

Plateau Regions: An Implementation Concept for Fuzzy Regions in Spatial Databases and GIS

Virupaksha Kanjilal, Hechen Liu & Markus Schneider*

University of Florida, Gainesville, FL 32611, USA,
{vk4, heliu, mschneid}@cise.ufl.edu

Abstract. Many geographical applications need to model spatial phenomena with vague or indeterminate boundaries and interiors. A popular paradigm adopted by the GIS community for this task at the modeling level is fuzzy set theory. A spatial object is fuzzy if locations exist that cannot be assigned completely to the object or to its complement. In previous work, we have proposed an abstract data model of *fuzzy spatial data types* for *fuzzy points*, *fuzzy lines*, and *fuzzy regions* to represent the indeterminacy of spatial data. This paper focuses on the problem of finding an appropriate implementation approach to fuzzy regions. The idea is to approximate a fuzzy region by a so-called *plateau region* consisting of a finite number of crisp regions that are all adjacent or disjoint to each other and associated with different membership values determining the degree of belonging to the fuzzy region. *Geometric union*, *geometric intersection*, and *geometric difference* on fuzzy regions are expressed by corresponding operations on the underlying crisp regions. We leverage that several implementations are already available for crisp regions.

1 Introduction

Spatial databases as the data management foundation of Geographical Information Systems (GIS) represent point, line, and region objects by special data types called *spatial data types* [7]. These data types can be used in the same way as attribute data types as integers, floats, or characters. Their objects have the fundamental feature that they are crisp, that is, they have a definite extent, boundary and shape. However, many spatial objects cannot be described by crisp concepts since they are fuzzy, vague, or indeterminate. A spatial object is fuzzy if locations exist that cannot be assigned completely to the object or to its complement. Hence, spatial fuzziness captures the property of objects that do not have sharp boundaries but rather vague or indeterminate boundaries and interiors. Examples are natural, social, or cultural phenomena like oceans, pollution areas, and English speaking regions. It is impossible to say with precision where the Indian Ocean ends and the Arabian Sea begins. So far, indeterminate spatial objects cannot be represented by available spatial database systems and GIS.

* This work was partially supported by the National Science Foundation under grant number NSF-CAREER-IIS-0347574.

From a modeling standpoint, especially the GIS field has proposed *fuzzy set theory* to characterize and describe indeterminate spatial data. The spatial database field has provided a few proposals to conceptually model such data through *fuzzy spatial data types*. *Fuzzy points*, *fuzzy lines*, and *fuzzy regions* have been defined by appropriately assigning a membership value ranging from 0 to 1 to each point of such an object. A membership value indicates here how strongly or weakly a point belongs to an object. While conceptually some progress has been made, adequate implementation approaches to representing fuzzy spatial data types in spatial database systems are lacking. The main reason is that the sole approximation of the boundary of a fuzzy region is insufficient; the challenge consists in representing its interior with its varying membership values.

The goal of this paper is to provide discrete representations of fuzzy region objects and to specify geometric set operations like union, intersection, and difference on these representations. The idea is to approximate a fuzzy region by a so-called *plateau region* consisting of a finite number of crisp regions that are all adjacent or disjoint to each other and associated with different membership values determining the degree of belonging to the fuzzy region. The benefit of this approach is that we can leverage widely available concepts and implementations of well known crisp region objects. Geometric set operations on plateau regions are expressed by corresponding operations on the underlying crisp regions.

Section 2 discusses related work on approaches to fuzzy spatial data handling. Section 3 reviews our abstract definition of fuzzy regions, describes our approach to representing them by means of plateau regions, and provides a formal definition of plateau regions. Section 4 focuses on the plateau versions of the geometric set operations *fuzzy union*, *fuzzy intersection*, and *fuzzy difference*. These versions are named *plateau union*, *plateau intersection*, and *plateau difference*. Section 5 draws some conclusions and considers future work.

