# Oriented Matroids as a Foundation for Space in GIS

**Abstract**

This paper explains why a particular mathematical structure, called an oriented matroid, is relevant to the foundations of geographic information science. A finite set of points in the plane can be given the structure of an oriented matroid and this finite structure captures enough of the geometrical structure of the idealized continuous plane to be able to perform useful computations such as finding the convex hull of a subset of the points. The advantage of oriented matroids over some approaches to discrete space, such as Schneider's realms, is that they are purely finite combinatorial structures that make no mention of $\mathbb{R}^n$ or $\mathbb{Q}^n$, even though the intended models are finite subsets of these spaces. We provide brief details of the theory of oriented matroids, but the emphasis is on providing a motivation for the study of oriented matroids in the GIS context.

**Keywords:** formal theory of space, discrete space, oriented matroids, computable geometry.

## 1 Introduction

This paper explains why a particular mathematical structure, called an oriented matroid, is relevant to the foundations of geographic information science (GIS). In order to place our explanation within the appropriate context, we begin by considering the general issue of the spatial foundation of GIS.

The classical conception of space in GIS, which is by no means universally accepted, has a number of features that include what may be termed continuity and externality. By continuity we mean that a region can always be subdivided into strictly smaller subregions; by externality we mean that space exists independently of the objects or fields about which we record information. The view that space is continuous is found not only in the quantitative approach based on real numbers, but also in the qualitative approach of the region-connection calculus [Coh96]. Discrete alternatives to space have been proposed by several authors including Galton [Gal99].

Both the object and the field paradigms in GIS are based on a view of the world where space exists prior to entities which may populate it (objects), or vary over it (fields). A radical alternative would be that a place is constructed by the events that happen there and that places are different only because different things happen there. In this view space does not exist on its own. A full development of this approach does not appear to have been carried out. The terminology *object-centered* as opposed to *space-centred* is apposite. However, Goodchild's usage [Goo01, p3], views the objects as located in a continuous space, rather than the space itself being created by the relationships between objects.

There is good motivation, from the GIS standpoint, for investigating space which is neither continuous, nor exists apart from relationships between events or processes. One might suppose that such approaches are no more than a computationally convenient way of modelling the world, and that we would be approximating a genuinely continuous and absolute space. However, it is remarkable that recent ideas in theoretical physics suggest that space might 'really' be both discrete and created by relationships rather than being absolute. The two aspects of space are in fact intimately connected As Smolin puts it [Smo00, p96]

> "To understand what we mean when we say that space is discrete, we must put our minds completely into the relational way of thinking, and really try to see the world around us as nothing but a network of evolving relationships. These relationships are not among things situated in space – they are among the events that make up the history of the world. The relationships define the space, not the other way round."

If then we want space which is both discrete and relational where should we look? What might be the mathematical form of such a theory? At present, no such theory has been constructed in a completed form, either for the needs of the foundations of GIS or of quantum gravity. It would be premature to guess what all the ingredients should be, but our purpose in this paper is to demonstrate that for the needs of GIS, oriented matroids are a particularly promising basis. Their value in theoretical physics is outside our scope, but we note that connections are being investigated, for example Nieto's work [Nie02].

### Overview

The structure of the paper is as follows. In section 2 we consider first the classical notion of space based on the real numbers, $\mathbb{R}$, discuss its shortcomings, and describe briefly some existing alternatives. In the following section we give an overview of Knuth's $CC$-systems [Knu91]. These form a special class of oriented matroids and Knuth's motivation for investigating these was the development of a Delaunay triangulation algorithm, which is important in processing geographic data [Jon97, p200]. We introduce the definition and key concepts of matroid theory in section 4, providing two illustrative examples in sections 4.2 and 4.3. The matroid concepts are developed in section 5 where we consider oriented matroids and extend the examples of section 4. Our proposals for how oriented matroids might be used in GIS are presented in section 6, and the final section provides some conclusions.

## 2 Spatial Representations

The most common view of geographical space is that it forms a plane which can be described in terms of real number coordinates. For three dimensional data, $\mathbb{R}^3$, can be used; taking time as a fourth dimension leads to $\mathbb{R}^4$. In this section we review the drawbacks of of this approach, and consider some alternatives to spaces based on $\mathbb{R}$.

### 2.1 Deficiency of $\mathbb{R}^n$

The problem with using $\mathbb{R}^n$ to reason about computation is that its mathematics has to be translated into computational terms. Any program written in terms of this mathematics has to be translated in order to be implemented, and it might not be obvious how to do this. Worse, a completely accurate translation might not even be possible.

