a digital map of an indoor environment. They then programmatically construct the location and exit hierarchies to derive the symbolic location model. At the same time, this symbolic model is built on some geometric information that maintains distance semantics to support services such as nearest neighbour search, shortest path, and location-aware navigation. Although this approach has dealt with many
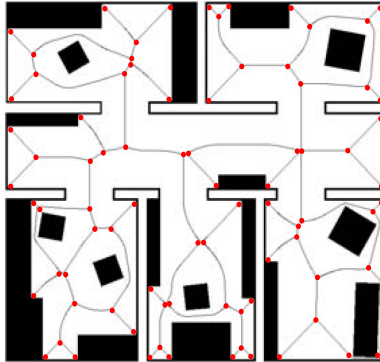
Figure 10: A generalised voronoi graph of an indoor space [73]

critical problems, other advanced contextual queries were still not addressed because the model cannot fully support object movement and can provide only one type of service, that is, the *reachability*.

Generalised Voronoi Graph   As previously-mentioned, the GVD is an attractive approach that represents the built environment as a network of one-dimensional curves which concisely exhibit pathways suitable for navigational purposes. There is a way to extract a route-based graph that brings together all voronoi edges provided by the GVD. Such a route graph is referred to as the generalised Voronoi graph (GVG) which directly reflects the pathways through obstacles [17]. Voronoi-based approaches are perfectly suitable for navigation services such as finding a collision-free path to a destination and other navigational queries described earlier. In addition, a GVG inherits all advantages from applying a symbolic representation so that it can, for instance, annotate its nodes by additional information thus favouring location-aware communication. Some techniques were also proposed to further prune and remove irrelevant nodes and edges so that the whole graph can be more compact [73], [74]. A main drawback of this approach is its inability to provide activity-oriented interactions between moving users and their physical surroundings. Moreover, an indoor space, such as an office building, can be populated with a huge number of objects of interest, which may significantly increase the number of nodes and edges that constitute the graph. Furthermore, the location accuracy problem is raised in specific situations when moving in free-areas, which means that the model does not lend itself to completely support ubiquitous positioning sensors.

Fine-grained graphs   Although the preceding approaches consider an indoor space at a more abstract level and so have a more compact representation, they appear to be badly-suited to context-aware navigation services
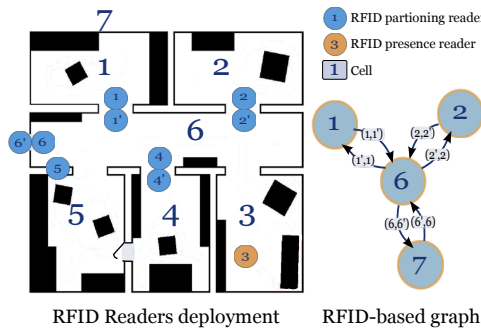
Figure 11: An object-based graph

since interactions at a fine level within the space are increasingly needed. In a companion paper [51], the authors have presented a graph-based model at a fine level of granularity that retains continuous and structural-based properties of the space. Nodes in this model represent cells within an occupancy grid, while connections between cells are materialized by edges. Nodes and edges are labelled according to their belonging to a given spatial unit such as a room or a connecting space. One advantage of this modelling approach is that it achieves a maximum coverage of the space so that continuous positioning sensors used for indoor navigation can be supported. Indeed, the geometrical properties are implicitly represented by the continuous layout of the graph and reflect indirectly the shapes of places being considered. However, a major difficulty that still needs to be confronted regarding performance-related criteria and specially when considering efficiency and scalability issues. A large indoor space could comprise thousands of nodes and edges so that managing and executing real-time queries could be an excessive time- and memory-consuming process.

Sensor-based graphs Many approaches have adopted a more sensor-oriented point of view when the objective is mostly oriented to the tracking and monitoring of mobile objects [34], [8]. For instance, the model presented in [8] describes the space as a set of layers, the topographic space and the sensor space, and establishes connections between layers based on the sensor coverage. Deployment graphs were proposed in [34] by using different types of positioning sensors in order to improve indoor tracking accuracy (see Fig. 11).