2 Related Work

The geoscience and GIS communities have proposed a large number of conceptual approaches to handling spatial vagueness that are based on fuzzy set theory [10]. Unfortunately, all these approaches have not been devised for a use in a spatial database context and thus do not enable fuzzy spatial data representation, handling, and querying in databases. The work in [1] has been the first approach in this direction. It presents fuzzy set theoretic approaches for handling imprecision in spatial analysis and introduces *fuzzy regions* as a binary relation on the domain of \mathbb{N}^2 (\mathbb{N} denotes the set of natural numbers). The authors themselves have designed an abstract model of *fuzzy spatial data types* [8] based on fuzzy point set topology. This model provides data types for *fuzzy points*, *fuzzy lines*, and *fuzzy regions* as special fuzzy sets from \mathbb{R}^2 and also includes important operations on these types like *fuzzy geometric union*, *fuzzy geometric intersection*, and *fuzzy geometric difference*. These concepts have been deliberately developed as a specification for a possible implementation in spatial database systems. A similar type system of so-called *vague spatial data types* is introduced in [5]. Instead

of the term “fuzzy”, the authors use the term “vague”. They distinguish vague points, vague lines, and vague regions with single and multiple components. The operations are similar to our approach described above. All approaches do not provide an implementation of these data types in a spatial database system.

Implementation approaches are only available for a limited class of indeterminate spatial objects. The approaches in [3, 4] and our own approach, the *Vague Spatial Algebra* (VASA) [6], leverage a three-valued logic with the truth values *true*, *false*, and *maybe*. They approximate a fuzzy region by identifying parts which *definitely* belong to the object, parts which do *definitely not* belong to the object and parts which *maybe* belong to the object. The attractiveness and benefit of these approaches rest on the fact that their conceptual and implementation framework is based on well known, general, and exact models of crisp spatial data types and thus on a wide range of existing definitions, techniques, data structures, and algorithms for crisp spatial objects that need not be redeveloped but only modified and extended, or simply used. However, these approaches are not general enough to represent fuzzy region objects due to their restriction to three truth values and thus three membership values (0, 0.5, 1). On the other hand, plateau region objects with their n -valued logic extend these approaches and preserve their benefits. They provide a better approximation of fuzzy regions as there is no restriction on the number of approximation levels.

3 Plateau Regions

In this section, we propose so-called *plateau regions* as an implementation concept for the fuzzy spatial data type *fregion* for fuzzy regions. We first informally review our earlier abstract definition of fuzzy regions from [8] in Section 3.1. Section 3.2 informally introduces plateau regions for their implementation. In Section 3.3, we give a formal definition of plateau regions.

3.1 Fuzzy Regions

Spatial data handling in available GIS and spatial database systems rests on the assumption that spatial objects like region objects are precisely determined, that each interior point fully belongs to that object, and that the object is delimited by a precisely specified boundary. Many spatial objects, especially those describing natural, social, and cultural phenomena, do not follow this pattern. They are characterized by the feature of *spatial vagueness*. For indeterminate regions this means that the interior may be vague and that the boundary may be blurred.

Figure 1a illustrates an air-polluted area around a chemical factory located at position A . The exhaust fumes emitted by the factory spread around in the region surrounding the factory at A and create a pollution cloud. The shaded region shows the area which has been affected by the pollution particles. The density of pollution particles around the factory is not uniform but varies. The central zone indicated by a darker gray shading has a higher pollution density, and the surrounding zone shown by a lighter gray shading has a lower density.

