INT. J. GEOGRAPHICAL INFORMATION SCIENCE, 1999, VOL. 13, NO. 8, 775–796

Research Article

Progress in computational methods for representing geographical concepts

MAX J. EGENHOFER

National Center for Geographic Information and Analysis, and Department of Spatial Information Science and Engineering, Department of Computer Science, University of Maine, Orono, ME 04469-5711, USA.
E-mail: max@spatial.maine.edu

JANICE GLASGOW

Department of Computing and Information Science, Queen's University, Kingston, Ontario, K7L 3N6, Canada

OLIVER GÜNTHER

Institut für Wirtschaftsinformatik, Humboldt University, Berlin, Germany, and Pôle Universitaire, Paris, France

JOHN R. HERRING

Oracle Corporation, 196 Van Buren Street, Herndon, VA 22070, USA

and DONNA J. PEUQUET

Department of Geography, Pennsylvania State University, 302 Walker Building, University Park, PA 16802, USA

Abstract. Over the past ten years, a subfield of GIScience has been recognized that addresses the linkage between human thought regarding geographical space, and the mechanisms for implementing these concepts in computational models. This research area has developed an identity through a series of successful international conferences and the establishment of a journal. It has also been complemented through community activities such as international standardization efforts and GIS interoperability. Historically, much of the advancement in computational methods has occurred at—or close to—the implementation level, as exemplified by attention to the development of spatial access methods. Significant progress has been made at the levels of spatial data models and spatial query languages, although we note the lack of a comprehensive theoretical framework comparable to the relational data model in database management systems. The difficult problems that need future research efforts are at the highly abstract level of capturing semantics of geographical information. A cognitive motivation is most promising as it shapes the focus on the users' needs and points of view, rather than on efficiency as in the case of a bottom-up system design. We also identify the need for new research in fields, models of qualitative spatial information, temporal aspects, knowledge discovery, and the integration of GIS with database management systems.

International Journal of Geographical Information Science
ISSN 1365-8816 print/ISSN 1362-3087 online
http://www.tandf.co.uk/JNLS/gis.htm
http://www.taylorandfrancis.com/JNLS/gis.htm

1. Introduction

In the past, much research in the computational domain of geographical information systems (GIS) concerned the development of fast and efficient implementations of traditional cartographic concepts for data storage, retrieval, and analysis. Increased functionality had also characteristically been accompanied by increased conceptual complexity, as improvements were most often motivated by short-term needs, resulting in *ad hoc* solutions. With the increasing availability of GIS, there is an increasing need to provide users—from scientists to average citizens—with tools that allow them to solve their problems better in a more intuitive and user-friendly manner.

In light of these observations, a subfield within Geographical Information Science (GIScience) has developed over the last ten years that addresses the linkage between human thought about geographical space and the mechanisms of computational models. It is particularly concerned with the interface between the real world as perceived and computational geographical worlds. This aspect is important from a scientific perspective, because through such a cognitively-motivated approach, geographical concepts that have always been intuitive but never formalized can be developed into a more formal framework. Such formalizations will enable space-time analyses of large-scale geographical processes that are impossible to perform without the aid of computers.

As part of this latter process, interactions with other fields, including mathematics, computer science, and statistics, need to be tailored. Through formalization, geographical concepts become unambiguously defined such that the danger of miscommunication and misuse is reduced. There is a strong tradition in some parts of GIS that draws upon fundamental mathematical principles, such as the use of algebraic topology (White 1984, Herring 1991), point-set theory (Egenhofer and Franzosa 1991), and partially ordered sets (Kainz *et al.* 1993, Stell and Worboys 1998). Other approaches have relied on models developed in computer science, probably best shown in GIS debates over the use of relational (Morehouse 1985) and object-oriented (Egenhofer and Frank 1987, Worboys *et al.* 1990) or logic-based (Frank 1984, Fernandez *et al.* 1999) data models. Formalizations of geographical processes also lead to spatial languages—not natural language, but formal languages understandable to people as well as machines. These considerations make computational methods an indispensable part of GIScience.

Computational implementations are complementary to other areas of GIScience (e.g. cognitive models and geographies of the information society), and they play a central part in making advances in GIS theory accessible to a large audience, the audience of the users. While computational implementations undoubtedly require at one point engineers and programmers to cut code, the steps that fuel these implementations are complicated and cannot be accomplished without considering how people think about geographical space and time, how to translate human conceptualizations into formalisms that allow these processes to be repetitively consistent, and how to make people interact more naturally with information systems. These three concerns show how computational methods span across GIScience: Spatial thinking extends to the cognitive area; use and interaction about geographical information reaches out to the societal implications; and formalizations of geographical concepts are the core of the research agenda in computational methods.

This article assesses the progress and status of research in the area of computational methods for representing geographical information. After an assessment of this field's community with its major activities, conferences, and publication outlets (the next section), we review some of the results of 15 years of research in computational methods for representing geographical information. In the final section, we look ahead with a discussion of the needs for future research.

2. The community

Researchers and practitioners in the geosciences have long been working on computer solutions to match their specific needs. Commercial GISs are among the most important outcomes of these efforts. Traditionally, computer scientists have been involved only marginally in the development of such systems. This started to change in the 1980s, and the increasing number of contacts is now bearing fruit. The design and implementation of data management tools for spatial applications is pursued by an interdisciplinary community of people from academia, government, and industry. A critical foundation lies in the concept of spatial database systems as an enabling technology (Frank 1988) for a variety of application software, such as CAD systems or GIS. There have been several interdisciplinary projects of high visibility, including the US National Center for Geographic Information and Analysis (*http://www.ncgia.org*), the Sequoia 2000 project (Stonebraker *et al.* 1993, Frew 1994), and the Alexandria Digital Library (Smith 1996).

It was also in the 1980s that a number of conference series were launched that provided a forum for the exchange of research results. In 1984, GIS researchers initiated the International Symposium on Spatial Data Handling (SDH), which quickly became a forum for communication between geographers, engineers, and computer scientists interested in analysing and manipulating geographical data with the help of computers. Five years later, an NCGIA research initiative (Smith and Frank 1990) led to the First Symposium on Large Spatial Databases (SSD) (Buchmann *et al.* 1989). Compared to SDH, SSD has a stronger emphasis on computer technology and draws its audience primarily from researchers who focus on database systems. In 1993 a new direction was established with the First International Conference on Spatial Information Theory (COSIT), which aims at bringing together an interdisciplinary group of researchers who span an even wider range of disciplines, from cognitive science to geography and computer science. All these conference series continue to be held biennially, and their proceedings typically appear as books with major publishers, such as SSD (Buchmann *et al.* 1989, Günther and Schek 1991, Abel and Ooi 1993, Egenhofer and Herring 1995, Scholl and Voisard 1997), SDH (Waugh and Healey 1994, Kraak and Molenaar 1996), and COSIT (Frank and Campari 1993, Frank and Kuhn 1995, Hirtle and Frank 1997). These activities focus on the GIScience community and are complemented by publications in the traditional computer science domain. Such conferences as ACM SIGMOD and Very Large Data Bases (VLDB) typically each year have a session on spatial data management. In addition, an annual ACM Workshop on GIS has been held since 1993, catering primarily to computer scientists who work on models for spatial data and their efficient implementations.

Progress in computational methods for representing geographical information can also be measured in terms of archival outlets that regularly publish articles on this topic. For many years, the *International Journal of Geographical Information Systems* (now the *International Journal of Geographical Information Science*) has been the umbrella for most GIS journal papers. Occasional papers on computational issues in GIS have appeared in several computer science journals, such as *IEEE*

Transactions on Knowledge and Data Engineering, *ACM Transactions on Database Systems*, the *Journal of Visual Languages and Computing*, and the *VLDB Journal*, which published a special issue on Spatial Databases (Güting 1994a). With the formation of the journal *GeoInformatica* in 1996, the field passed another important milestone, resulting in a journal that is dedicated to research results at the interface between GIS and computer science. Overview articles on the status of the field are infrequent. The field moves quickly and several issues that were seen as critical in the early 1990s (Günther and Buchmann 1990) have lost their appeal. Two more recent reviews (Bauzer Medeiros and Pires 1994, Güting 1994b) provide complementary material to the present assessment of the status in the field.

Although the book market in GIS has been growing quickly, there have been few books focusing on computational aspects of GIS. Laurini and Thompson (1992) exposed technical issues in GIS to a larger audience. Hanan Samet's books (1989a, b) provide an in-depth coverage of spatial data structures and geo-algorithms. More recently, Worboys (1996) gave an up-to-date computing perspective on GIS.

Another measure of success is how results from research have found their way into widely available products and standards that may have a broad impact on designers and users. In both aspects, strong linkages to research results are very visible. GIS have become significantly easier to use through the adoption of a Windows-Icons-Menus-Pointers (WIMP) design. GIS also have started to embrace database management systems, an approach that has been suggested and requested since the early 1980s (Frank 1981). It is now being enabled by industry's support of specialized spatial versions of database systems (e.g. Spatial Data Blades and the Spatial Data Option) or middleware (e.g. the Spatial Data Engine). Today's implementations show significant similarities with such research results as R-trees (Guttman 1984), quadtrees (Samet 1989a, b), models for topological relations (Egenhofer and Franzosa 1991), database architecture (Abel and Smith 1986, Abel 1988), and object-oriented models (Worboys *et al.* 1990, Egenhofer and Frank 1992). Other impact, particularly at the level of geo-algorithms, is more difficult to assess since in most cases industry has abstained from publishing technical details about their approaches, and user documentation reveals no view behind the scenes.