Some authors have expressed the view that for practical purposes the inadequacies of the translation are negligible compared with the accuracy of available data. Goodchild [Goo01, p2] says *"Double precision offers 14 decimal digits . . . , an absurd level of precision given the typical accuracy of* *geographical data. ... only when coordinates are represented by short integers is there a need to be concerned about machine precision".* There are a number of objections to this view which need to be considered. While the precision of 14 digits is more than adequate, it has to be remembered that iterative calculations performed on such data can readily reduce the precision to a level which does have practical consequences. Another objection is that blithely computing with computer represented reals is not acceptable if we want to be able to argue rigourously about the behaviour of our systems. To be able to provide assurance that systems have specified behaviour, an underlying theory is necessary. Without such an underlying theory we cannot specify exactly what an implementation of a convex hull algorithm, say, does. We can specify the algorithm's behaviour when it operates on real numbers, but these cannot be implemented.

Computing the intersection of two lines is an instance of a class of geometric intersection problems the practical importance of which for GIS is well known, see, for example, [Jon97, pp187–192] and [Wis02, chap. 3]. Suppose we need to compute the point of intersection, $p$, if it exists, of straight line segments $[w, x]$ and $[y, z]$ in the plane. It may be assumed that the endpoints are inputs or outputs of some computation and so each contains only a finite amount of information; so they are points of $\mathbb{Q}^2$ (i.e. with rational coordinates). The calculation is mathematically very simple: if either of the line segments is horizontal or vertical it is even simpler, but otherwise $p = \lambda w + (1-\lambda)x$, where

$$\lambda = \frac{(y_2 - z_2)(x_1 - z_1) - (x_2 - z_2)(y_1 - z_1)}{(w_2 - x_2)(y_1 - z_1) - (w_1 - x_1)(y_2 - z_2)}$$

where $w = (w_1, w_2)$, etc. Obviously $p \in \mathbb{Q}^2$, and the numerator and denominator of its coordinates are not arbitrarily large, but are bounded by the size of the numerator and denominator of the other points. It often happens, however, perhaps during a process that involves many iterations, that they are still too large to be stored in the computer at hand. Therefore, unless the computation is to be aborted, $p$ is *rounded* to the closest point $q$ that can be stored. Then $p$ is the point of intersection in the mathematics but $q$ is the computed point of intersection, which is a **rounding error**.

The problem is illustrated in figure 1. Think of the set of pairs of floating-point numbers in a computer as a grid of points in the plane. The intersection point $p$ is not a point in the computer, so is rounded to the closest point that is, which is $q$. But this rounding gives an error, for example, when it is asked on which side of the dotted line the point of intersection lies: $p$ lies on one side and $q$ on the other.

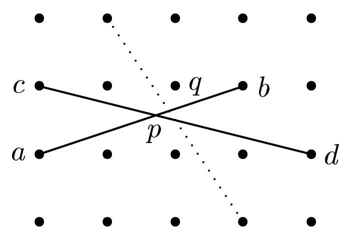


Figure 1: The Intersection Problem

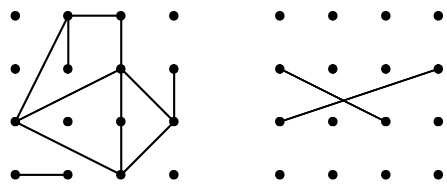

Figure 2: Example of realm on left, but right hand diagram is not a realm.

### 2.2 Discrete Spaces Embedded in $\mathbb{R}^n$

We now consider two approaches to a discrete space embedded in $\mathbb{R}^n$. These contrast sharply with oriented matroids which provide an approach independent of $\mathbb{R}^n$.

#### 2.2.1 Realms

A **realm** [Sch97] is a grid of points in the plane together with a collection of straight line segments between pairs of these points such that the intersection of any two line segments is either empty or an end-point of both. In a computer, the points are pairs of floating-point numbers, and a line is represented by its two end-points. It is precisely because of the intersection problem above that the line segments in a realm are subject to the restriction that they are; built into the very definition of a realm is a restriction on the amount of mathematics of $\mathbb{R}^n$ or $\mathbb{Q}^n$ that is to be used in computation.

