A System for Querying With Qualitative Distances in Networks

Carl P. L. Schultz, Hans W. Guesgen, Robert Amor

Abstract—**A central role of Geographic Information Systems (GIS) is to allow the identification and visualisation of relevant spatial features from typically large volumes of data. This requires a querying system to provide both flexibility and usability. While standard GIS querying capabilities are often either very limited, or require a user to have knowledge in specialised areas, techniques in qualitative spatial reasoning have been developed that provide a powerful and intuitive method of representing and reasoning about spatial information. In this paper we present a method for querying about the qualitative distance between features using dynamic networks such as roads, bus or ferry services, and flight paths, rather than only using the Euclidean distance. Linguistic values are used to implement qualitative distances for the linguistic variable proximity. TreeSap GIS with qualitative querying support is presented to demonstrate how qualitative distance measures through a network can provide both a computationally practical solution and a mechanism through which non-experts benefit from powerful search tools.**

I. INTRODUCTION

Modern Geographic Information Systems (GIS) commonly provide powerful tools that allow a user to manipulate, view and query geographic information, allowing the identification and visualisation of relevant spatial features from typically large volumes of data. An effective querying system must provide flexibility, to appropriately capture a user's desired search criteria, and usability, so that the system is appropriately accessible [1]. Despite this, standard GIS querying capabilities are often very limited, (particularly many publicly accessible web-based GIS), or require a user to have knowledge in specialised areas such as Structured Query Language (SQL) or set theory. Furthermore, it is not always appropriate to completely exclude spatial features that are reasonably close to satisfying a query's criteria.

These difficulties stem from the sole reliance of GIS on numerical approaches for representing and analysing spatial data. As previously argued [2] the level of precision and absolute frame of reference employed by numerical methods are not used or needed for a person's cognitive model of space. One result is that people find numerical methods less intuitive than other approaches.

Furthermore, uncertainty is an intrinsic property of information about the physical world. For example,

Carl P. L. Schultz and Robert Amor are with the Department of Computer Science, The University of Auckland, Private Bag 92019, Auckland, New Zealand (email: csch050@ec.auckland.ac.nz, trebor@cs.auckland.ac.nz)

Hans W. Guesgen is with the Institute of Information Sciences and Technology, Massey University, Private Bag 11222, Palmerston North, New Zealand (email: h.w.guesgen@massey.ac.nz)

measuring instruments will always be limited to a finite level of precision, and geographic features continuously change, resulting in numerical data becoming out-of-date [3]. Managing this uncertainty using numerical methods can be awkward, as these methods often rely on the availability of suitable error intervals or probability distributions. A further difficulty is managing the inherent vagueness in statements such as "The Forest is *near* the Pond".

On the other hand, people reason with imprecise and uncertain information in everyday situations, and in particular, a qualitative method for reasoning about spatial information is often employed [4]. In response to this, techniques in Qualitative Spatial Reasoning (QSR) have been developed to address the limitations of purely numerical GIS, by identifying and reasoning about coarse, relevant distinctions between feature relations. Allen's temporal logic [3] has been particular influential, where a set of 13 atomic temporal relations between two time intervals are defined (including: "before", "meets", "overlaps", and so on) and a composition table is provided so that, given three time intervals A, B, C, and given a relation between (A, B) and a relation between (B, C) we can infer the possible relations that can hold between (A, C). For example, if A meets B, and B is before C, then A must also be before C. The composition table forms the basis for an algorithm used to reason about networks of temporal relations.

A qualitative treatment of orientation includes Frank's [5] calculi for reasoning with cardinal directions ("N", "E", "S", "W"), and Freksa's [6] approach where reasoning is conducted based on relative orientations ("front", "left", "back", "right"). Schlieder's panorama [7], is a method where an ordering of visible landmarks (e.g. obtained by turning clockwise) can be used to qualitatively specify a map location. Reasoning about topology includes Region Connection Calculus [8] which uses relations such as overlapping, touching, and proper part of, and Hernandez [2] presents a combined orientation and distance system.