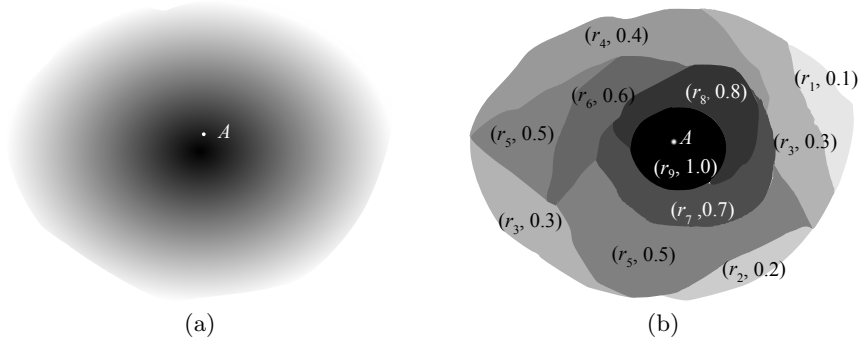


Fig. 1. An example of a fuzzy region modeling an air-polluted area (a) and its representation as a plateau region (b)

There is no clear boundary of this region. We model such a spatial phenomenon by a *fuzzy region*.

Fuzzy set theory [10] has been a popular approach to modeling vague spatial objects and resulted in a concept of fuzzy regions [5, 8]. A crisp region object is conceptually modeled as a particular point set of the Euclidean plane [7, 9]. Each of its points belongs definitely and completely to it. Let *region* be the spatial data type for crisp region objects. In contrast, a fuzzy region object is conceptually modeled as a particular point set of the Euclidean plane such that each of its points may completely, partially, or not at all belong to it. This especially means that a point can belong to multiple fuzzy spatial objects. Let *fregion* be the spatial data type for fuzzy region objects. If $\tilde{A} \in \text{fregion}$, this means that each point of \mathbb{R}^2 is mapped to a value of the real interval $[0, 1]$ that represents the degree of its membership in \tilde{A} . Hence, for a fuzzy region \tilde{A} , $\mu_{\tilde{A}} : \mathbb{R}^2 \rightarrow [0, 1]$ is its *membership function*, and $\tilde{A} = \{(p, \mu_{\tilde{A}}(p)) \mid p \in \mathbb{R}^2\}$ describes all its points in \mathbb{R}^2 with their membership values. The distribution of membership values within a fuzzy region may be smooth, continuous, or piecewise continuous.

3.2 Plateau Regions as a Representation of Fuzzy Regions

To the authors' best knowledge, implementations of fuzzy regions are not available, especially not in a spatial database and GIS context. A crisp, curvilinear region is usually approximated by well known polygonal structures for outer cycles and holes cycles of its components with the assumption that the enclosed interior belongs completely to the region. However, such an approximation is not so easy to obtain for fuzzy regions since first, they usually have an indeterminate boundary and a blurred interior, second, they have infinitely many interior points but only finitely many representations can be kept in a computer, and third, each point can have a different membership value.

In this paper, the fundamental idea for representing and approximating fuzzy regions is to leverage available crisp spatial data types [7] and software packages

implementing them. Several reasons have led to this design decision. First, this strategy enables us to take advantage of existing definitions, techniques, data structures, algorithms, etc., that need not be redeveloped but only modified and extended, or simply used. Second, at the conceptual level, the correctness of the definitions of fuzzy spatial concepts largely rests on the correctness of the already defined crisp spatial concepts; thus, we reduce the chance of errors in our definitions. Third, operations and predicates can be easily translated to the implementation level (see Section 4). This means that having an available correct implementation of crisp regions and their operations, we can correctly implement fuzzy regions and their operations on top of them.