5 Towards hybrid spatial models

Numerous spatial models have been presented in different application domains with the aim of combining advantages of geometric and symbolic

Model	Localisation	Navigation	Communication	Interaction	Spatial analysis	Modelling effort	Efficiency	Flexibility
Place-based set [7], [22], [48]	symbolic, set-based, room level accuracy	supports containment and adjacency relationships as well as overlapping in case of lattice-based models, not suitable to compute geometric functions	supports human-recognizable symbols	no	room-level analysis	Medium to high (for managing symbols)	efficient and scalable thanks to the hierarchical structure	supports a large range of applications
Object-oriented set [11]	symbolic, set-based, no accurate geometric information	supports containment relationship, not directly suitable to various navigational tasks	supports human-recognizable contextual information	enrich semantics with and roles	allows for semantic analysis	Medium to high (for managing semantic information)	scalable thanks to the hierarchical structure	supports semantic-based applications
Place-based graph [31], [49]	symbolic, graph-based, room level accuracy	connectedness relationship is materialized in addition to those expressed by set-based models, not suitable to compute geometric functions	supports human-recognizable symbols	no	room-level analysis	Medium to high (for managing symbols)	good	supports a large range of applications
Generalised voronoi graph [17], [73]	symbolic, graph-based, poor accuracy in free space	perfectly suitable for basic navigation services	needs to integrate annotations	no	does not support fine-grained analysis	Sophisticated, takes considerable time	could be efficient by removing irrelevant nodes	supports a large range of navigation services
Fine-grained graph [51]	symbolic, graph-based, highly accurate	perfectly suitable for various navigation services	needs to integrate annotations	needs to enrich with semantics and roles	supports fine-grained analysis	easy to build and to maintain	consumes high memory and processor time, poor in large space	supports a large range of navigation services
sensor-based graph [34], [8]	symbolic, graph-based, accuracy depends on sensor range	yes (depends always on sensor range)	needs to integrate annotations	no	Coarse-grained analysis	sophisticated	Very compact	supports tracking services

Table 2: An assessment of symbolic approaches based on our pre-defined criteria

approaches [72], [47], [35], [12], [73]. Obviously, both classes of models have complementary strengths and weaknesses since they have two fundamentally different purposes. Whereas geometric models can efficiently integrate metric properties to provide highly accurate location and distance information that are necessary elements in most of context-aware applications, symbolic models maintain a more abstract view of the space by providing users with easily-recognizable information and by materializing more complex relationships between entities. Neither approach is, therefore, directly suitable for fulfilling the requirements of context-aware services. Clearly, integration of geometrical and topological representations imply the integration of qualitative and quantitative points of view, thus favouring a large spectrum of interacting representations and applications as suggested by the spatial semantic hierarchy (SSH) introduced by Benjamin Kuipers [41]. This enables human reasoning and human- or robot-centred activities and even sensor-based interactions within the environment.

Accordingly, the idea of integrating different coexistent models of space appeared really as a promising alternative. Hybrid spatial model can be produced in different manner by applying various kinds of organization [12], [73], [7]: (1) parallel models (also referred to as overlays) aim at using different spatial models (usually a combination of geometric and symbolic models) that cover the entire space [72], [73], [67], [8], [51], (2) patchwork-based approaches represent a space with several local, usually geometrical, models that are linked together to form a global, usually symbolic, model of the space [41], and (3) hierarchical models that embeds different layers with different levels of abstraction [24], [25], [35].

In modern robot mapping, many studies have focused on how to extract a topological graph from a basic geometric map such as a grid-based map or a voronoi diagram. The resulting graph is then pruned by applying some algorithms that select the more relevant nodes, thus implicitly providing a hierarchical representation as those encompass knowledge of the environment at different levels of abstraction [72], [73]. Earlier, the SSH model has been presented by Kuipers in [42], [41], and further extended in [43], [10]. The SSH consists of a hierarchy of representations that are inter-dependent. This hierarchy is mainly made of local geometrical maps that correspond to the human or robotŠs sensory horizon, and are merged together based on topological relationships which, in turn, are derived thanks to causal state-action-state schemas.