In this paper we address the problem of qualitative spatiotemporal reasoning about spatial features related in dynamic networks, that is, networks grounded in a spatial context (such as transportation networks: roads, walkways, bus routes, flight paths, and so on), where the cost of travelling through the network can change. In the following section the qualitative proximity formalism used to transform numerical distance data into qualitative representations is reviewed. Networks are then presented as a distance measure for relating features on a map, followed by a method of network generation from geometrically defined map features, binding features to networks, and combining networks. A qualitative approach for managing the continuously changing travel costs in a network is then

presented. Finally, TreeSap GIS supporting qualitative querying is described, to demonstrate how qualitative spatial reasoning can provide both a computationally practical solution to GIS querying, and a mechanism through which non-experts benefit from powerful search tools.

II. QUALITATIVE PROXIMITY FORMALISM

Qualitative methods aim at taking the coarsest possible approach to representing and reasoning about information, while still being able to accomplish the required tasks of a given domain [9]. Qualitative Spatial Reasoning (QSR) is restricted to specifying spatial relationships and properties. Being motivated by a cognitive perspective of spatial concepts, QSR commonly makes use of everyday language such as "near", "far", "fast", "slow", "disconnected", and "overlapping". The Qualitative Proximity (QP) formalism is an adapted version of the Fuzzy Proximity formalism described in [10], and is used to facilitate querying using qualitative distance relationships between spatial objects. The proximity linguistic variable may take linguistic values that represent possible qualitative distance relationships. The linguistic values chosen for this example are: touching, very near, near, moderately near, moderately far, far, very far. Figure 1 illustrates two example relationships between objects A, B and A, C. The proximity concepts are relative to a particular dataset (for example, A is near B relative to the much greater distance between A and C), and the context of the application (for example, the distance between A and B is likely to be considered near by a typical target user).
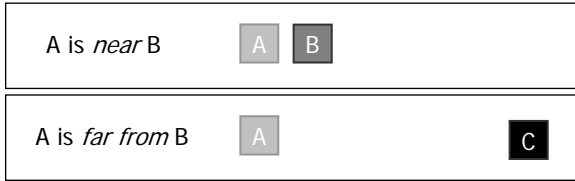

Fig. 1. Subset of the distance relationships defined in QP, where A and B are objects or regions.

We have previously applied fuzzy logic to qualitative formalisms for managing vagueness between spatial relations [11-14]. A fuzzy set is used to characterise each qualitative linguistic value that the proximity linguistic variable can take. A distance relation is then assigned a membership degree in each of these fuzzy sets according to a membership function. To illustrate this, consider the following query: "Find all objects *near* A". As shown in figure 2, a "near" membership value is assigned to every distance relationship that A shares with some other object, indicating how closely each relationship matches the "near" relationship type. More generally, the standard alpha notation can be used, where α_0 indicates the highest possible membership (a value of 100%, where the relationship is definitely considered a "near" relationship), and: $\alpha_1 > \alpha_2 > \alpha_3 > \dots$ indicating decreasing membership values, where the exact values of α_1, α_2, $\alpha_3, \dots$ can be determined according to the application.

Fuzzy membership values are thus assigned by referring to Freksa's conceptual neighbourhood approach [15] as proposed in [13, 14]. Membership grades in each fuzzy set are determined according to the distance the associated qualitative relation is from a reference relation in the conceptual neighbourhood graph [13, 14]. The further away a relation is from the reference relation, the lower its membership grade. Figure 3 illustrates the assignment of membership grades to relations with respect to the "near" relationship type.
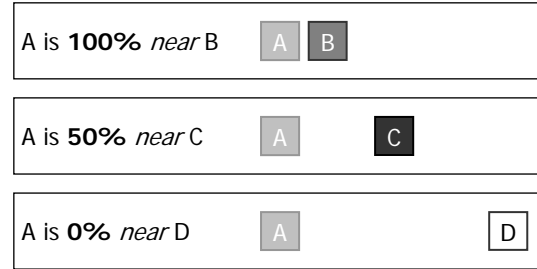