Our implementation concept for fuzzy regions is called *plateau region*. A plateau region is a finite collection of crisp regions where each region is associated with a membership value and thus forms a “plateau” consisting of a conceptually infinite number of points of equal membership. Figure 1b illustrates the concept and shows a representation of the fuzzy region in Figure 1a as a plateau region with the nine crisp regions r_1, \dots, r_9 and their associated membership values. All membership values are different, and any pair of crisp regions is either disjoint (e.g., r_1 and r_9) or adjacent (e.g., r_2 and r_5). A single crisp region can consist of several components (e.g., r_3) that all have the same membership value. Any two crisp regions of a plateau region must be either disjoint or adjacent since otherwise two crisp regions would share interior points with different membership values. While this can be avoided for interior points, this is not the case for the boundaries of two or more adjacent crisp regions since they have common points with different membership values. In Figure 1b, e.g., r_3 and r_5 have the membership values 0.3 and 0.5, respectively, and r_4 , r_6 , and r_8 have the membership values 0.4, 0.6, and 0.8 respectively. We solve this inconsistency by assigning the highest membership value to all common boundary points of two or more adjacent crisp regions (e.g., in Figure 1b, this is 0.5 for r_3 and r_5 , and 0.8 for r_4 , r_6 , and r_8). The reason is that a boundary point shared by n crisp regions ($n \geq 2$) is guaranteed to belong to the fuzzy region with the highest membership value among the membership values of the n regions.

3.3 Formal Definition of Plateau Regions

We are now able to give a formal definition of the spatial data type *fregion* for fuzzy regions based on the concept of plateau regions.

$$\begin{aligned}
 \text{fregion} &= \{pr_1, \dots, pr_k \mid \\
 \text{(i)} & \quad k \in \mathbb{N} \\
 \text{(ii)} & \quad \forall 1 \leq i \leq k : pr_i = \langle (r_{i,1}, m_{i,1}), \dots, (r_{i,n_i}, m_{i,n_i}) \rangle \\
 \text{(iii)} & \quad \forall 1 \leq i \leq k : n_i \in \mathbb{N} \cup \{0\} \\
 \text{(iv)} & \quad \forall 1 \leq i \leq k \quad \forall 1 \leq j \leq n_i : r_{i,j} \in \text{region} \\
 \text{(v)} & \quad \forall 1 \leq i \leq k \quad \forall 1 \leq j \leq n_i : m_{i,j} \in]0, 1] \\
 \text{(vi)} & \quad \forall 1 \leq i \leq k \quad \forall 1 \leq j < l \leq n_i : r_{i,j} \text{ disjoint } r_{i,l} \vee r_{i,j} \text{ meets } r_{i,l} \\
 \text{(vii)} & \quad \forall 1 \leq i \leq k \quad \forall 1 \leq j < l \leq n_i : m_{i,j} < m_{i,l} \\
 \text{(viii)} & \quad \forall 1 \leq i \leq k \quad \forall 1 \leq j \leq n_i \quad \forall p \in r_{i,j}^\circ : \mu(p) = m_{i,j} \\
 \text{(ix)} & \quad \forall 1 \leq i \leq k \quad \forall p \in \bigcup_{j=1}^{n_i} \partial r_{i,j} : \mu(p) = \max\{m_{i,j} \mid 1 \leq j \leq n_i, p \in \partial r_{i,j}\}
 \end{aligned}$$

Each plateau region is represented as a finite sequence of pairs (condition (ii)) consisting of an object of the well known *spatial data type region* [7] for complex crisp regions (condition (iv)) and a membership value (condition (v)) indicating the degree of belonging of the crisp region to the fuzzy region. We call such a crisp region a *subregion* of the plateau region. The number of pairs, and hence regions, depends on each plateau region and can thus be different for different plateau regions (condition (iii)). If the number of pairs is equal to zero, we obtain the *empty plateau region*. Condition (vi) states that any two crisp regions associated with a plateau region are topologically either disjoint or adjacent to each other. This is expressed by the well known *topological predicates disjoint* and *meet* on complex regions [9]. Condition (vii) requires that all membership values are different and that all pairs of the sequence are ordered by their membership values. This caters for a unique representation of a plateau region. Conditions (viii) and (ix) take care of a precise assignment of membership values to the points of a plateau region. The \circ operator and the ∂ operator used in the conditions are point-set topological operators which determine all interior points and boundary points, respectively, of a point set. All interior points of a crisp subregion obtain the membership value associated with the region in a corresponding pair of the sequence (condition (viii)) of a plateau region. Each boundary point obtains the highest membership value of all crisp subregions of a plateau region to which the point belongs (condition (ix), compare to Section 3.2). Note that we do not explicitly represent and store single boundary points or linear boundary parts that have a different membership value than the interior of a pertaining subregion.