A good example of a hybrid spatial model designed to handle location-dependent queries was presented in [35]. The model is a combination of a hierarchical, set-based representation of the space and a geometric representation of places, sensor ranges, and objects of interest. These geometric annotations are attached to the corresponding elements in the symbolic set. The model achieves a good trade-off between geometrical and symbolic approaches by combining the benefits from both sides. Although this model

handles some relevant location-dependent queries such as *Şfind the nearest object of interestŤ*, it lacks a clear process description of how to acquire an accurate location of the mobile user continuously. In addition, queries are handled based only on the distance parameter while most of context-aware applications require the integration of context dimensions that aim to find the optimal solution not always the nearest one. In [67], the author proposed to symbolize a physical space with a set of layers that corresponds to different location models designed to meet various activities performed by the users. At the basic layer, the model embeds a quadtree by considering some points of interest. On top of this layer, various topological models were added depending on usersŠ activities and formed a lattice model. The lattice model has been used together with a simple graph model, extracted to materialize connections, to perform relevant location-dependent queries such as position, range, and path queries.

The annotated hierarchical graph model (AH-graph), presented by Fernandez & Gonzalez [25], [24], constitutes a comprehensive framework for efficient mobile navigation. The AH-graph consists of multiple topological layers defined at different levels of abstraction and linked together based on abstraction functions developed for nodes and arcs. An annotation function is also defined at each level so that information can be attached to nodes and arcs. This allows performing basic queries (e.g., path searching) hierarchically, thus achieving better performance. An extension of the model was then presented by adding multiple hierarchies produced based on several scenarios specified at the application level, thus favouring a large spectrum of applications and achieving a high level of flexibility. The model has been further used in [61] to implement the SSH topological level thanks to its efficiency and flexibility. In [28], the AH-graph hierarchy was appended to a semantic hierarchy to further improve human-robot communication.

Generally, parallel models that combine, for instance, a graph-based model with a geometrical model underneath, tend not to scale well to large environments since they could not get rid of weaknesses of geometric models with respect to efficiency and scalability. Hierarchical models, in contrast, scale very to large environments since queries such as path search are performed hierarchically by switching from fine to coarser levels and vice versa. In addition, a specific level of granularity can be used in specific situations with respect to application constraints and usersŠ preferences. Approaches that integrate hierarchical organization require, however, to maintain connections between levels and to integrate a fine-grained geometric model that guarantees accurate localisation for specific navigational purposes. In a previous work [2], we have introduced a fine-grained and multi-layered spatial model that provides a flexible representation of an indoor space, and also takes into account the objects supported and acting in the environment. The modelling approach integrates different levels of granularity, from micro to macro levels, according to available positioning techniques, the infrastruc-

29

ture, device capabilities and user preferences.

6 Conclusion

Location information is an unavoidable context dimension to be considered in the design of context-aware information systems. Context-aware navigation systems require the use and integration of an appropriate indoor spatial model that meets the requirements and satisfies application and structural constraints. In this paper, we defined several service- and performance-related criteria that aim to evaluate existing models from a context-aware systems perspective. A taxonomy of indoor spatial models has been presented and assessed based on these criteria. Two classes of models are discerned: symbolic and geometric spatial models. Whereas geometric models can efficiently integrate metric properties to provide highly accurate location and distance information that are necessary elements in most of context-aware applications, symbolic models maintain a more abstract view of the space by providing users with easily-recognizable information and by materializing more complex relationships between entities. Hybrid spatial models are then presented as an appealing approach that can fulfil context-aware application requirements. Particularly, we believe that a hierarchical organization of location information, which materializes containment and connectedness relationships between entities, coupled with a fine-grained geometrical model may guarantee flexibility, efficiency, and accuracy so that a large range of context-aware queries can be performed.