Fig. 2. Results of the query "Find all objects near A". A fuzzy membership value is assigned to every relationship that A shares with another object, representing how well each relationship matches the definition of "near".
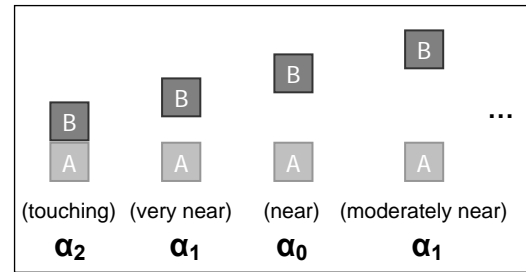

Fig. 3. Extract from the set of QP relations, arranged according to their conceptual neighbourhoods. Membership values (alpha notation) have been assigned with respect to the "near" relationship type.

Given numerical geographic data about features on a map, a network of qualitative relationships between the features is constructed. In the case of QP this firstly requires the distance between each pair of features to be calculated. A typical approach is to calculate the Euclidean straight-line distance between each pair of objects, for example, in a 2D scene:

$$\text{distance}(a,b) = \min\left(\sqrt{\left(x_b - x_a\right)^2 + \left(y_b - y_a\right)^2} \right)$$

The membership degree of the distance relation must then be determined for each fuzzy set characterising a proximity linguistic value. In this case we have experimented with a discrete membership function in order to explore possible tractability benefits, however other membership functions can be applied. Thus, the distance value (d) is mapped to one of the seven linguistic values that the proximity linguistic variable can take. A standard criterion for determining the distance range of each linguistic value (i.e. $d_{upper} - d_{lower}$) is that the distance ranges must be monotonically increasing (e.g. very $\text{near}_{range} \leq \text{near}_{range} \leq$ moderately near_{range}) [16]. The following mapping function

has been used, where d is the distance from the reference object to the beginning of the nth linguistic value distance range (for n = 0,1,...,6 representing each linguistic value from touching to very far away):

$$d = c.n^2$$

The factor c is used to scale the ranges based on the decision as to what distance qualifies as *near* and can be modified for each dataset. In this case a quadratic relationship has been chosen, however, other mapping functions may be more suitable depending on the application (for example, the touching relation is represented by a range of distances rather than only a distance of zero, which may not be appropriate in other contexts). Once the mapping has identified a reference qualitative distance, membership values in fuzzy sets associated with the remaining linguistic values are then assigned according to the conceptual neighbourhood graph.

The set of qualitative relationships (between every pair of objects) makes up the complete relationship network, which can then be referred to, in order to facilitate more advanced query support.

III. DISTANCE RELATIONSHIPS USING NETWORKS

In many cases the Euclidean distance is not an appropriate measure for determining the proximity of two objects. For example, other geographical features such as impassable rivers or cliff edges can interfere with a straight-line path, and the topography can affect the effort required to reach a destination. Other mechanisms for appropriately determining proximity are required.

In this paper we consider the use of directed networks to provide distance measures between features on a map. Transportation networks (roads, walkways, bus routes, flight journeys) are an example of geographically based networks, where the components of the network are grounded in a spatial context, and where travelling through the network corresponds to a change in spatial location.

A network consists of nodes and weighted arcs. The nodes represent path junctions (such as road intersections) or network entry and exit points (such as bus stops, or departure and arrival cities for a flight network). Nodes are related by arcs, which are weighted with a cost value for travelling from the head node to the tail node. The following sections describe the process of creating networks from geometrically defined features, binding other (non-network) features to a network, and combining networks.

A. Generating Networks

A network must be constructed from the geometric objects that are used to describe features on a map, such as points, lines, and polygons. Figures 4 and 5 illustrate the process for generating a road network and a ferry network.
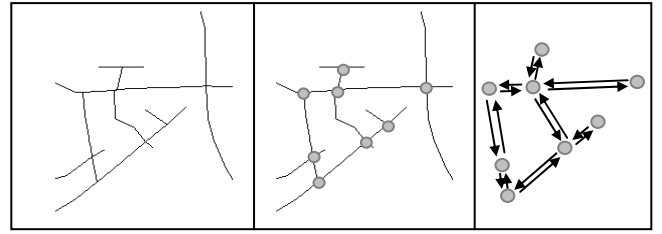