#### 2.2.2 Digital Topology and Geometry

A computer screen is a set of pixels, and a **digital object** is any subset of this set. Many questions concerning the topology of digital objects arise naturally in image analysis: When is an object connected? What is its boundary? Does it contain any holes?

The meaning of these questions depends on how a screen is interpreted mathematically, and a natural interpretation is to regard a screen as a grid of squares in the plane $\mathbb{R}^2$, each pixel representing a square. This approach suffers from the same problem as that noted for realms: the space is embedded in idealized Euclidean space. We argue that it is far more computationally relevant to embed them in a finite mathematical structure.

The problem with an embedding in continuous space is that the definitions of digital connectivity etc. are in terms of infinite criteria (a digital image may be defined as being connected if its corresponding uncountably infinite point set in the plane is connected), whereas any program that checks whether a digital image is connected is, at least if it terminates, implementing finite criteria. There is therefore a gap between the mathematics of digital images and its implementation. Digital topology was introduced in order to close this gap, by regarding a screen as a finite mathematical structure. The original approach was to regard a screen as a graph (in which the vertices are pixels, and the edges represent adjacency of pixels). A subsequent approach, by Khalimsky, and later by Kovalevsky, developed mainly to obtain an improved digital Jordan curve theorem, is to regard a screen as a finite non-Hausdorff topological space. The two approaches are described and compared in [KR91].

Digital topology is a well-established discipline, and the topology of digital images is by now well-understood. The same cannot be said for the geometry of digital images. Although the name "digital geometry" is being used more often, most of the work that goes under this heading is still entirely topological. To provide a digital geometry prompts a number of questions, as yet unanswered. Are there useful finite geometries associated with a screen that would provide a mathematical basis for digital geometry? Are these geometries axiomatic? Can, for example, a straight line or a convex set on a screen be regarded as actually being a straight line or convex set in some autonomous axiomatic geometry, and not merely as an approximation of some Euclidean straight line or convex set? We believe that matroids and oriented matroids are highly relevant to these issues, and for a concrete example of the use of oriented matroids in spatial computation we turn to the work of Knuth on the foundations of computational geometry.

## 3 Knuth's CC Systems

Knuth [Knu91] investigated the connection between geometric computation and matroids through his work on $CC$-systems, which are closely related to oriented matroids. Below we see how $CC$-systems encode some geometrical structure of certain finite sets of points using a three-place relation. Knuth showed that this provided enough structure to carry out useful computations, such as a convex hull algorithm. This provides a good illustration of the general oriented matroid approach of capturing geometric information in a purely finite combinatorial structure that makes no mention of $\mathbb{R}^n$ or $\mathbb{Q}^n$, even though the intended models are finite subsets of these spaces.

### 3.1 Definition of $CC$-System

A $CC$**-system** consists of a finite set $E$ together with a ternary relation $B$ (that $(x, y, z) \in B$ is written simply as $xyz$, and that $(x, y, z) \notin B$ is written $\neg xyz$) such that

($CC1$) $xyz \Rightarrow yzx$;

($CC2$) $xyz \Rightarrow \neg xzy$;

($CC3$) $xyz \vee yxz$;

($CC4$) $zxy \wedge wzy \wedge wxz \Rightarrow wxy$;

($CC5$) $zyv \wedge zyw \wedge zyx \wedge zvw \wedge zwx \Rightarrow zvx$.

These structures have the following geometry. A **straight line** is any pair of distinct points $\{a, b\}$, and its corresponding **open half-spaces** are the sets $\{x \mid abx\}$ and $\{x \mid axb\}$. The **convex hull** of a set is then the set of those points that cannot be separated from it: $x$ is in the convex hull of $X$ if there is no open half-space that contains $x$ and does not intersect $X$. Knuth then proceeds to develop algorithms, expressed in terms of the structure of the systems alone, for computing the geometry of CC-systems, and one of the main points is that this leads to significantly better algorithm design for computational geometry.

### 3.2 Models of $CC$-Systems

The main intended models are as follows. Consider any finite subset $E$ of the plane that does not contain any triple of collinear points, and let $B(x, y, z)$ if $x, y, z$ are distinct and, when travelling clockwise around the unique circle that contains them, after passing through $x$ then $y$ is reached before $z$. This relation is called the **clockwise ordering**, and satisfies the five conditions above.