References

[1] L. Aalto, N. Gothlin, J. Korhonen, and T. Ojala. Bluetooth and WAP push based location-aware mobile advertising system. In *Proceedings of the 2nd International Conference on Mobile Systems, Applications, and Services*, pages 49–58. ACM, 2004.

[2] I. Afyouni, C. Ray, and C. Claramunt. A fine-grained context-dependent model for indoor spaces. In *Proceedings of the 2nd ACM SIGSPATIAL International Workshop on Indoor Spatial Awareness*, pages 33–38. ACM, 2010.

[3] E. Aitenbichler. Event-based and publish/subscribe communication. In *Handbook of Research on Ubiquitous Computing Technology for Real Time Enterprises*, pages 152–171. Information Science Publishing, 2008.

[4] F. Aurenhammer. Voronoi diagrams: A survey of a fundamental geometric data structure. *ACM Computing Surveys*, 23(3):345–405, 1991.

[5] J.E. Bardram. Activity-based computing: Support for mobility and collaboration in ubiquitous computing. *Personal and Ubiquitous Computing*, 9(5):312–322, 2005.

[6] S. Basagni, I. Chlamtac, and V.R. Syrotiuk. Geographic messaging in wireless ad hoc networks. In *Proceedings of the 49th Vehicular Technology Conference, VTC'99*, volume 3, pages 1957–1961. IEEE, 1999.

[7] C. Becker and F. Durr. On location models for ubiquitous computing. *Personal and Ubiquitous Computing*, 9(1):20–31, 2005.

[8] T. Becker, C. Nagel, and TH Kolbe. Supporting contexts for indoor navigation using a multilayered space model. In *Proceedings of the 10th International Conference on Mobile Data Management: Systems, Services and Middleware*, pages 680–685. IEEE, 2009.

[9] T. Becker, C. Nagel, and T.H. Kolbe. 1st technical report discussion of euclidean space and cellular space and proposal of an integrated indoor spatial data model. Technical report, Institute of Geodesy and Geoinformation Science, 2010.

[10] P. Beeson, N.K. Jong, and B. Kuipers. Towards autonomous topological place detection using the extended voronoi graph. In *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA*, pages 4373–4379. IEEE, 2005.

[11] M. Bhatt, F. Dylla, and J. Hois. Spatio-terminological inference for the design of ambient environments. In *Proceedings of the 9th International Conference on Spatial Information Theory*, pages 371–391. Springer-Verlag, 2009.

[12] P. Buschka. *An investigation of hybrid maps for mobile robots*. PhD thesis, Örebro Universitetsbibliotek, Sweden, 2005.

[13] A.J. Champandard. *AI game development: Synthetic creatures with learning and reactive behaviors*. New Riders Pub, 2003.

[14] R. Chatila and J. Laumond. Position referencing and consistent world modeling for mobile robots. In *Proceedings of the International Conference on Robotics and Automation*, volume 2, pages 138–145. IEEE, 1985.

[15] G. Chen and D. Kotz. A survey of context-aware mobile computing research. Technical Report TR2000-381, Dartmouth College Hanover, NH, USA, 2000.

[16] H. Choset. Incremental construction of the generalized voronoi diagram, the generalized voronoi graph, and the hierarchical generalized voronoi

graph. In *Proceedings of the 1st CGC Workshop on Computational Geometry*, 1997.

[17] H. Choset and J. Burdick. Sensor-based exploration: The hierarchical generalized Voronoi graph. *International Journal of Robotics Research*, 19(2):96–125, 2000.

[18] H. Christensen and J. Bardram. Supporting human activities exploring activity-centered computing. In *Proceedings of the 4th International Conference on Ubiquitous Computing, UbiComp*, pages 107–116. Springer-Verlag, 2002.