Fig. 4. Using line segment data representing roads (left), determine road junctions (centre) to generate nodes of a network (right).
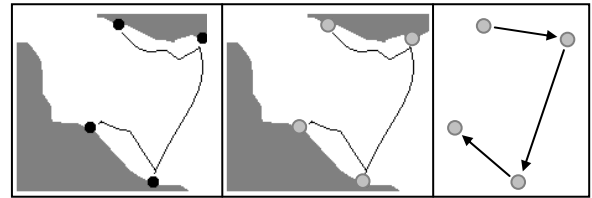

Fig. 5. Using point and line segment data representing ferry journeys between wharfs (left), determine the ferry service entry and exit points (centre), to generate nodes in a network (right).

Firstly the network nodes must be identified, by determining (a) points where paths in the network join, and (b) the network entry and exit points. In the example of a road network, nodes represent road junctions, and are determined by calculating the intersection points of line-segments which belong to different roads. In the ferry network, nodes represent departure and arrival wharfs, and are determined by taking the points used to represent those wharfs.

The second step is to create the arcs of the network. An arc is formed between two nodes whenever it is possible to travel directly between two nodes, without passing through a third. A cost is assigned to the arc, by applying some cost function, representing the cost of travelling from the first node to the second node through the network. For example, three different functions for determining the cost of travelling between two junctions in a road network are: (i) distance to travel (summation of the line segment lengths between the nodes); (ii) time taken to travel between the junctions (distance travelled combined with the road's speed limit); (iii) time taken when considering traffic flow and congestion (distance travelled combined with average car speed along the road). Two different cost functions for travelling between ferry wharfs are: (i) average travel time; (ii) fare charged.

The next step is to determine the cost of the shortest path between every pair of nodes in the network. This requires the application of an all pairs shortest path algorithm, such as the Floyd-Warshall algorithm [17].

Finally, each cost value is used to produce fuzzy sets for the linguistic values of the proximity linguistic variable, as described in section II. This new set of qualitative relationships can now be used to process queries about different nodes in the network, such as "Find all wharfs *near* Devonport wharf via the ferry network".

B. Binding Objects to a Network

The next step, once a network has been generated, is to relate objects and features to one another as a measure of distance through this network.

As illustrated in figure 6, the network and the objects lie on separate layers. A mapping between these layers is required to bind the objects to the network. This requires binding criteria, a node selection strategy, and a binding cost function.

Binding criteria determine whether a feature is directly accessible through the network, by considering the validity of network types, and other geometric and geographic conditions. For example, a café may not be considered directly accessible via a bus service, however, it may be accessible via the footpath network on the condition that some footpath exists in its immediate vicinity.
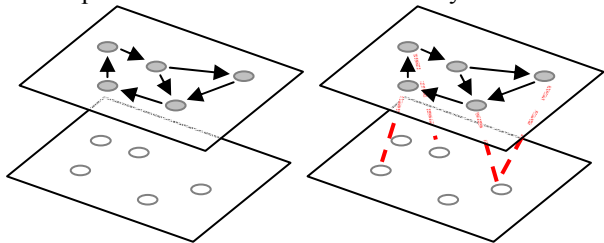

Fig. 6. Objects (e.g. buildings) and networks (e.g. roads) lie on separate layers (left). By applying binding criteria, node selection strategies, and binding cost functions, the objects layer is mapped to the network layer (right).

Once an object has met the criteria for binding, one or more nodes in the network must be selected to which the object will be bound. For example, a café is bound to the node representing the nearest junction (according to the network cost function) along the road that best satisfied the binding criteria.

Finally, the binding must be associated with two costs representing the cost of entering and exiting the network. There is no single way to automatically determine a binding cost between a feature and a network, and so a binding cost function is required. For example, the cost for entering a bus service network can be the journey fare, or the average waiting time before the bus arrives. Binding cost may also be zero, for example, when stepping out of a café onto the footpath, or exiting a bus.