At the organizational level, an important event happened in 1994, when an international group of GIS users and vendors founded the Open GIS Consortium (OGC, *http://www.opengis.org*). OGC has quickly become a powerful interest group to advance open systems approaches to geoprocessing and promote an Open Geodata Interoperability Specification. Such a computing framework and set of software specifications would support interoperability in the distributed management of geographical data (Kottman 1999). OGC seeks to make geographical data and geoprocessing an integral part of enterprise information systems. Together with other standardization efforts, such as the International Organization for Standardization's Technical Committee on Geographic Information/Geomatics (ISO TC/211), often diverse computational methods are being consolidated, thereby making GIS technologies more mature.

3. Looking back

We have organized the large variety of computational methods into parts that follow the traditional approach used in computer science for modelling and implementing application domains. At the highest level, one is concerned with spatial data models and spatial query languages, as these parts of a GIS are visible and accessible to users. Since the late 1980s query languages have migrated from an end-user facility to an entry for application programmers, facilitating standardized storage, updating, and retrieval of information. Discussions about higher-level aspects of access have found new fields in the area of user interface design and human-computer interaction. Considerations about user interface design are truly at the interface between the cognitive and computational domains of GIScience and successful approaches typically require multidisciplinary participation, including designers and users. Current GIS user interface techniques (Egenhofer and Kuhn 1999) have moved from the more archaic command lines through WIMP (Window-Icon-Menu-Pointer) designs to direct manipulation (Ahlberg and Shneiderman 1994, Stone *et al.* 1994, Richards and Egenhofer 1995, Bruns and Egenhofer 1997), an interaction mechanism by which users see and manipulate objects to affect commands (Shneiderman 1983).

To implement geographical concepts in a high-level, compact, and reusable way, spatial data types have become the standard method. Although early attempts were made to rely exclusively on those data types that are offered by standard programming languages and database systems (Go *et al.* 1975, Berman and Stonebraker 1977), it has become common practice to identify spatial data types and to link them with their related operations. Object-oriented methods, with encapsulation and hiding of implementations, have favoured this approach, and it also has made its way into relational database systems in the form of extended relational or object-relational database systems.

The third level of our review is concerned with implementation aspects, primarily the need for fast access to the necessary spatial, multidimensional data elements from linear storage devices (Frank 1981). This topic has a long research tradition, and accounts probably for the most frequently researched topic related to GIScience. It includes some of the most frequently cited papers in the database domain—Guttman's R-tree (1984) and the R+ -tree by Sellis *et al.* (1987). We focus on implementations that are typically tailored to vector data models, since this has been the traditional viewpoint in much of the computational GIS domain. Complementary areas, such as image processing, remote sensing, and vision, have relied on different data models and representations, which start with a representation of the underlying space, often in the form of regular spatial subdivisions such as pixels, and aim at extracting higher-level spatial concepts from this structure.

In this section, we review some aspects that have been identified as stable foundations for modelling and implementing geographical concepts. It is by no means exhaustive.

3.1. *Spatial data models and spatial query languages*

The data management requirements of spatial applications differ substantially from those of traditional business applications. Business applications tend to have simply structured data records. There is only a small number of relationships between data items, and transactions—converting a database from one consistent state into another (Gray and Reuter 1993)—are comparatively short. Relational database systems meet these requirements extremely well. Their data model is table-oriented, therefore providing a natural fit to business requirements. By means of the transaction concept, one can check integrity constraints and reject inconsistencies.

For spatial applications, however, conventional concepts from database management systems are often inadequate. Spatial databases contain multidimensional data with explicit knowledge about objects, their extents, and their locations in space.

The objects are usually represented in the cartographic tradition of some vector-based format, and their relative position may be explicit or implicit. They often have a complex structure: a spatial data object may be composed of a single point or several thousands of polygons or various collections of polygons, lines, and points often with complicated consistency constraints. These objects rarely follow regular shapes or patterns of distribution across space. It is usually impossible to store collections of such objects in a single relational table with a fixed tuple size. Moreover, spatial data are dynamic: insertions and deletions are interleaved with updates, and data structures have to support this dynamic behaviour without deteriorating over time. Spatial databases tend to be large, typically occupying several hundred giga-bytes of storage. The seamless integration of secondary and tertiary memory is, therefore, essential for efficient processing (Chen *et al.* 1995).

Recent database research has helped to solve many related problems. These include both extensions to the relational data model (Stonebraker and Rowe 1986, Kemper and Wallrath 1987, Haas *et al.* 1990, Schek *et al.* 1990, Silberschatz *et al.* 1991) and the development of flexible object-oriented approaches for spatial information (Egenhofer and Frank 1987, 1992, Worboys *et al.* 1990, Orenstein 1990, Günther and Lamberts 1994, Shekhar *et al.* 1997).

Any serious attempt to manage spatial data in a relational database framework requires some significant extensions at the logical and the physical level. These kinds of extension need to be supported at the query language level as well. Besides an ability to deal with spatial data types and operators, this involves concepts to support the interactive working mode that is typical for many GIS applications. Pointing to objects or drawing on the screen with the mouse are typical examples of these dynamic interactions. Further extensions at the user interface level include (Egenhofer 1990, Voisard 1995) the graphical display of query results, including legends and labels; the display of unrequested context to improve readability; and the possibility of stepwise refinement of the display (logical zooming).

For many years, the database market has been dominated by the Structured Query Language SQL. There has been a long discussion in the literature as to whether SQL is suitable for querying spatial databases. It was recognized early on that relational algebra and SQL alone are not able to provide this kind of support (Frank 1982, Härder and Reuter 1985, Egenhofer and Frank 1988). 'Why not SQL!' (Egenhofer 1992) gives numerous examples of SQL's lack of expressive power and limitations of the relational model in general when applied to spatial data. At the user interface level, one encounters difficulties when trying to combine retrieval and display aspects in a single SQL query. Besides requiring specialized operators, this kind of combination usually leads to long and complex queries. The integration of selection by pointing (to the screen) is also problematic. There is no support in SQL for the stepwise refinement of queries, which is particularly important in a spatial database context where users often ask questions iteratively. The underlying problem is that SQL does not provide a notion of state maintenance that allows users to interrupt their dialogue at a given point and resume their work later.

In some sense, however, with the success of SQL the discussion about its appropriateness has become a moot point. The question is not whether SQL should be used—SQL is and in the foreseeable future will be used to query spatial databases. The question is rather which kind of extensions are desirable to optimize user friendliness and performance of the resulting spatial data management system.

Various extensions to SQL have been proposed to deal with spatial data

(Egenhofer 1992), including PSQL (Roussopoulos *et al.* 1988), Spatial SQL (Egenhofer 1991, 1994), GEOQL (Ooi and Sacks-Davis 1989, Ooi *et al.* 1989, Ooi 1990), and the SQL-based GIS query languages for KGIS (Ingram and Phillips 1987), and TIGRIS (Herring *et al.* 1988). Current efforts under the umbrella of the ANSI Committee on SQL3 are developing an integrated version of such spatial extensions, called SQL/Multimedia (SQL/MM), which is a suite of standards that specify type libraries using SQL's object-oriented facilities.

3.2. *Spatial data types*

An essential weakness in traditional commercial databases is that they do not provide any spatial data types (Cox *et al.* 1979). Following their orientation towards classical business applications, they may sometimes offer non-standard types such as date and time in addition to the classical data types integer, real, character, and string. Spatial data types, however, are not included in any of today's standard commercial DBMS. On the other hand, such data types are a crucial requirement when it comes to processing geographic data.

For vector data, there have been several proposals on how to define a coherent and efficient spatial algebra (Frank and Kuhn 1986, Güting 1988, 1989, Egenhofer *et al.* 1989, Scholl and Voisard 1989, Güting and Schneider 1993, Egenhofer 1994, Güting *et al.* 1995, Schneider 1997). It is generally assumed that the data objects are embedded in d-dimensional Euclidean space E^d or a suitable subspace. Any point object stored in a spatial database has a unique location in the universe, defined by its d coordinates. Any point in space can be occupied by several point objects stored in the database. A (convex) d-dimensional polytope P in E^d is defined as the intersection of some finite number of closed halfspaces in E^d, such that the dimension of the smallest affine subspace containing P is d. A hyperplane H supports a polytope P if $H \cap P \neq \emptyset$ and P is completely contained in one of the halfspaces defined by H. If H is any hyperplane supporting P then $H \cap P$ is a face of P. The faces of dimension 1 are called edges; those of dimension 0 vertices. By forming the union of some finite number of polytopes $Q_1, \ldots, Q_n$, one obtains a (d-dimensional) polyhedron Q in E^d that is not necessarily convex. Following the intuitive understanding of polyhedra, one usually requires that the Q_i ($i = 1, \ldots, n$) have to be connected. This also allows for polyhedra with holes.