4 Geometric Set Operations on Plateau Regions

Geometric set operations belong to the most important operations on spatial objects. In the fuzzy region case, the operations fuzzy intersection, fuzzy union, and fuzzy difference all have the signature *region* \times *region* \rightarrow *region* and are supposed to be represented by corresponding operations on plateau regions. These operations are defined on the basis of the already existing geometric set operations *intersection* (\otimes), *union* (\oplus), and *difference* (\ominus) on crisp regions.

Their formal definition requires an auxiliary construction operator \odot that enables us to insert a pair $(r, m) \in \text{region} \times [0, 1]$ into the ordered representation of a plateau region $pr = \langle (r_1, m_1), \dots, (r_n, m_n) \rangle$ for some $n \in \mathbb{N}$. We define:

$$pr \odot (r, m) = \begin{cases} pr & \text{if } r = \emptyset \text{ or } m = 0 \\ \langle (r, m) \rangle & \text{if } pr = \langle \rangle \text{ and } r \neq \emptyset \\ \langle (r_1, m_1), \dots, (r_i \oplus r, m_i), \dots, (r_n, m_n) \rangle & \text{if } r \neq \emptyset \text{ and } n \geq 1 \text{ and } \exists i \in \{1, \dots, n\} : m_i = m \\ \langle (r_1, m_1), \dots, (r_i, m_i), (r, m), (r_{i+1}, m_{i+1}), \dots, (r_n, m_n) \rangle & \text{if } r \neq \emptyset \text{ and } n \geq 2 \text{ and } \exists i \in \{1, \dots, n-1\} : m_i < m < m_{i+1} \\ \langle (r, m), (r_1, m_1), \dots, (r_n, m_n) \rangle & \text{if } r \neq \emptyset \text{ and } n \geq 1 \text{ and } m < m_1 \\ \langle (r_1, m_1), \dots, (r_n, m_n), (r, m) \rangle & \text{if } r \neq \emptyset \text{ and } n \geq 1 \text{ and } m > m_n \end{cases}$$

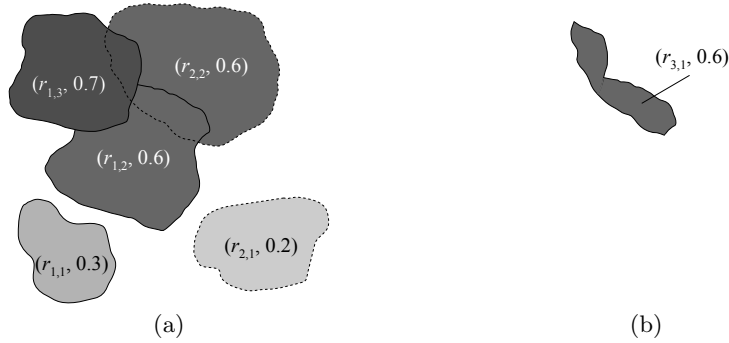


Fig. 2. A scenario of two fuzzy regions modeled as plateau regions pr_1 and pr_2 (a) and their geometric intersection $pr_3 = \text{fintersection}(pr_1, pr_2)$ (b)

Note that \odot is left-associative, i.e., $pr \odot (r_1, m_1) \odot (r_2, m_2) = (pr \odot (r_1, m_1)) \odot (r_2, m_2)$. For $pr \odot (r_1, m_1) \odot \dots \odot (r_n, m_n)$ we also write $pr \odot \bigodot_{i=1}^n (r_i, m_i)$.