Two arcs (one for entry, one for exit) are then formed between the feature and each of the selected nodes in a binding, weighted with the appropriate costs. These binding arcs provide a mapping between the network layer and the object layer, allowing the distance between two features (i_1, i_2) via the network to be calculated as:

$$\text{cost}(i_1, j_1) + \text{cost}(j_1, j_2) + \text{cost}(j_2, i_2)$$

where j_1, j_2 are the network nodes that i_1 and i_2 are bound to respectively, and $\text{cost}(x,y)$ provides the cost of travelling from x to y.

These costs are then used to produce fuzzy sets for the linguistic values of the proximity linguistic variable, as described in section II. It is assumed that the costs of travelling through the network and entry and exit costs are compatible. This assumption is discussed in more detail in a later section.

C. Combining Networks

It is often necessary to deal with multiple networks in a single query. For example, finding the shortest journey from a café to The University of Auckland can involve walking along the footpath, using the bus service, and taking a ferry ride. This requires a strategy for combining the networks, so that the shortest path spanning multiple networks can be determined.

Combining two or more networks is similar to the previously discussed case of binding objects to a network as illustrated in figure 7. Every node in the first network is treated like an object in an object layer, and bound to the second network. This requires (a) binding criteria, (b) a node selection strategy, and (c) a binding cost function specifying the cost of transferring between networks.
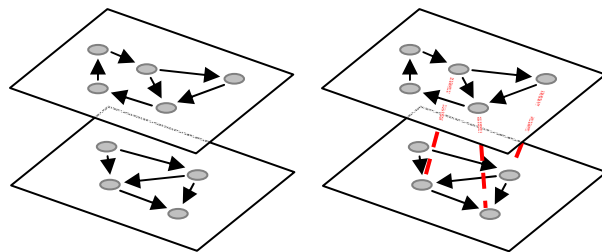

Fig. 7. Different networks lie on separate layers (left). Networks are combined by providing a mapping between the networks (right), analogous to the previous case of binding objects to a network.

The result of combining networks and binding objects is a 3D structure with n-layers. The shortest path between two nodes or objects can cover nodes and arcs from any layer.

IV. DYNAMIC NETWORKS

The cost of travelling through a network is often not static. For example, due to the increased traffic load during rush hour, the travel time along an inner city road is much greater than during off-peak hours. In many cases the travel cost is changing continuously, and is thus difficult to manage from a computational perspective due to issues of appropriate granularity (deciding on the level of precision to model the continuous change), accuracy of the network model (determining the degree of uncertainty in the model, given a level of precision), and resource limitations.

On the other hand, people manage continuous change very effectively in everyday life by forming clusters (rush hour, lunch time, off-peak, and so on) where the network travel costs within a cluster are qualitatively similar. For example, when a person states that it takes 30 minutes to bus to work at 8am, it is implied that the case holds for a range of departure times including 8:01am, 8:02am and so on, and a range of travel durations including 29 minutes, 31 minutes, etc.

The fuzzy community has presented tools to generate these clusters from sample data (for example [18]). A qualitative approach is to then discretise the continuously changing network by taking a representative sample of the network for each identifiable cluster, e.g. the median, mode, maximum, or minimum sampled speeds, or calculating a representative cost, e.g. the average. The result is a discrete set of network states, where the costs between states are qualitatively distinct. This qualitative approach is far less computationally expensive compared to more detailed numerical approaches (such as calculating differential equations), and for many tasks the degree of imprecision is satisfactory.

V. QUALITATIVE QUERYING WITH TREESAP GIS

To demonstrate the usability advantages of QSR in GIS, a desktop application called TreeSap (Topographic Reasoning Application) has been developed that allows qualitative querying about geographic information [19, 20]. The query criteria consists of qualitative relationships between geographic features, for example "Find all Roads *near* all Railways", rather than requiring a numerical distance value.