One often uses the terms *line* and *polyline* to denote a one-dimensional polyhedron, and the terms polygon and region to denote a two-dimensional polyhedron. If, for each k ($0 \leq k \leq d$), one views the set of k-dimensional polyhedra as a data type, one obtains the common collection of spatial data types (i.e. point, line, and polygon). Combined types sometimes also occur. Curved objects can be obtained by extending these definitions.

Since there is neither a standard spatial algebra nor a standard spatial query language, there is also no consensus on a canonical set of spatial operators. Different applications use different operators, although some operators (such as intersection) are more common than others. Spatial operators can be classified in several different ways, reflecting fundamentally different perspectives and objectives. A common distinction is based on different geometric properties of spatial relations, leading to groups of topological, directional, and metric relations. Topological relations are invariant under topological transformations, such as translation, rotation, and scaling (Egenhofer and Franzosa 1991). Examples are overlap, disjoint, and inside. Direction relations refer to the location of two spatial objects with respect to a reference frame

(Peuquet and Zhan 1987, Retz-Schmidt 1988, Frank 1991b), yielding quantitative values (e.g. 44° 34′) or qualitative values (e.g. north and south-west; or left and right). Metric relations (Peuquet 1992, Hernández *et al.* 1995) capture distances, either quantitatively (e.g. 24.5 km) or qualitatively (e.g. near and far). For the evaluation of such spatial predicates in a database context, the spatial join operator has been introduced (Orenstein 1986, Rotem 1991, Becker and Güting 1992, Brinkhoff *et al.* 1993, 1994, Günther 1993, Aref and Samet 1994, Gaede 1995b). A spatial join takes two sets of spatial objects as input and produces a set of pairs of spatial objects as output, such that each pair fulfils the given spatial predicate. Examples include, 'Find all houses that are less than 10 km from a lake' or 'Find all buildings that are located within a wetland'.

A different query perspective is given if operators are classified according to their signatures (Günther 1998), that is, the input and output behaviour of each operation. In order to be considered a spatial operator, at least one of the operators has to be of a spatial data type. The input behaviour refers to whether it is a unary or binary operator, as well as to the type of its operands. Operators over more than two operands are typically broken down into a sequence of binary operations. The output behaviour refers to the type of result. This categorization distinguishes unary and binary spatial operators with Boolean, scalar, or spatial results. Of particular interest are set operators that compute the union, difference, or intersection of two spatial objects or sets of objects, for which Tomlin's (1990) Map Algebra provides a framework. Map overlays (Frank 1987) are an important application of set operators and a series of efficient operators have been proposed (Saalfeld 1991, Finke and Hinrichs 1993). Notorious problems regarding these operators, however, include the lack of a closure property and the handling of boundary phenomena. The efficient computation of set operators has received a lot of attention in the computational geometry literature (Preparata and Shamos 1985, Edelsbrunner 1987).

The efficient computation of spatial operators requires special implementations of spatial data types. Over the years, the shortcomings of some more primitive representations have been recognized and semantically more powerful methods have been developed. The early representations in terms of vertex lists have been typically replaced by representations that better capture topological properties. A vertex list is a list of a polygon's vertices. It is sufficient for basic graphic output and well suited to support certain similarity operators, such as translation, because it corresponds to the addition of the translation vector to each of the coordinates. Problems with this particular representation, however, arise when comparing polygons because the list is not unique. For example, the same triangle could be described by the lists [(1,1), (5,1), (4,4)], [(5,1), (4,4), (1,1)], [(4,4), (1,1), (5,1)], or [(2.5,2.5), (1,1), (5,1), (4,4)]. Furthermore, there are no invariants with respect to set operations; therefore, translations, rotations, or scalings change each element of the representation. This also means that it is difficult to determine whether two vertex lists represent congruent or similar polygons. Most critical is the introduction of redundancy if two or more lines or polygons coincide in one or more points. The coincidence is only captured through the common (identical) coordinate values, which provides significant problems during updates. For instance, any consistent update of the boundary of two neighbouring land parcels becomes a task of finding all vertex lists that contain a particular coordinate pair, and at the outset it is unclear how many such occurrences there are. Another data structure with similar deficiencies is the representation of a line as a 4-dimensional point (and a polygon as a list of such lines). While access methods and indices can be designed to cluster and retrieve such stored elements efficiently, the loss in the semantics is significant. The reliance on coordinates to determine identity has been found to be a particular fallacy as geometric transformations over the finite number systems of computers often violate basic assumptions about geometry (Franklin 1984).

To store the geometric data structures, most commercial (relational) databases provide long fields (also called binary large objects or memo fields) that serve as simple containers. One of the columns in the relation is declared to have variable length. The geometric representation is then stored in such a long field in a way that only the application programs can interpret, while the database system itself usually cannot decode the representation. It is, therefore, impossible to formulate or process SQL queries against that column. This long field approach complies with the OpenGIS Simple Features Specifications proposed by the Open GIS Consortium.

Abstract data types (ADTs) provide a more robust way to integrate complex types into a database system. The basic idea is to encapsulate the implementation of a data type in such a way that one can communicate with instances of the data type only through a set of well-defined operators. The internal implementation of the data type and its operators are hidden to any external users, who have no way to review or modify those interior features. Object-oriented and object-relational databases systems use the concept of the abstract data type for defining the structure of object classes. A class is a collection of objects of the same abstract data type. They thus all have the same structure and behaviour as they share the same operations. Classes support two basic concepts underlying abstract data types: abstraction and encapsulation. An object can only be accessed through the operators defined on its class, that is, it is only characterized through its behaviour. The user is prevented from applying unsuitable operators to the object, and its internal representation is hidden. Operators (methods) and attributes are attached to a class, which means that they are valid for all objects that belong to it. Classes may form an inheritance hierarchy. This means that all attributes and methods of a class apply to its subclasses as well, unless they are explicitly overwritten. Object-oriented concepts can easily be adapted to the implementation of spatial data types and operators (Günther and Lamberts 1994).

3.3. *Spatial access methods*

Retrieval queries on a spatial database often require the fast execution of geometric search operations, such as point or range queries or spatial joins. Of particular concern is here the often massive spatial data sets that need to be searched. Early proposals for multidimensional data structures, such as the K-D-tree (Bentley 1975) or the quadtree (Samet 1984, 1989a, b), focused on memory-resident data and, therefore, did not take secondary storage management explicitly into account. Despite the growing size of available RAM, GIS applications are typically disk resident as the size of the datasets is still too large to be stored entirely in RAM. Sequential search is unacceptably slow for most spatial databases; therefore, spatial search operators need special support at the physical level to guarantee good performance for spatial query processing, particularly as the size of a database grows. Traditional databases, however, lack explicit support for searching spatially. To support efficient spatial search, one needs special multidimensional access methods.

The access methods designed with secondary storage management in mind allow their operations to be closely coordinated with the operating system to ensure that overall performance is optimized. Of importance in the design of spatial access methods is the physical organization of storage devices and the goal to minimize the number of operations to secondary storage. A common assumption is that most spatial searches are I/O-bound rather than CPU-bound. Since CPU performance continues to increase at a rate much faster than disk access time, it is likely that in the future spatial access methods will depend even more on I/O. Applications with objects of complex shapes, however, may incur major CPU costs for the refinement steps necessary to filter data retrieved, thereby changing the balance with I/O (Gaede 1995a, Hoel and Samet 1995).

Today's secondary storage devices are linearly structured. The main problem for the design of spatial access methods is that there exists no total ordering among multi-dimensional spatial objects that would preserve their spatial proximity. Most spatial queries (Frank 1981, 1991a) and interesting geographic configurations (Tobler 1970) are related to the neighbourhood of a specific phenomenon; therefore, it is detrimental that there exists no mapping from a two- or higher-dimensional space onto a one-dimensional space such that any two objects that are spatially close in the higher-dimensional space are also close to each other in the one-dimensional sorted sequence. This makes the design of efficient access methods in the spatial domain much more difficult than in traditional databases, where a broad range of efficient and well-understood access methods is available.

One-dimensional access methods, such as linear hashing (Larson 1980, Litwin 1980), extendible hashing (Fagin *et al.* 1979), and the B-tree (Bayer and McCreight 1972, Comer 1979), are an important foundation for multidimensional access methods. A natural approach to handle multidimensional search queries consists in the consecutive application of such single key structures, one per dimension. This approach can be inefficient (Kriegel 1984), however, since each index is traversed independently of the others without exploiting the possibly high selectivity in one dimension for narrowing down the search in the remaining dimensions. Another interesting approach is to extend hashing by using a hash function that takes a d-dimensional vector as argument. A structure based on this idea is the grid file (Nievergelt *et al.* 1984). Unfortunately, this approach suffers from possibly superlinear directory growth.