[19] J.L. Crowley. World modeling and position estimation for a mobile robot using ultrasonic ranging. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 674–680. Computer Society Press, 1989.

[20] D. Demyen and M. Buro. Efficient triangulation-based pathfinding. In *Proceedings of the 21st National Conference on Artificial Intelligence*, pages 942–947, 2006.

[21] P. Dourish. *Where the action is: The foundations of embodied interaction.* The MIT Press, 2004.

[22] F. Durr and K. Rothermel. On a location model for fine-grained geocast. In *Proceedings of the 5th International Conference on Ubiquitous Computing, UbiComp*, pages 18–35. Springer, 2003.

[23] A. Elfes. Using occupancy grids for mobile robot perception and navigation. *Computer*, 22(6):46–57, 1989.

[24] J.A. Fernandez and J. Gonzalez. Hierarchical graph search for mobile robot path planning. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 656–661. IEEE, 1998.

[25] J.A. Fernandez and J. Gonzalez. *Multi-hierarchical representation of large-scale space: Applications to mobile robots.* Kluwer Academic Publishers Norwell, MA, USA, 2002.

[26] S. Fortune. A sweepline algorithm for voronoi diagrams. *Algorithmica*, 2(1):153–174, 1987.

[27] G. Franz, H. Mallot, J. Wiener, and K. Neurowissenschaft. Graph-based models of space in architecture and cognitive science: A comparative analysis. In *Proceedings of the 17th International Conference on Systems Research, Informatics and Cybernetics, InterSymp*, pages 30–38, 2005.

[28] C. Galindo, A. Saffiotti, S. Coradeschi, P. Buschka, JA Fernandez-Madrigal, and J. Gonzalez. Multi-hierarchical semantic maps for mobile robotics. In *Proceedings of the International Conference on Intelligent Robots and Systems, IROS*, pages 2278–2283. IEEE, 2005.

[29] B. Ganter, R. Wille, and R. Wille. *Formal concept analysis.* Springer Berlin, 1999.

[30] D. Heckmann and A. Krueger. A user modeling markup language (UserML) for ubiquitous computing. In *Proceedings of the 9th International Conference on User Modeling*, pages 393–397. Springer-Verlag, 2003.

[31] H. Hu and D.L. Lee. Semantic location modeling for location navigation in mobile environment. In *Proceeding of the IEEE International Conference on Mobile Data Management*, pages 52–61, 2004.

[32] S. Ilarri, E. Mena, and A. Illarramendi. Location-dependent query processing: Where we are and where we are heading. *ACM Computing Surveys (CSUR)*, 42(3):1–73, 2010.

[33] G.S. Iwerks. *Maintenance of spatial queries on continuously moving points.* PhD thesis, University of Maryland at College Park College Park, MD, USA, 2004.

[34] Christian S. Jensen, Hua Lu, and Bin Yang. Graph model based indoor tracking. In *Proceedings of the 10th International Conference on Mobile Data Management: Systems, Services and Middleware*, pages 122–131. IEEE, 2009.

[35] C. Jiang and P. Steenkiste. A hybrid location model with a computable location identifier for ubiquitous computing. In *Proceedings of the 4th International Conference on Ubiquitous Computing, UbiComp*, volume 2498, pages 246–263. Springer-Verlag, 2002.

[36] W. Kainz, M.J. Egenhofer, and I. Greasley. Modelling spatial relations and operations with partially ordered sets. *International Journal of Geographical Information Science*, 7(3):215–229, 1993.

[37] H.K. Kang, K.J. Li, and M.S. Kim. A framework for dynamic updates of map data in mobile devices. *International Journal of Web Engineering and Technology*, 3(2):176–195, 2007.

[38] V. Kaptelinin and B.A. Nardi. *Acting with technology: Activity theory and interaction design.* The MIT Press, 2009.

[39] A. Kofod-petersen and J. Cassens. Using activity theory to model context awareness. In *Proceedings of the 2nd International Workshop on Modeling and Retrieval of Context*, pages 1–17. Springer, 2006.