A. Querying with Multiple Distance Measures

The query is built as a hierarchical tree structure of conditions between features. The criteria are described in terms of a subject feature, and the required qualitative spatial relationships between other features. Relationships may be nested to an arbitrary depth, e.g. as in figure 8, the Building must be *moderately near* a Powerline, such that the Powerline is *very near* Pylon_16. The query can be represented in either a descriptive format (figure 8 top) or a concise presentation (figure 8 bottom).
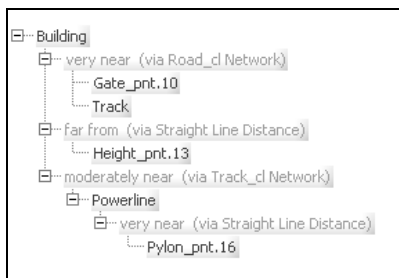

Fig. 8. Screenshots of TreeSap's query interface. The user builds their query in a hierarchical tree structure of conditions, using either a descriptive query representation (top) or a concise representation (bottom).

B. Network Queries

Figure 9 illustrates the results for a query: "Find all Buildings *near* King Edward Parade", with a network cost function of road length. Transparency is used to indicate

how poorly a feature meets the criteria. The roads on the eastern Devonport peninsula are opaque, and as we follow the network north and west the roads grow increasingly transparent. Roads to the east of the southern shore have a small Euclidean distance from Devonport, however they are not considered "near" in terms of the road network.

A second visualisation scheme has been implemented where the user controls an alpha-cut for the query solution quality. Only features that meet the query criteria by the specified threshold or more are displayed, for example, as the threshold is lowered (illustrated in figure 10) the roads firstly expand across the northern coastline, and eventually down and across the southern coastline. This clearly presents the underlying patterns within the query solution.
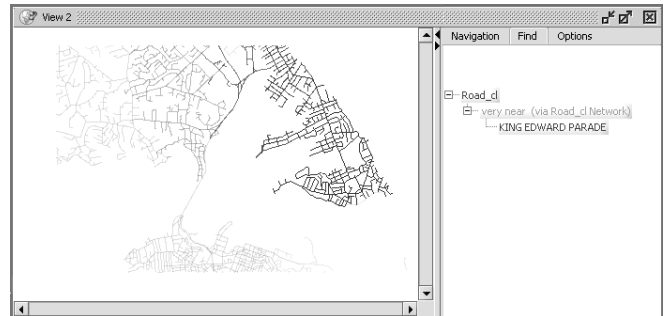

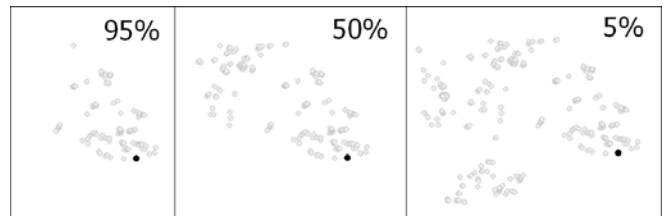Fig. 9. Transparency used to indicate features that do not completely satisfy the query criteria.

Fig. 10. User controls an alpha-cut for the query: "Find Buildings *near* Devonport café (black circle)" so that a feature is only displayed if it meets the query criteria by the given threshold or more.

VI. DISCUSSION AND CONCLUSIONS

We have presented a method for determining the qualitative distance between two features through a network in which the cost of travelling through the network may change over time. TreeSap GIS has been developed to demonstrate how qualitative spatial reasoning can provide both an intuitive and flexible system for querying geographic data, primarily due to the effective handling of vagueness and uncertainty.

One concern that arises when dealing with multiple networks and features is the mixing of different cost units, such as monetary cost and travel time. In this paper we have assumed that costs can be flattened into a single uniform measure, as it is awkward to automatically combine disparate costs without any information about the user or the task. However, with user input or a clear application context, costs can be combined by first normalising the costs

to derive a common unit (e.g. a value between 0 and 1) and then weighting them appropriately.

Network feedback is another issue, where the cost of travelling along a path in the network may affect the state of the network and the costs. For example, if a journey is started during off-peak time (e.g. at 7a.m.), the travel time may be great enough to lead into the next rush hour (e.g. at 8:00a.m.). To deal with this, a cost threshold can be used. Cost calculations then refer to a succeeding network state when the threshold is exceeded.