There is a great variety of requirements that multidimensional access methods should meet, based on the properties of spatial data and their applications (Robinson 1981, Lomet and Salzberg 1989, Frank 1991a):

- *Dynamics*: As data objects are inserted and deleted from the database in any given order, access methods should continuously keep track of the changes.
- *Secondary/tertiary storage management*: Despite growing main memories, it is often not possible to hold the complete database in main memory. Access methods therefore need to integrate secondary and tertiary storage in a seamless manner.
- *Broad range of supported operations*: Access methods should not support just one particular type of operation (such as retrieval) at the expense of other tasks (such as deletion).
- *Independence of the input data*: Access methods should maintain their efficiency even when the input data are highly skewed. This point is especially important for data that are distributed differently along the various dimensions.
- *Simplicity*: Intricate access methods with special cases are often error-prone to implement and thus not sufficiently robust to be used in large-scale applications.

- *Scalability*: Access methods should adapt well to growth in the underlying database.
- *Time efficiency*: Spatial searches should be fast.
- *Space efficiency*: An index should be small in size compared to the size of the data set.
- *Concurrency and recovery*: In modern databases where multiple users concurrently update, retrieve, and insert data, access methods should provide robust techniques for transaction management without significant performance penalties.
- *Minimum impact*: The integration of an access method into a database system should have minimum impact on existing parts of the system.

A common approach to meet these requirements consists of a two-step process: (1) choosing an approximation (e.g., a simpler shape, such as a bounding rectangle) that can be indexed and serves as a fast filter and (2) using the original geometry to assert the retrieval condition only for the initially retrieved objects to eliminate false hits. An index may only administer the MBR (minimum bounding rectangle) of each object, together with a pointer to the description of the object's database entry (the object ID). With this design, the index only produces a set of candidate solutions. This step is therefore termed the filter step. For each element of that candidate set we have to decide whether the MBR is sufficient to decide that the actual object must indeed satisfy the search predicate. In those cases, the object can be added directly to the query result. However, there are often cases where the MBR does not prove to be sufficient. In a refinement step we then have to retrieve the exact shape information from secondary memory and test it against the predicate. If the predicate evaluates to true, the object is added to the query result as well, otherwise we have a false drop.

Spatial access methods have been among the most extensively investigated research areas in computational implementations for GIS. Details about the large variety of methods, and the often subtle differences, are given by Samet (1989b) and Gaede and Günther (1998). Research in this area has been theoretical as well as experimental, typically with a straightforward hypothesis that the new access method requires less disk accesses or simply runs faster than a subset of previously developed methods. While the differences can be measured, even for experts it has become increasingly difficult to recognize the pros and cons of each access structure, because every new method seems to claim better theoretical or empirical performance than at least one other access method that has been published previously. There is no lack of experimental and theoretical studies that analyse and compare the performance of many of the access methods; however, at present no access method has proven itself to be superior to all its competitors in whatever sense. Even if one benchmark declares one structure to be the clear winner, another benchmark may prove the same structure to be inferior. A key question is how generalizable the results are. More complexly structured data often lead to significantly different performance figures. Often also variations in distribution or density affect how suitable a particular method is. Both time and space efficiency of an access method strongly depend on the data to be processed and the queries to be answered. An access method that performs reasonably well for rectangles with the same orientation may fail for arbitrarily oriented lines. Strongly correlated data may render an otherwise fast access method irrelevant for any practical application. An index that has been optimized for point queries may be highly inefficient for arbitrary region queries. Large numbers of insertions and deletions may degrade a structure that is efficient in a more static environment. Initiatives to set up standardized testbeds for benchmarking and comparing access methods under different conditions are important steps in the right direction (Kriegel *et al.* 1989).

Commercial products have resorted to access methods that are easy to understand and implement. Typical examples are quadtrees (Samet 1989a, b) in Oracle 8, SICAD, and Smallworld GIS; R-trees (Guttman 1984) in the relational database system Informix; and z-ordering (Orenstein 1986), which was adapted and integrated under the term HHCODE (Varma *et al.* 1990) into Release 7.3 of Oracle. Performance seems to be of secondary importance for the selection, which comes as no surprise given the relatively small differences among methods in virtually all published analyses. Simple and robust methods are preferred, which can be tuned and tightly integrated with other system components.

4. Looking forward

As with any forecasting, it is difficult and risky to predict where future computational methods for representing geographical concepts will lead. We can only observe trends that are either underway or have recently started. A complementary approach is a discussion of what aspects would benefit from further research because they have not yet been developed sufficiently.

Since the early 1990s, we have seen a growing influence of cognitive considerations on the next generation of computational methods (Mark and Frank 1991). In the past, often hardware aspects were driving the development of computational methods. In the future, the semantics of spatial information need to be addressed. A second motivation for new research in computational methods is the continuing technological push with new information technologies that need to be integrated with traditional GIS functionalities. Miniature GPS receivers, cellular phones, and other wireless communication devices will contribute to making GIS a mobile technology with the potential for the development of novel Spatial Information Appliances (Egenhofer and Kuhn 1998). They will contribute to larger spatial data collections, which will get offered through the Web, some commercially, some for free.

The need for research in computational methods for representing geographical phenomena has been recently emphasized as part of several workshops at the US National Science Foundation. The reports on Critical Research Issues in Geographic Information Science and the NSF Digital Government Initiative include complementary, and at times overlapping, discussions with this viewpoint.

4.1. *Fields*

Most data models of today's GIS follow the cartographic tradition of organizing spatial data as points, lines, and polygons. Since this approach facilitates the creation of good quality graphics in the form of maps, it has almost become a synonym for GIS. Significant amounts of geographical information, however, do not match easily with this model and complementary alternatives need to be investigated.

Fields in GIS (Goodchild 1992) are characterized by values at locations, have a continuous distribution of values of a domain, an inability to be completely measured, and a need for approximation (through interpolation or functions). Since they expose distinct ontological foundations and properties (Couclelis 1992), they lead to different data structures and implementations (Peuquet 1988). The current imbalance between models for discrete and continuous geographical phenomena needs to be overcome by truly interoperating GIS, allowing users to perform analyses beyond the limitations of a single spatial conceptualization. While object representations have reached a level of maturity, the lack of a similar level of formalization, compatibility, and general acceptance is a major impediment. A tight integration with database systems is also necessary, and provides new challenges through the need to retrieve derived (i.e. interpolated) values rather than stored values. The research needs can be broadly structured into four domains (Peuquet *et al.* 1999):

- ontological perspectives of fields,
- cognitive aspects of fields,
- definitions of operations on fields, and
- formalization of fields.

4.2. *Representations of qualitative spatial information*

Today's GIS capture geographical information at the geometric level in quantitative terms. One needs to know about an object's location, extent, and shape in order to record it in a GIS. It has been recognized, however, that a purely quantitative approach does not match human cognition and spatial reasoning (Kuipers 1978). For example, mapping biologists' narrative descriptions of geographical locations (Futch *et al.* 1992) into a Cartesian coordinate space is a struggle. The properties of such a setting are different from the traditional approach—small sets of symbols on an ordinal and nominal scale in a discrete space versus quantitative calculations in an infinitely precise, continuous space (de Kleer and Brown 1984).

Representations of qualitative spatial information are needed to deal with partial information, which is particularly important for spatial applications when only incomplete data sets are available. Natural language descriptions and discourse are typical examples. Neither Cartesian coordinates nor pictorial representations are adequate. Foundations for the representation of qualitative spatial information have been developed in Artificial Intelligence (Glasgow and Papadias 1992, Hernández 1994) with the primary focus on qualitative spatial relations and their inferences (Frank 1992, Freksa 1992, Papadias and Sellis 1992, Smith and Park 1992, Egenhofer and Sharma 1993). New challenges are:

- inferences across different types of qualitative spatial information through interoperability across different models,
- integration of qualitative and quantitative spatial reasoning methods, and
- linkage between the semantics of natural-language terminology and models for qualitative spatial information.

4.3. *Temporal aspects*

Most of today's computational methods in GIScience treat geographical phenomena as static. A variety of conceptual models for time in GIS have been studied (Hazelton 1991, Kelmelis 1991, Langran 1992, Al-Taha 1992), but to date little impact has been made on commercially available tools. The linkage between space and time requires, among others, the modelling of different types of times (Frank 1998), the incorporation of processes (Peuquet 1994, Peuquet and Wentz 1994), and the most fundamental aspects of change (Hornsby and Egenhofer 1997).

Temporal databases (Snodgrass 1992) and temporal reasoning (Allen 1983) have a long tradition in computer science, but the integration with spatial databases (http://www.dbnet.ece.ntua.gr/~choros/) and spatial reasoning (Egenhofer and Golledge 1998) is only in its infancy. With the organization of several workshops, however, this research area has developed significant momentum.

A promising approach is the focus on a set of particular spatial phenomena that share the same space-time behaviour. An ESP workshop focused on Life and Motion of Socio-Economic Units (Frank *et al.* 1999) and recently others have placed emphasis on continuously moving, point-like objects (Güting *et al.* 1998).