As an illustrating example for demonstrating the geometric set operations, we use the spatial scenario of two plateau regions $pr_1 = \langle (r_{1,1}, 0.3), (r_{1,2}, 0.6), (r_{1,3}, 0.7) \rangle$ and $pr_2 = \langle (r_{2,1}, 0.2), (r_{2,2}, 0.6) \rangle$ given in Figure 2a. Note that each subregion in our example is a simple region but might be a complex region that includes multiple components labeled with the same membership value and that contains holes in the general case.

4.1 Plateau Intersection

Intersecting two plateau regions pr_1 and pr_2 means that each subregion of pr_1 must be geometrically intersected with each subregion of pr_2 and that their smaller membership value is assigned to the resulting non-empty subregion [8]. In our example in Figure 2a, we obtain $3 \cdot 2$ subregion pairs that have to be intersected since pr_1 contains three components and pr_2 contains two components. If the geometric intersection of two subregions is empty, we discard this result. Otherwise, the found subregion is part of the resulting plateau region, and we assign the smaller membership value of both operand subregions to it. The reason is that only the smaller membership value guarantees that the points of the intersection belong to both subregions. For example, in Figure 2a, the intersection of the subregions $r_{1,3}$ and $r_{2,2}$ leads to a non-empty subregion that is assigned the membership value 0.6.

Since the creation of subregion pairs and their intersection is a local operation, it can happen that different resulting subregions are labeled with the same membership value. For example, in Figure 2a, the intersection of $r_{1,3}$ and $r_{2,2}$ will obtain the membership value 0.6. Similarly, the intersection of $r_{1,2}$ and $r_{2,2}$ will obtain the same membership value. Since according to the plateau region definition in Section 3.3 all subregions of a plateau region must have different membership values, we have to compute the geometric union of both subregions obtained so far and assign the common membership value 0.6 to it. Figure 2b

shows the result of the plateau intersection for our example. It is a plateau region $pr_3 = \langle (r_{3,1}, 0.6) \rangle$.

The formal definition of the plateau intersection operation leverages the \odot operator that takes care of all the aforementioned particular situations when constructing the resulting plateau region.

Let $pr_1, pr_2 \in fregion$ with $pr_k = \langle (r_{k,1}, m_{k,1}), \dots, (r_{k,n_k}, m_{k,n_k}) \rangle$ for $k \in \{1, 2\}$. Then

$$fintersection(pr_1, pr_2) = \langle \odot \bigcirc_{\substack{1 \leq i \leq n_1 \\ 1 \leq j \leq n_2}} (r_{1,i} \otimes r_{2,j}, \min(m_{1,i}, m_{2,j})) \rangle$$

This definition uses an incremental strategy by starting with the empty plateau region $\langle \rangle$ and incrementally adding local results from intersections of subregion pairs.

4.2 Plateau Union

Forming the union of two plateau regions pr_1 and pr_2 means that each subregion of pr_1 must be geometrically merged with each subregion of pr_2 and that their larger membership value is assigned to the resulting non-empty subregion [8]. Two main spatial configurations can arise. If the intersection of the two subregions is empty, both subregions are copied with their respective (equal or unequal) membership values into the resulting plateau region. In Figure 2a, this is the case for the subregions $r_{1,1}$ and $r_{2,1}$. Otherwise, if the two subregions intersect, three new subregions are stored in the resulting plateau region, namely the subregion that is the result of the intersection and that is labeled with the larger membership value, and the two subregions from which we subtract the intersection and which we label with their original membership values. A subregion obtained as an intersection of two subregions gets their larger membership value since at least one of them can guarantee the higher extent of belonging. In Figure 2a, the intersections of $r_{1,3}$ and $r_{2,2}$ as well as of $r_{1,2}$ and $r_{2,2}$ are non-empty. In the first case, we obtain a subregion as an intersection with the membership value 0.7, the subregion of $r_{1,3}$ from which we subtract the intersection with the membership value 0.7, and the subregion of $r_{2,2}$ from which we subtract the intersection with the membership value 0.6. In the second case, the same strategy leads to three subregions that all have the membership value 0.6. Subregions with the same membership value have always to be merged in the resulting plateau region. Figure 3a shows the result of the plateau union for our example. It is a plateau region $pr_4 = \langle (r_{4,1}, 0.2), (r_{4,2}, 0.3), (r_{4,3}, 0.6), (r_{4,4}, 0.7) \rangle$.