REFERENCES

[1] M. Sebillo, G. Tortora, and G. Vitiello, "The Metaphor GIS Query Language," *Journal of Visual Languages and Computing*, vol. 11, pp. 439-454, 2000.

[2] D. Hernandez, "Qualitative representation of spatial knowledge," *Lecture Notes in Artificial Intelligence*: Springer-Verlag, vol. 804, 1994.

[3] J. F. Allen, "Maintaining Knowledge about Temporal Intervals," *Communications of the ACM*, vol. 26, pp. 832-843, 1983.

[4] A. G. Cohn and S. M. Hazarika, "Qualitative spatial representation and reasoning: An overview," *Fundmenta Informaticae*, vol. 46, pp. 2-32, 2001.

[5] A. U. Frank, "Qualitative spatial reasoning: cardinal directions as example," *International Journal of Geographical Information Science*, vol. 10, pp. 269-290, 1996.

[6] C. Freksa, "Using orientation information for qualitative reasoning," in *Theories and methods of spatio-temporal reasoning in geographic space*, *Lecture Notes in Computer Science*, A. Frank, I. Campari, and U. Formentini, Eds. Berlin: Springer, vol. 639, pp. 162-178, 1992.

[7] C. Schlieder, "Representing visible locations for qualitative navigation," *Qualitative reasoning and decision technologies, N. Pierra Carrete and M.G.*, pp. 523-532, 1993.

[8] A. G. Cohn, B. Bennett, J. Gooday, and N. M. Gotts, "RCC: a calculus for region based qualitative spatial reasoning," *Geoinformatica*, 1997.

[9] D. Hernandez and J. E., "Special Section on Qualitative Spatial Reasoning (Guest Editors' Introduction)," *Journal of Visual Languages and Computing*, vol. 9, 1998.

[10] H. W. Guesgen, "Reasoning about distance based on fuzzy sets," *Applied Intelligence*, vol. 17, pp. 265-270, 2002.

[11] H. W. Guesgen, "From the egg-yolk to the scrambled-egg theory," *FLAIRS-02*, pp. 476-480, 2002.

[12] U. Loerch and H. W. Guesgen, "Qualitative spatial reasoning under uncertainty in geographical information systems," *IEEE International conference on systems man and cybernetics*, vol. 3, pp. 2169-2174, 1997.

[13] H. W. Guesgen, "Fuzzifying spatial relations," *Studies in fuzziness and soft compting*, vol. 106, pp. 1-16, 2002.

[14] H. W. Guesgen, "Fuzzy reasoning about geographic regions," *Fuzzy modeling with spatial information for geographic problems*, pp. 1-14, 2004.

[15] C. Freksa, "Temporal reasoning based on semi-intervals," *Artificial Intelligence*, vol. 54, pp. 199-227, 1992.

[16] D. Hernandez, E. Clementini, and P. Di Felice, "Qualitative distances," *Spatial Information Theory: a theroretical basis for GIS*, 1995.

[17] T. H. Cormen, C. E. Leiserson, and R. L. Rivest, *Introduction to Algorithms*, first ed: MIT Press and McGraw-Hill, 1990.

[18] W. Pedrycz, *Knowledge-Based Clustering: From Data to Information Granules*, Hoboken, N.J.: Wiley-Interscience, 2005.

[19] C. P. L. Schultz, H. W. Guesgen, and R. Amor, "Computer-Human Interaction Issues when Integrating Qualitative Spatial Reasoning into Geographic Information Systems," *Proc. 7th International Conference, NZ Chapter of the ACM's Special Interest Group on Human-Computer Interaction (SIGCHI-NZ), CHINZ 2006: Design Centered HCI*, Christchurch, New Zealand, pp. 43-51, 2006.

[20] C. P. L. Schultz, T. R. Clephane, H. W. Guesgen, and R. Amor, "Utilization of Qualitative Spatial Reasoning in Geographic Information Systems," *Proc. Progress in Spatial Data Handling, 12th International Symposium on Spatial Data Handling* Vienna, Austria, pp. 27-42, 2006.