4.4. *Knowledge discovery in massive spatial data sets*

Massive amounts of spatial data are being collected, either now or in the near future. New technologies will lead to ever increasing sizes of data sets, at greater levels of spatial and temporal detail. Already in place are plans for EOS, which will generate several terabytes a day of remotely sensed imagery. In addition, the integration of GPS receivers into a large variety of spatial appliances will lead to massive records about movements of people and objects at high levels of temporal resolution. In a similar way, airborne or terrestrial video will soon become a vast data source, providing near-continuous coverage of selected activities. Such new, highly detailed, and massive data sources have the potential of enabling the scientists to perform novel types of analyses.

Such large datasets are beyond the comprehension of a single person and computational methods are indispensable to discover new knowledge. Such knowledge may be about recurring spatial and spatio-temporal patterns, clusters, associations, outliers, or anomalies that characterize interesting situations (Koperski and Han 1995). Effective spatial data mining methods need to be coupled with efficient algorithms that schedule the processing of very large, possibly distributed data sources (Ng and Han 1994).

4.5. *GIS and database systems*

Modern database technology is essential for the efficient handling of geographical data. For the necessary integration of GIS and modern database technology, there are essentially four options:

- Extension of an existing GIS with database functionalities. Most GISs in the 1980s and early 1990s were representative of this approach.
- Coupling of a GIS with a commercial DBMS. Such a coupling has been common for some time for storing the non-spatial data in a commercial relational database. Since the mid-1990s, many vendors started to store spatial data in a relational DBMS as well. Usually, the relations just serve as containers that manage the geometries as unstructured long fields.
- Extension of an existing DBMS with spatial functionalities (van Oosterom and Vijlbrief 1991, Scholl and Voisard 1992). Recent commercial products include Illustra's DataBlades and Oracle's Cartridges, which package domain-specific extensions in modules.
- Open toolbox approaches that see a GIS just as a collection of specialized services in an electronic marketplace. While there is currently no commercial system that strictly follows this architecture, many vendors are starting to integrate similar ideas into their products.

Due to strong customer pressure, the trend towards such open GIS (fourth option) continues to increase significantly. Commercial database systems can be integrated into open architectures in a relatively simple manner. A GIS can thus gain directly from the traditional strengths of a modern database system, including an SQL query facility, persistence, transaction management, and distribution. Most importantly, however, more openness would facilitate the integration of GIS with mainstream business software. While data analysis has always been an important part of GIS, the breadth and depth of related work has increased considerably since the early 1990s. By extending their functional spectrum beyond the traditional domains of data capture, storage, and visualization, GIS is gradually moving into the mainstream of computing. Rather than providing support just for the geosciences, GIS vendors are trying to position their products as spatial data management components that should be a part of just about any information system architecture—simply because just about any information has a spatial aspect. Interfaces to business software such as Microsoft Office or SAP's R/3, and the development of spatial decision support systems (Densham *et al.* 1995) are among the most visible signs of this trend.

In order to achieve these ambitious goals, GIS vendors have to provide data analysis capabilities that go far beyond simple map overlays. Moreover, they have to package their modules in a manner that allows the easy integration of selected functionalities into a given business package. The new eXtended Markup Language XML may play an important role in this integration process. The resulting shift from GIS to GIServices (Günther and Müller 1999) is one of the great challenges for the next decade.

5. Conclusions

There is a need to continue to improve the foundations of computational methods, advancing them to the next level of sophistication. For example, new multi-media data types are becoming available, and we need computational methods to extract geographical content. Fresh approaches, particularly those based on cognitive considerations, should be pursued rather than making small increments to established algorithms and data structures. The most dramatic effect of such approaches will be on the user interfaces of geographical information systems. In lieu of training people, GIS user interfaces need to be made more intuitive, providing also better integration into the problem solving process. We recognize that this subfield of GIScience relies on researchers from diverse backgrounds, and close interactions are needed to make significant progress. Such interactions need to span academia, industry, and government to address the users' needs, account for technological advancements, and enable technology transfer. An important economic factor is the high demand in industry for people with knowledge in computational GIS methods.

The scope for GIS applications is broad for the future. Only through the design, development, and evaluation of complex real-world systems will we realize the full potential of computation for GIS. Some of the more important domains that should be considered for future research are spatial navigation, transportation, environmental modelling, and sales and marketing. GIS research can also benefit from and have impact on other areas of computational science. For example, advanced vision systems contribute to the understanding of satellite data. Intelligent robotics requires the storage and manipulation of geographical information. What we learn from the implementation of GIS can also be transferred to other domains, such as the analysis of molecular spatial databases.

Acknowledgments

In addition to the Varenius project funding from the National Science Foundation, Max Egenhofer's work is further supported by NSF grants IRI-9613646, BDI-9723873 and IIS-9970723, by grants from the National Imagery and Mapping Agency under grant number NMA202-97-1-1023; the Air Force Research Laboratory under grant number F30602-95-1-0042; the National Aeronautics and Space Administration under grant number COE/97-0015; the National Institute of Environmental Health Sciences, NIH, under grant number 1 R 01 ESO9816-01, Bangor Hydro-Electric Co.; by a research contract from GE Research and Development Center; and by a Massive Digital Data Systems contract sponsored by the Advanced Research and Development Committee of the Community Management Staff.

Thanks to Volker Gaede (Oracle, Germany) for his contributions to §3.3.

References

Abel, D., 1988, Relational data management facilities for spatial information systems. In *Proceedings of Third International Symposium on Spatial Data Handling*, edited by D. Marble (Columbus: International Geographical Union), pp. 9–18.

Abel, D., and Ooi, B. C., editors, 1993, *Third International Symposium on Large Spatial Databases, SSD '93* (New York: Springer-Verlag).

Abel, D., and Smith, J., 1986, A relational GIS database accommodating independent partitionings of the region. In *Proceedings of Second International Symposium on Spatial Data Handling*, edited by D. Marble (Columbus: International Geographical Union), pp. 213–224.

Ahlberg, C., and Shneiderman, B., 1994, Visual information seeking: Tight coupling of dynamic query filters with starfield displays. In *Proceedings of CHI 94*, pp. 313–317.

Al-Taha, K., 1992, Temporal Reasoning in Cadastral Systems. Ph.D. Thesis, Department of Surveying Engineering, University of Maine, Orono, ME.

Allen, J., 1983, Maintaining knowledge about temporal intervals. *Communications of the ACM*, **26**, 832–843.

Aref, W., and Samet, H., 1994, The spatial filter revisited. In *Proceedings of Sixth International Symposium on Spatial Data Handling, Edinburgh, Scotland,* edited by T. Waugh and R. Healey (London: Taylor & Francis), pp. 190–208.

Bauzer Medeiros, C., and Pires, F., 1994, Databases for GIS. *SIGMOD Record*, **23**, 107–115.

Bayer, R., and McCreight, E., 1972, Organization and maintenance of large ordered indices. *Acta Informatica*, **1**, 173–189.

Becker, L., and Güting, H., 1992, Rule-based optimization and query processing in an extensible geometric database system. *ACM Transactions on Database Systems*, **17**, 247–303.

Bentley, J., 1975, Multidimensional binary search trees used for associative searching. *Communications of the ACM*, **18**, 509–517.

Berman, R., and Stonebraker, M., 1977, GEO-QUEL, A system for the manipulation and display of geographic data. *ACM Computer Graphics*, **11**, 186–191.

Brinkhoff, T., Kriegel, H.-P., Schneider, R., and Seeger, B., 1994, Multi-step processing of spatial joins. *ACM SIGMOD International Conference on Management of Data*, **23**, 197–208.

Brinkhoff, T., Kriegel, H.-P., and Seeger, B., 1993, Efficient processing of spatial joins using R-trees. *SIGMOD Record*, **22**, 237–246.

Bruns, T., and Egenhofer, M., 1997, User interfaces for map algebra. *Journal of the Urban and Regional Information Systems Association*, **9**, 44–54.

Buchmann, A., Günther, O., Smith, T., and Wang, Y., editors, 1989, *Symposium on the Design and Implementation of Large Spatial Databases, Santa Barbara, CA*, Lecture Notes in Computer Science 409 (New York: Springer-Verlag).

Chen, L., Drach, R., Keating, M., Louis, S., Rotem, D., and Shoshani, A., 1995, Access to multidimensional datasets on tertiary storage systems. *Information Systems*, **20**, 155–183.

Comer, D., 1979, The ubiquitous B-tree. *ACM Computing Surveys*, **11**, 121–138.

Couclelis, H., 1992, People manipulate objects (but cultivate fields): beyond the raster-vector debate in GIS. In *Theories and Methods of Spatio-Temporal Reasoning in Geographic Space*, Lecture Notes in Computer Science 639, edited by A. U. Frank, I. Campari and U. Formentini (New York: Springer-Verlag), pp. 65–77.

Cox, N., Aldred, B., and Rhind, D., 1979, A relational data base system and a proposal for a geographic data type. *Geo-Processing*, **1**, 217–229.

de Kleer, J., and Brown, J., 1984, A qualitative physics based on confluences. *Artificial Intelligence*, **24**, 7–83.

Densham, P., Armstrong, M., and Kemp, K., 1995, *Collaborative Spatial Decision-Making: Scientific Report for the Initiative 17 Specialist Meeting*, Technical Report 95-14 (Santa Barbara, CA: National Center for Geographic Information and Analysis).