[40] K.W. Kolodziej and J. Hjelm. *Local positioning systems: LBS applications and services*. CRC Press, 2006.

[41] B. Kuipers. The Spatial Semantic Hierarchy. *Artificial Intelligence*, 119(1-2):191–233, 2000.

[42] B. Kuipers and Y.T. Byun. A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations. *Robotics and Autonomous Systems*, 8(1-2):47–63, 1991.

[43] B. Kuipers, J. Modayil, P. Beeson, M. MacMahon, and F. Savelli. Local metrical and global topological maps in the hybrid spatial semantic hierarchy. In *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA*, volume 5, pages 4845–4851. IEEE, 2004.

[44] H. Ledoux. *Modelling three-dimensional fields in geoscience with the Voronoi diagram and its dual*. PhD thesis, GIS Research Centre, School of Computing, 2006.

[45] H. Ledoux and C.M. Gold. Modelling three-dimensional geoscientific fields with the voronoi diagram and its dual. *International Journal of Geographical Information Science*, 22(5):547–574, 2008.

[46] D.L. Lee, M. Zhu, and H. Hu. When location-based services meet databases. *Mobile Information Systems*, 1(2):81–90, 2005.

[47] U. Leonhardt. *Supporting Location-Awareness in Open Distributed Systems*. PhD thesis, Imperial College, 1998.

[48] D. Li and D.L. Lee. A lattice-based semantic location model for indoor navigation. In *Proceedings of the 9th International Conference on Mobile Data Management*, pages 17–24. IEEE, 2008.

[49] D. Li and D.L. Lee. A topology-based semantic location model for indoor applications. In *Proceedings of the 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 1–10. ACM, 2008.

[50] Ki-Joune Li. Indoor space: A new notion of space. In *Proceedings of the 8th International Symposium on Web and Wireless Geographical Information Systems, W2GIS*, pages 1–3, 2008.

[51] X. Li, C. Claramunt, and C. Ray. A grid graph-based model for the analysis of 2D indoor spaces. *Computers, Environment and Urban Systems*, 34(6):532–540, 2010.

[52] F. Lyardet, D.W. Szeto, and E. Aitenbichler. Context-aware indoor navigation. In *Proceedings of the 2nd European Conference in Ambient Intelligence, AmI*, pages 290–307. Springer-Verlag, 2008.

[53] M. Mekni. *Automated generation of geometrically-precise and semantically-informed virtual geographic environments populated with spatially-reasoning agents*. Universal-Publishers, 2010.

[54] M.F. Mokbel and J.J. Levandoski. Toward context and preference-aware location-based services. In *Proceedings of the 8th ACM International Workshop on Data Engineering for Wireless and Mobile Access*, pages 25–32. ACM, 2009.

[55] D.R. Montello. Navigation. *The Cambridge handbook of visuospatial thinking*, 18:257–294, 2005.

[56] HP Moravec and A. Elfes. High Resolution Maps from Wide Angle Sonar. In *Proceedings of the International Conference on Robotics and Automation*, volume 2, pages 116–121. IEEE, 1985.

[57] B.A. Nardi. *Context and consciousness: activity theory and human-computer interaction*. The MIT Press, 1996.

[58] J.C. Navas and T. Imielinski. GeoCastŮgeographic addressing and routing. In *Proceedings of the 3rd Annual ACM/IEEE International Conference on Mobile Computing and Networking*, pages 66–76. ACM, 1997.

[59] M. Petit. *Approche spatiale pour la caractérisation du contexte d'exécution d'un système d'information ubiquitaire*. PhD thesis, Arts et Métiers ParisTech - Institut de Recherche de l'Ecole Navale, 2010.

[60] N.B. Priyantha, A. Chakraborty, and H. Balakrishnan. The cricket location-support system. In *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*, pages 32–43. ACM, 2000.