For the formal definition of the plateau union operation we assume again two plateau regions $pr_1, pr_2 \in fregion$ with $pr_k = \langle (r_{k,1}, m_{k,1}), \dots, (r_{k,n_k}, m_{k,n_k}) \rangle$ for $k \in \{1, 2\}$. Then

$$funion(pr_1, pr_2) = \langle \odot \bigcirc_{\substack{1 \leq i \leq n_1 \\ 1 \leq j \leq n_2}} \left((r_{1,i} \otimes r_{2,j}, \max(m_{1,i}, m_{2,j})) \right. \\ \left. \odot (r_{1,i} \ominus r_{2,j}, m_{1,i}) \right. \\ \left. \odot (r_{2,j} \ominus r_{1,i}, m_{2,j}) \right) \rangle$$

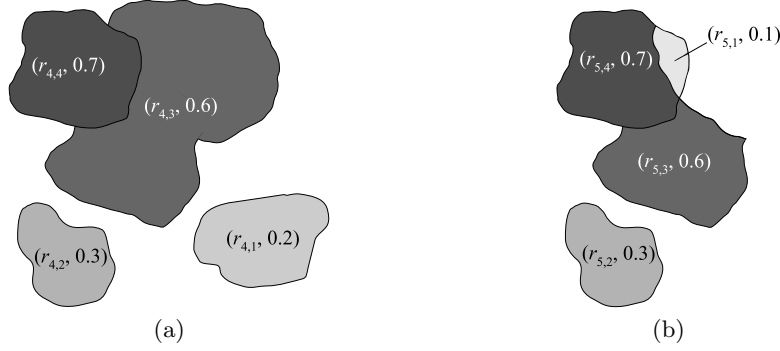


Fig. 3. The geometric union $pr_4 = \text{union}(pr_1, pr_2)$ (a) and the geometric difference $pr_5 = \text{fdifference}(pr_1, pr_2)$ (b) of the two plateau regions pr_1 and pr_2 in Figure 2a

Again, the \odot operator merges different subregions with equal membership values and orders the resulting subregions with respect to increasing membership values.

4.3 Plateau Difference

Forming the difference of two plateau regions pr_1 and pr_2 means that each subregion of pr_2 must be geometrically subtracted from each subregion of pr_1 and that the membership value of the latter subregion is diminished by the membership value of the former subregion [8]. Two main spatial configurations can arise. If the two subregions do not intersect, then the subregion of pr_1 is copied with its membership value into the resulting plateau region. In Figure 2a, this is the case for the subregion pairs $r_{1,1}$ and $r_{2,1}$, $r_{1,1}$ and $r_{2,2}$, $r_{1,2}$ and $r_{2,1}$, and $r_{1,3}$ and $r_{2,1}$. Otherwise, if the two subregions intersect, we first add the geometric difference of the subregion of pr_1 and the subregion of pr_2 with the membership value of the former subregion to the resulting plateau region. Finally, we add the intersection of both subregions with the difference of both membership values to the resulting plateau region. Figure 2a shows two such scenarios. First, the subregions of $r_{1,3}$ and $r_{2,2}$ intersect. The geometric difference of both subregions is added with the membership value 0.7 to the resulting plateau region. Then the intersection of both subregions is added with the membership value $0.7 - 0.6 = 0.1$ to the new plateau region. Hence, the second subregion “weakens” the first subregion. Second, the subregions of $r_{1,2}$ and $r_{2,2}$ intersect. Again the geometric difference is added in a similar way as before. For the intersection of both subregions we obtain a subregion with membership value 0. This corresponds to an empty region. The \odot operator prevents such a region from being inserted into the resulting plateau region. Figure 3b shows the result of the plateau difference for our example. It is a plateau region $pr_5 = \langle (r_{5,1}, 0.1), (r_{5,2}, 0.3), (r_{5,3}, 0.6), (r_{5,4}, 0.7) \rangle$.