Edelsbrunner, H., 1987, *Algorithms in Combinatorial Geometry* (New York: Springer-Verlag).

Egenhofer, M., 1990, Manipulating the graphical representation of query results in geographic information systems. *Proceedings of Workshop on Visual Languages (Skokie, IL)*, pp. 119–124.

Egenhofer, M., 1991, Extending SQL for cartographic display. *Cartography and Geographic Information Systems*, **18**, 230–245.

Egenhofer, M., 1992, Why not SQL! International Journal of Geographical Information Systems, **6**, 71–85.

Egenhofer, M., 1994, Spatial SQL: A query and presentation language. *IEEE Transactions on Knowledge and Data Engineering*, **6**, 86–95.

Egenhofer, M., and Frank, A., 1987, Object-oriented databases: Database requirements for GIS. In *Proceedings of International Geographic Information Systems (IGIS) Symposium: The Research Agenda*, pp. 189–211.

Egenhofer, M., and Frank, A., 1988, Towards a spatial query language: User interface considerations. In D. DeWitt and F. Bancilhon, editors, *Proceedings of 14th International Conference on Very Large Data Bases, Los Angeles, CA*, edited by D. DeWitt and F. Bancilhon, pp. 124–133.

Egenhofer, M., and Frank, A., 1992, Object-oriented modeling for GIS. *Journal of the Urban and Regional Information Systems Association*, **4**, 3–19.

Egenhofer, M., Frank, A., and Jackson, J., 1989, A topological data model for spatial databases. In *Symposium on the Design and Implementation of Large Spatial Databases*, Lecture Notes in Computer Science 409, edited by A. Buchmann, O. Günther, T. Smith, and Y. Wang (New York: Springer-Verlag), pp. 271–286.

Egenhofer, M., and Franzosa, R., 1991, Point-set topological spatial relations. *International Journal of Geographical Information Systems*, **5**, 161–174.

Egenhofer, M., and Golledge, R., editors, 1998, *Spatial and Temporal Reasoning in Geographic Information Systems* (New York: Oxford University Press).

Egenhofer, M., and Herring, J., editors, 1995, *Advances in Spatial Databases—4th International Symposium, SSD '95, Portland, ME,* Lecture Notes in Computer Science 951 (Berlin: Springer-Verlag).

Egenhofer, M., and Kuhn, W., 1998, Beyond desktop GIS. *Proceedings of GIS PlaNET, Lisbon, Portugal.*

Egenhofer, M., and Kuhn, W., 1999, Interacting with GIS. In *Geographical Information Systems: Principles, Techniques, Applications and Management*, edited by P. Longley, M. Goodchild, D. Maguire and D. Rhind (New York: Wiley), pp. 401–412.

Egenhofer, M., and Sharma, J., 1993, Assessing the consistency of complete and incomplete topological information. *Geographical Systems*, **1**, 47–68.

Fagin, R., Nievergelt, J., Pippenger, N., and Strong, R., 1979, Extendible hashing: a fast access method for dynamic files. *ACM Transactions on Database Systems*, **4**, 315–344.

Fernandez, A., Dinn, A., Paton, N., Williams, M., and Liew, O., 1999, Extending a deductive object-oriented database system with spatial data handling facilities. *Information and Software Technology*, **11**, 483–497.

Finke, U., and Hinrichs, K., 1993, A spatial data model and a topological sweep algorithm for map overlay. In *Third International Symposium on Large Spatial Databases*, Lecture

Notes in Computer Science 692, edited by D. Abel and B. Ooi (New York: Springer-Verlag), pp. 162–177.

Frank, A., 1981, Applications of DBMS to land information systems. In C. Zaniolo and C. Delobel, editors, *Proceedings of Seventh International Conference on Very Large Data Bases*, pp. 448–453.

Frank, A., 1982, MAPQUERY—Database query language for retrieval of geometric data and its graphical representation. *ACM Computer Graphics*, **16**, 199–207.

Frank, A., 1984, Extending a database with Prolog. In *First International Workshop on Expert Database Systems, Kiawah Island, SC*, edited by L. Kerschberg, pp. 665–676.

Frank, A., 1987, Overlay processing in spatial information systems. In *AUTO-CARTO 8, Eighth International Symposium on Computer-Assisted Cartography*, edited by N. Chrisman (Bethseda, MD: American Congress on Surveying and Mapping), pp. 16–31.

Frank, A., 1988, Requirements for a database management system for a GIS. *Photogrammetric Engineering & Remote Sensing*, **54**, 1557–1564.

Frank, A., 1991a, Properties of geographic data: requirements for spatial access methods. In *Advances in Spatial Databases—Second Symposium, SSD '91*, Lecture Notes in Computer Science 525, edited O. Günther and H. J. Schek (New York: Springer-Verlag), pp. 225–234.

Frank, A., 1991b, Qualitative spatial reasoning about cardinal directions. In *Proceedings of Autocarto 10, Baltimore, MD*, edited D. Mark and D. White, (Bethesda: American Congress on Surveying and Mapping), pp. 148–167.

Frank, A., 1992, Qualitative spatial reasoning about distances and directions in geographic space. *Journal of Visual Languages and Computing*, **3**, 343–371.

Frank, A., 1998, Different types of 'times' in GIS. In *Spatial and Temporal Reasoning in Geographic Information Systems*, edited by M. Egenhofer and R. Golledge (New York: Oxford University Press), pp. 40–62.

Frank, A., and Campari, I., editors, 1993, *Spatial Information Theory, European Conference, COSIT '93*, Lecture Notes in Computer Science 716 (New York: Springer-Verlag).

Frank, A., and Kuhn, W., 1986, Cell Graph: A provable correct method for the storage of geometry. In *Proceedings of Second International Symposium on Spatial Data Handling, Seattle, WA*, edited by D. Marble (Columbus: International Geographical Union), pp. 411–436.

Frank, A., and Kuhn, W., editors, 1995, *Spatial Information Theory—A Theoretical Basis for GIS, International Conference COSIT '95, Semmering, Austria*, Lecture Notes in Computer Science 988 (Berlin: Springer-Verlag).

Frank, A., Raper, J., and Cheylan, J.-P., 1999, *Life and Motion of Socio-Economic Values* (London: Taylor & Francis).

Franklin, W. R., 1984, Cartographic errors symptomatic of underlying algebra problems. In *International Symposium on Spatial Data Handling, Zurich, Switzerland,* edited by K. Brassel (Zurich: International Geographical Union), pp. 190–208.

Freksa, C., 1992, Using orientation information for qualitative spatial reasoning. In *Theories and Methods of Spatio-Temporal Reasoning in Geographic Space*, Lecture Notes in Computer Science 639, edited by A. Frank, I. Campari, and U. Formentini (New York: Springer-Verlag), pp. 162–178.

Frew, J., 1994, Bigfoot: An Earth science computing environment for the Sequoia 2000 project. In *Environmental Information Management and Analysis: Ecosystem to Global Scales*, edited by W. Michener, J. Brunt and S. Stafford (London: Taylor & Francis), pp. 113–125.

Futch, S., Chin, D., McGranaghan, M., and Lay, J.-G., 1992, Spatial-linguistic reasoning in LEI (Locality and Elevation Interpreter). In *Theories and Methods of Spatio-Temporal Reasoning in Geographic Space*, Lecture Notes in Computer Science 639, edited by A. Frank. I. Campari and U. Formentini (New York: Springer-Verlag), pp. 318–327.

Gaede, V., 1995a, Geometric information makes spatial query processing more efficient. In *Proceedings of Third International Workshop on Advances in Geographic Information Systems (ACM-GIS '95), Baltimore, MD*, edited by P. Bergougnoux, K. Makki, and N. Pissinou, pp. 45–52.

Gaede, V., 1995b, Optimal redundancy in spatial database systems. In *Advances in Spatial Databases—4th International Symposium, SSD '95, Portland, ME*, edited by M. Egenhofer and J. Herring, Lecture Notes in Computer Science 951 (Berlin: Springer-Verlag), pp. 96–116.

GAEDE, V., and GÜNTHER, O., 1998, Multidimensional access methods. *ACM Computing Surveys*, **30**, 170–231.

GLASGOW, J., and PAPADIAS, D., 1992, Computational imagery. *Cognitive Science*, **16**, 355–394.

GO, A., STONEBRAKER, M., and WILLIAMS, C., 1975, *An Approach to Implementing a Geo-Data System*. Memo ERL-M529, Electronics Research Laboratory, University of California.

GOODCHILD, M., 1992, Geographical data modeling. *Computers and Geosciences*, **18**, 401–408.

GRAY, J., and REUTER, A., 1993, *Transaction Processing: Concepts and Techniques* (San Mateo, CA: Morgan Kaufmann).

GÜNTHER, O., 1993, Efficient computation of spatial joins. *Proceedings of Ninth International Conference on Data Engineering, Vienna, Austria,* pp. 50–59.

GÜNTHER, O., 1998, *Environmental Information Systems* (Berlin: Springer-Verlag).

GÜNTHER, O., and BUCHMANN, A., 1990, Research issues in spatial databases. *SIGMOD Record*, **19**, 61–68.