[61] E. Remolina, J.A. Fernandez, B. Kuipers, and J. Gonzalez. Formalizing regions in the spatial semantic hierarchy: An AH-graphs implementation. In *Proceedings of the International Conference on Spatial Information Theory: Cognitive and Computational Foundations of Geographic Information Science, COSIT*, pages 109–124. Springer Verlag, 1999.

[62] E. Remolina and B. Kuipers. Towards a general theory of topological maps. *Artificial Intelligence*, 152(1):47–104, 2004.

[63] G. Retscher. Ubiquitous positioning solutions for pedestrian navigation. In *Proceedings of the 5th International Symposium on Mobile Mapping Technology*, 2007.

[64] H. Samet. The quadtree and related hierarchical data structures. *ACM Computing Surveys*, 16(2):187–260, 1984.

[65] I. Satoh. Location-aware communications in smart spaces. In *proceedings of the International Conference on Multimedia and Ubiquitous Engineering, MUE*, pages 1027–1034. IEEE, 2007.

[66] R. Siegwart and I.R. Nourbakhsh. *Introduction to autonomous mobile robots*. The MIT Press, 2004.

[67] V. Stirbu. NavTag: An inter-working framework based on tags for symbolic location coordinates for smart spaces. In *Proceedings of the 6th International Conference on Ubiquitous Intelligence and Computing, UIC*, pages 266–280. Springer, 2009.

[68] Edgar-Philipp Stoffel. *Hierarchical graphs as organisational principle and spatial model applied to pedestrian indoor navigation*. PhD thesis, LMU München: Faculty of Mathematics, Computer Science and Statistics, 2009.

[69] E.P. Stoffel, K. Schoder, and H.J. Ohlbach. Applying hierarchical graphs to pedestrian indoor navigation. In *Proceedings of the 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM, 2008.

[70] Y. Tao, D. Papadias, and Q. Shen. Continuous nearest neighbor search. In *Proceedings of the 28th International Conference on Very Large Data Bases*, pages 287–298. VLDB Endowment, 2002.

[71] S. Thrun. To know or not to know: On the utility of models in mobile robotics. *AI Magazine*, 18(1):47–55, 1997.

[72] S. Thrun. Learning metric-topological maps for indoor mobile robot navigation. *Artificial Intelligence*, 99(1):21–71, 1998.

[73] J.O. Wallgrun. Autonomous construction of hierarchical voronoi-based route graph representations. In *Spatial Cognition IV. Reasoning, Action, and Interaction*, pages 413–433. Springer-Verlag, 2005.

[74] J.O. Wallgrun. *Hierarchical voronoi graphs: Spatial representation and reasoning for mobile robots*. Springer, 2010.

[75] R. Want, A. Hopper, V. Falcao, and J. Gibbons. The active badge location system. *ACM Transactions on Information Systems (TOIS)*, 10(1):91–102, 1992.

[76] A. Ward, A. Jones, and A. Hopper. A new location technique for the active office. *IEEE Personal Communications*, 4(5):42–47, 1997.

[77] S. Werner, B. Krieg-Bruckner, and T. Herrmann. Modelling navigational knowledge by route graphs. In *Spatial Cognition II, Integrating Abstract Theories, Empirical Studies, Formal Methods, and Practical Applications*, pages 295–316. Springer-Verlag, 2000.

[78] K.L. Wu, S.K. Chen, and P.S. Yu. Incremental processing of continual range queries over moving objects. *IEEE Transactions on Knowledge and Data Engineering*, 18(11):1560–1575, 2006.

[79] S. Wu and H.H. Fan. Activity-based proactive data management in mobile environments. *IEEE Transactions on Mobile Computing*, 9(3):390–404, 2009.

[80] W. Yuan and M. Schneider. Supporting continuous range queries in indoor space. In *Proceedings of the 11th International Conference on Mobile Data Management*, pages 209–214. IEEE, 2010.

[81] J. Zhang, M. Zhu, D. Papadias, Y. Tao, and D.L. Lee. Location-based spatial queries. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 443–454. ACM, 2003.