For the formal definition of the plateau difference operation let $pr_1, pr_2 \in \text{fregion}$ with $pr_k = \langle (r_{k,1}, m_{k,1}), \dots, (r_{k,n_k}, m_{k,n_k}) \rangle$ for $k \in \{1, 2\}$. For $a, b \in \mathbb{R}$, we define further that $a \dot{-} b = a - b$ if $a > b$, and $a \dot{-} b = 0$ otherwise. Then

$$fdifference(pr_1, pr_2) = \langle \rangle \odot \bigcirc_{\substack{1 \leq i \leq n_1 \\ 1 \leq j \leq n_2}} ((r_{1,i} \ominus r_{2,j}, m_{1,i}) \odot (r_{1,i} \otimes r_{2,j}, m_{1,i} \dot{-} m_{2,j}))$$

5 Conclusions and Future Work

This paper introduces plateau regions as an implementation concept of fuzzy regions in the context of spatial databases and GIS. A special characteristic of our approach is that plateau regions rest on well known concepts, data structures, algorithms, and implementations of crisp regions. Since crisp regions are designed and implemented as abstract data types, all structural details are hidden, and all crisp concepts only have to be called and applied.

This paper is the beginning of a larger effort to design and implement a so-called *Spatial Plateau Algebra* that is supposed to offer a type system including *plateau points*, *plateau lines*, and *plateau regions* together with a comprehensive collection of plateau operations and predicates as implementations of their fuzzy counterparts. As to operations, our particular interest relates to metric operations and to topological predicates for fuzzy spatial objects.

References

1. D. Altman. Fuzzy Set Theoretic Approaches for Handling Imprecision in Spatial Analysis. *Int. Journal of Geographical Information Systems*, 8(3):271–289, 1994.
2. P. A. Burrough and A. U. Frank, editors. *Geographic Objects with Indeterminate Boundaries*. GISDATA Series, vol. 2., Taylor & Francis, 1996.
3. E. Clementini and P. Felice. A Spatial Model for Complex Objects with a Broad Boundary Supporting Queries on Uncertain Data. *Data & Knowledge Engineering*, 37:285–305, 2001.
4. A. G. Cohn and N. M. Gotts. *The ‘Egg-Yolk’ Representation of Regions with Indeterminate Boundaries*. In Burrough and Frank [2], 1996.
5. A. Dilo, R. By, and A. Stein. A System of Types and Operators for Handling Vague Spatial Objects. *Int. Journal of Geographical Information Science*, 21(4):397–426, 2007.
6. A. Pauly and M. Schneider. VASA: An Algebra for Vague Spatial Data in Databases. *Information Systems*, 2009. In press.
7. M. Schneider. *Spatial Data Types for Database Systems - Finite Resolution Geometry for Geographic Information Systems*, volume LNCS 1288. Springer-Verlag, 1997.
8. M. Schneider. Uncertainty Management for Spatial Data in Databases: Fuzzy Spatial Data Types. In *6th Int. Symp. on Advances in Spatial Databases*, LNCS 1651, pages 330–351. Springer-Verlag, 1999.
9. M. Schneider and T. Behr. Topological Relationships between Complex Spatial Objects. *ACM Trans. on Database Systems*, 31(1):39–81, 2006.
10. L. A. Zadeh. Fuzzy Sets. *Information and Control*, 8:338–353, 1965.