GÜNTHER, O., and LAMBERTS, J., 1994, Object-oriented techniques for the management of geographic and environmental data. *The Computer Journal*, **37**, 16–25.

GÜNTHER, O., and MÜLLER, R., 1999, From GISystems to GIServices: spatial computing on the Internet marketplace. In *Interoperating Geographic Information Systems*, edited by M. Goodchild, M. Egenhofer, R. Fegeas and C. Kottman (Norwell, MA: Kluwer), pp. 427–442.

GÜNTHER, O., and SCHEK, H.-J., editors, 1991, *Advances in Spatial Databases—Second Symposium, SSD '91, Zurich, Switzerland*, Lecture Notes in Computer Science 525 (New York: Springer-Verlag).

GÜTING, R., 1988, Geo-relational algebra: a model and query language for geometric database systems. In *Advances in Database Technology—EDBT '88 International Conference on Extending Database Technology, Venice, Italy*, Lecture Notes in Computer Science 303, edited by J. Schmidt, S. Ceri and M. Missikoff (New York: Springer-Verlag), pp. 506–527.

GÜTING, R., 1989, Gral: An extensible relational database system for geometric applications. In *Proceedings of 15th International Conference on Very Large Data Bases, Amsterdam*, edited by P. Apers and G. Wiederhold, pp. 33–44.

GÜTING, R., 1994a, editor, *VLDB Journal: Special Issue on Spatial Database Systems*, **3**.

GÜTING, R., 1994b, An introduction to spatial database systems. *VLDB Journal*, **3**, 357–399.

GÜTING, R., BOEHLEN, M., ERWIG, M., JENSEN, C., LORENTZOS, N., SCHNEIDER, M., and VAZIRGIANNIS, V., 1998, *A Foundation for Representing and Querying Moving Objects*, Informatik-Report 238, FernUniversität, Hagen, Germany.

GÜTING, R., and SCHNEIDER, M., 1993, Realms: A foundation for spatial data types in database systems. In *Third International Symposium on Large Spatial Databases*, Lecture Notes in Computer Science 692, edited by D. Abel and B. Ooi (New York: Springer-Verlag), pp. 14–35.

GÜTING, R., DE RIDDER, T., and SCHNEIDER, M., 1995, Implementation of the ROSE algebra: efficient algorithms for Realm-based spatial data types. In *Advances in Spatial Databases—4th International Symposium, SSD '95, Portland, ME*, Lecture Notes in Computer Science 951, edited by M. Egenhofer amd J. Herring (Berlin: Springer-Verlag), pp. 216–239.

GUTTMAN, A., 1984, R-trees: A dynamic index structure for spatial searching. In B. Yormark, editor, *Proceedings of Annual Meeting ACM SIGMOD*, pp. 47–57.

HAAS, L., CHANG, W., LOHMAN, G., MCPHERSON, J., WILMS, P., LAPIS, G., LINDSAY, B., PIRAHESH, H., CAREY, M., and SHEKITA, E., 1990, Starburst mid-flight: As the dust clears. *IEEE Transactions on Knowledge and Data Engineering*, **2**, 143–161.

HÄRDER, T., and REUTER, A., 1985, Architecture of database systems for non-standard applications (in German). In *Database Systems in Office, Engineering, and Science*, Informatik Fachberichte 94, edited by A. Blaser and P. Pistor (New York: Springer-Verlag), pp. 253–286.

HAZELTON, N. W. J., 1991, Integrating Time, Dynamic Modelling, and Geographical Information Systems: Development of Four-Dimensional GIS. Ph.D. Thesis, Department of Surveying and Land Information, The University of Melbourne, Melbourne, Australia.

HERNÀNDEZ, D., 1994, *Qualitative Representation of Spatial Knowledge*, Lecture Notes in Artificial Intelligence 804 (New York: Springer-Verlag).

HERNÀNDEZ, D., CLEMENTINI, E., and DI FELICE, P., 1995, Qualitative distances. In *Spatial Information Theory—A Theoretical Basis for GIS, International Conference COSIT '95,* *Semmering, Austria*, Lecture Notes in Computer Science 988, edited by A. Frank and W. Kuhn (Berlin: Springer-Verlag), pp. 45–57.

Herring, J., 1991, The mathematical modeling of spatial and non-spatial information in geographic information systems. In *Cognitive and Linguistic Aspects of Geographic Space*, edited by D. Mark and A. Frank (Dordrecht: Kluwer Academic Publishers), pp. 313–350.

Herring, J., Larsen, R., and Shivakumar, J., 1988, Extensions to the SQL language to support spatial analysis in a topological data base. *Proceedings of GIS/LIS '88*, *San Antonio, TX*, pp. 741–750.

Hirtle, S., and Frank, A., editors, 1997, *Spatial Information Theory—A Theoretical Basis for GIS, International Conference COSIT '97, Laurel Highlands, PA*, Lecture Notes in Computer Science **1329** (Berlin: Springer-Verlag).

Hoel, E., and Samet, H., 1995, Benchmarking spatial join operations with spatial output. In *Proceedings of 21st International Conference on Large Spatial Data Bases, Zurich, Switzerland*, edited by U. Dayal, P. Gray and S. Nishio, pp. 606–618.

Hornsby, K., and Egenhofer, M., 1997, Qualitative representation of change. In *Spatial Information Theory—A Theoretical Basis for GIS, International Conference COSIT '97, Laurel Highlands, PA*, Lecture Notes in Computer Science 1329, edited by S. Hirle and A. Frank(Berlin: Springer-Verlag), pp. 15–33.

Ingram, K., and Phillips, W., 1987, Geographic information processing using a SQL-based query language. In *Proceedings of AUTO-CARTO 8, Eighth International Symposium on Computer-Assisted Cartography, Baltimore, MD,* edited by N. Chrisman (Bethesda: American Congress on Surveying and Mapping), pp. 326–335.

Kainz, W., Egenhofer, M., and Greasly, I., 1993, Modelling spatial relations and operations with partially ordered sets. *International Journal of Geographical Information Systems*, **7**, 214–229.

Kelmelis, J., 1991, Time and Space in Geographic Information: Toward a Four-Dimensional Spatio-Temporal Data Model. Ph.D. Thesis, Department of Geography, Pennsylvania State University.

Kemper, A., and Wallrath, M., 1987, An analysis of geometric modeling in database systems. *ACM Computing Surveys*, **19**, 47–91.

Koperski, K., and Han, J., 1995, Discovery of spatial association rules in geographic information databases. In *Advances in Spatial Databases—4th International Symposium, SSD '95, Portland, ME*, Lecture Notes in Computer Science 951, edited by M. Egenhofer and J. Herring (Berlin: Springer-Verlag), pp. 47–66.

Kottman, C., 1999, Introduction. In *Interoperating Geographic Information Systems,* edited by M. Goodchild, M. Egenhofer, R. Fegeas and C. Kottman (New York: Kluwer).

Kraak, M.-J., and Molenaar, M., 1996, *Seventh International Symposium on Spatial Data Handling* (London: Taylor & Francis).

Kriegel, H.-P., 1984, Performance comparison of index structures for multikey retrieval. In *Proceedings of Annual Meeting ACM SIGMOD, Boston, MA*, edited by B. Yormark, pp. 186–196.

Kriegel, H.-P., Schiwietz, M., Schneider, R., and Seeger, B., 1989, Performance comparison of point and spatial access methods. In *Symposium on the Design and Implementation of Large Spatial Databases*, Lecture Notes in Computer Science 409, edited by A. Buchmann, O. Günther, T. Smith and Y. Wang (New York: Springer-Verlag), pp. 89–114.

Kuipers, B., 1978, Modeling spatial knowledge. *Cognitive Science*, **2**, 129–153.

Langran, G., 1992, *Time in Geographic Information Systems* (London: Taylor & Francis).

Larson, P. A., 1980, Linear hashing with partial expansions. *6th International Conference on Very Large Data Bases*, pp. 224–232.

Laurini, R., and Thompson, D., 1992, *Fundamentals of Spatial Information Systems* (London: Academic Press).

Litwin, W., 1980, Linear hashing: a new tool for file and table addressing. *Proceedings of 6th International Conference on Very Large Data Bases, Montreal, Canada*, pp. 212–223.

Lomet, D., and Salzberg, B., 1989, A robust multiattribute search structure. *Proceedings of 5th IEEE International Conference on Data Engineering, Los Angeles, CA*, pp. 296–304.

Mark, D., and Frank, A., 1991, *Cognitive and Linguistic Aspects of Geographic Space*, (Dordrecht: Kluwer Academic Publishers).

Morehouse, S., 1985, Arc/Info: A geo-relational model for spatial information. In *Proceedings of Auto-Carto 7*, (Bethesda: American Congress on Surveying and Mapping), pp. 388–397.

Ng, R., and Han, J., 1994, Efficient and effective clustering methods for spatial data mining. In *Proceedings of 20th International Conference on Very Large Data Bases (VLDB '94), Santiago, Chile*, edited by J. Bocca, M. Jarke and C. Zaniolo, pp. 144–155.

Nievergelt, J., Hinterberger, H., and Sevcik, K., 1984, The GRID FILE: an adaptable, symmetric multikey file structure. *ACM Transactions on Database Systems*, **9**, 38–71.

Ooi, B., 1990, *Efficient Query Processing in Geographic Information Systems*, Lecture Notes in Computer Science 471 (New York: Springer-Verlag).

Ooi, B., and Sacks-Davis, R., 1989, Query optimization in an extended DBMS. In *Third International Conference on Foundations of Data Organization and Algorithms (FODO), Paris, France*, Lecture Notes in Computer Science 367, edited by W. Litwin and H. Schek (New York: Springer-Verlag), pp. 48–63.

Ooi, B., Sacks-Davis, R., and McDonell, K., 1989, Extending a DBMS for geographic applications. *Proceedings of IEEE Fifth International Conference on Data Engineering, Los Angeles, CA*, pp. 590–597.

Orenstein, J., 1986, Spatial query processing in an object-oriented database system. *ACM-SIGMOD, International Conference on Management of Data, Washington, DC*, pp. 326–336.

Orenstein, J., 1990, An object-oriented approach to spatial data processing. In *Fourth International Symposium on Spatial Data Handling*, edited by K. Brassel and H. Kishimoto, pp. 820–829.

Papadias, D., and Sellis, T., 1992, Spatial reasoning using symbolic arrays. In *Theories and Methods of Spatio-Temporal Reasoning in Geographic Space*, Lecture Notes in Computer Science 639, edited by A. Frank, I. Campari and U. Formentini, (New York: Springer-Verlag), pp. 153–161.

Peuquet, D., 1988, Representations of geographic space: toward a conceptual synthesis. *Annals of the Association of American Geographers*, **78**, 375–394.

Peuquet, D., 1992, An algorithm for calculating minimum Euclidean distance between two geographic features. *Computers and Geosciences*, **18**, 989–1001.

Peuquet, D., 1994, It's about time: a conceptual framework for the representation of temporal dynamics in geographic information systems. *Annals of the Association of American Geographers*, **84**, 441–461.

Peuquet, D., Smith, B., and Brogaard, B., 1999, *The Ontology of Fields*, Technical Report (Santa Barbara, CA: National Center for Geographic Information and Analysis).

Peuquet, D., and Wentz, E., 1994, An approach for time-based analysis of spatiotemporal data. In *Advances in GIS: Proceedings of Sixth International Symposium on Spatial Data Handling, Edinburgh, Scotland*, edited by T. Waugh and R. Healey (London: Taylor & Francis), pp. 489–504.

Peuquet, D., and Zhan, C.-X., 1987, An algorithm to determine the directional relationship between arbitrarily-shaped polygons in the plane. *Pattern Recognition*, **20**, 65–74.

Preparata, F., and Shamos, M., 1985, *Computational Geometry* (New York: Springer-Verlag).

Retz-Schmidt, G., 1988, Various views on spatial prepositions. *AI Magazine*, **9**, 95–105.

Richards, J., and Egenhofer, M., 1995, A comparison of two direct-manipulation GIS user interfaces for map overlay. *Geographical Systems*, **2**, 267–290.

Robinson, J., 1981, The K-D-B-tree: a search structure for large multidimensional dynamic indices. In *Proceedings of ACM SIGMOD International Conference on Management of Data, Ann Arbor, MI*, edited by Y. E. Lien, pp. 10–18.

Rotem, D., 1991, Spatial join indices. *Proceedings of 7th IEEE Conference on Data Engineering* (Kobe, Japan: IEEE Computer Society), pp. 500 –509.

Roussopoulos, N., Faloutsos, C., and Sellis, T., 1988, An efficient pictorial database system for PSQL. *IEEE Transactions on Software Engineering*, **14**, 630–638.

Saalfeld, A., 1991, An application of algebraic topology to an overlay problem of analytical cartography. *Cartography and Geographic Information Systems*, **18**, 23–36.

Samet, H., 1984, The quadtree and related hierarchical data structures. *ACM Computing Surveys*, **16**, 187–260.

Samet, H., 1989a, *Applications of Spatial Data Structures: Computer Graphics, Image Processing, and GIS* (Reading, MA: Addison-Wesley).

Samet, H., 1989b, *The Design and Analysis of Spatial Data Structures* (Reading, MA: Addison-Wesley).

Schek, H.-J., Paul, H.-B., Scholl, M., and Weikum, G., 1990, The DASDBS project: Objectives, experiences and future prospects. *IEEE Transactions on Knowledge and Data Engineering*, **2**, 25–43.

Schneider, M., 1997, *Spatial Data Types for Database Systems*, Lecture Notes in Computer Science 1288 (Berlin: Springer-Verlag).

Scholl, M., and Voisard, A., 1989, Thematic map modeling. In *Symposium on the Design and Implementation of Large Spatial Databases, Santa Barbara, CA*, Lecture Notes in Computer Science 409, edited by A. Buchmann, O. Günther, T. Smith and Y. Wang (New York: Springer-Verlag), pp. 167–190.

Scholl, M., and Voisard, A., editors, 1992, Geographic applications: an experience with O_2. In *Building an Object-Oriented Database System—The Story of O_2*, edited by F. Bancilhon, C. Delobel and P. Kanellakis (San Mateo, CA: Morgan Kaufmann Publishers), pp. 585–618.

Scholl, M., and Voisard, A., 1997, *Advances in Spatial Databases, 5th International Symposium, SSD '97*, Lecture Notes in Computer Science **1262** (Berlin: Springer-Verlag).

Sellis, T., Roussopoulos, N., and Faloutsos, C., 1987, The R+ -tree: A dynamic index for multi-dimensional objects. In *13th International Conference on Very Large Data Bases*, edited by P. Stocker and W. Kent, pp. 507–518.

Shekhar, S., Coyle, M., Goyal, B., Liu, D.-R., and Sarkar, S., 1997, Data models in geographic information systems. *Communications of the ACM*, **40**, 103–111.

Shneiderman, B., 1983, Direct manipulation: A step beyond programming languages. *IEEE Computer*, **16**, 57–69.

Silberschatz, A., Stonebraker, M., and Ullman, J., 1991, Database systems: Achievements and opportunities. *Communications of the ACM*, **34**, 110–120.

Smith, T., 1996, Alexandria Digital Library. *Communications of the ACM*, **38**, 61–62.

Smith, T., and Frank, A., 1990, Very large spatial databases: Report from the specialist meeting. *Journal of Visual Languages and Computing*, **1**, 291–309.

Smith, T., and Park, K., 1992, Algebraic approach to spatial reasoning. *International Journal of Geographical Information Systems*, **6**, 177–192.

Snodgrass, R., 1992, Temporal databases. In *Theories and Methods of Spatio-Temporal Reasoning in Geographic Space*, Lecture Notes in Computer Science 639, edited by A. Frank, I. Campari and U. Formentini (New York: Springer-Verlag), pp. 22–64.

Stell, J., and Worboys, M., 1998, Stratified map spaces: A formal basis for multi-resolution spatial databases. In *Proceedings of Eighth International Symposium on Spatial Data Handling*, edited by T. Poiker and N. Chrisman, pp. 180–189.

Stone, M., Fishkin, K., and Bier, E., 1994, The movable filter as a user interface tool. *Proceedings of CHI 94*, pp. 306–312.

Stonebraker, M., Frew, J., and Dozier, J., 1993, The Sequoia 2000 project. In *Third International Symposium on Large Spatial Databases*, Lecture Notes in Computer Science 692, edited by D. Abel and B. Ooi (New York: Springer-Verlag), pp. 397–412.

Stonebraker, M., and Rowe, L. A., 1986, The design of POSTGRES. In *Proceedings of ACM-SIGMOD International Conference on the Management of Data*, edited by P. Stocker, W. Kent and P. Hammersley, pp. 340–355.

Tobler, W., 1970, A computer movie simulating urban growth in the Detroit region. *Economic Geography*, **46**, 234–240.

Tomlin, C. D., 1990, *Geographic Information Systems and Cartographic Modeling* (Englewood Cliffs, NJ: Prentice-Hall).

van Oosterom, P., and Vijlbrief, T., 1991, Building a GIS on top of the open DBMS 'Postgres'. *Proceedings of EGIS '91, Brussels, Belgium*, pp. 775–787.

Varma, H., Boudreau, H., and Prime, W., 1990, A data structure for spatio-temporal databases. *International Hydrographic Review*, **67**, 71–92.

Voisard, A., 1995, Mapgets: A tool for visualizing and querying geographic information. *Journal of Visual Languages and Computing*, **6**, 367–384.

Waugh, T., and Healey, R., editors, 1994, *Sixth International Symposium on Spatial Data Handling* (London: Taylor & Francis).

White, M., 1984, Technical requirements and standards for a multipurpose geographic data system. *The American Cartographer*, **11**, 15–26.

Worboys, M., 1996, *GIS: A Computing Perspective* (London: Taylor & Francis).

Worboys, M., Hearnshaw, H., and Maguire, D., 1990, Object-oriented data modelling for spatial databases. *International Journal of Geographical Information Systems*, **4**, 369–383.