An example is shown in Figure 3; consider the clockwise ordering on the set of six points in the plane. In the resulting CC-system, the open half-spaces corresponding to the straight line $\{a, b\}$ are $\{e\}$ and $\{c, d, f\}$; the convex hull of the set $\{a, b, c, d\}$ is the set $\{a, b, c, d, f\}$. It is always the case that the convex geometry a set inherits from the plane is encoded by its clockwise ordering: the geometry is the geometry of the CC-system.

Given, then, a set of "planar points" (pairs of floating point numbers, say) in a computer, one regards these not as embedded in the plane, but rather as points of a CC-system, and computes their geometry accordingly.

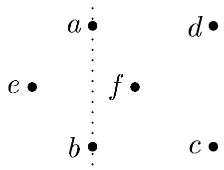


Figure 3: Clockwise ordering relation

Of course, one does not actually *store* (in principle possible, though in practice entirely infeasible) the CC-sytem in the computer memory, any more than the plane is ever stored (though this of course is impossible); one instead sets up a means of computing its structure. For planar points $x = (x_1, x_2)$, $y = (y_1, y_2)$ and $z = (z_1, z_2)$, let

$$det(x, y, z) = \begin{vmatrix} x_1 & x_2 & 1 \\ y_1 & y_2 & 1 \\ z_1 & z_2 & 1 \end{vmatrix} = x_1(y_2 - z_2) - y_1(x_2 - z_2) + z_1(x_2 - y_2),$$

and let $B(x, y, z)$ if $det(x, y, z) > 0$. This indeed gives a CC-system, and is the clockwise ordering were the planar points embedded in the plane.

It might be useful to summarize what is going on here. Knuth is focussing on one property of the plane, that of the clockwise order, and disregarding all others. This order can be computed via computing determinants of matrices, and its abstract properties form a purely combinatorial geometry – a CC-system. Geometric algorithms, such as a convex hull algorithm (and, subsequently, a Delaunay triangulation algorithm, which is based on an extension of the clockwise order), are designed in terms of this property alone, and Knuth states that this leads to significantly improved algorithm design.

As Knuth then observes, CC-sytems form a certain class of *oriented matroid*; in fact the axioms for a CC-system are close to the so-called *chirotope* axioms for oriented matroids. Oriented matroids admit many different axiomizations, and in section 5 we will give one of the most standard of these. In preparation for this, the next section introduces matroids.

## 4 Matroids

Oriented matroids can be described as matroids with additional structure. In this section we review some of the basic concepts of matroid theory. For the full story see the book [Oxl92].

The concept of a vector space is one of the basic building blocks underlying the algebraic approach to GIS proposed by Frank [Fra99, p101]. The insight of Whitney, the originator of matroid theory in the 1930s, was that one can throw away the vector space, keeping only the structure captured by the matroid, and still have a good theory of linear geometry. Thus, one view of matroids is that they abstract the notion of linear independence from vector space theory. However, matroids also capture the structure of affine geometry, many aspects of graph theory, and several other areas.

## 4.1 Definition and key concepts

A **matroid** $M$ is a finite set $E$ together with a collection $\mathcal{I}$ of subsets of $E$, called **independent sets**, such that:

$(I1)$ $\varnothing$ is independent;

$(I2)$ Every subset of an independent set is independent;

$(I3)$ For any independent sets $I, J$ such that $I$ has more elements than $J$, there exists some $x \in I \backslash J$ such that $J \cup x$ is independent.

The following are, very briefly, the most fundamental concepts in matroid theory. A **basis** of $X \subseteq E$ is any of its maximal independent subsets. It can be shown that all bases have the same number of elements. The **rank** of $X$, denoted $r(X)$, is the cardinality of any of its bases; the **closure** is defined as $\langle X \rangle = \{e \in E \mid r(X) = r(X \cup e)\}$. A **closed set** is any set that is equal to its closure. A **dependent set** is any set that is not independent, and a **circuit** is any minimal dependent set.

Equivalent axiomatizations of matroids exist in many different forms. To understand the relationship of matroids to oriented matroids, we give the description in terms of circuits. A collection of subsets, $\mathcal{C} \subseteq \mathcal{P}(E)$, is the collection of circuits of a matroid iff

$(C1)$ $\varnothing \notin \mathcal{C}$;

$(C2)$ $C_1, C_2 \in \mathcal{C}$, $C_1 \subseteq C_2 \Rightarrow C_1 = C_2$;

$(C3)$ $C_1, C_2 \in \mathcal{C}$, $C_1 \neq C_2$, $x \in C_1 \cap C_2 \Rightarrow \exists C_3 \in \mathcal{C}.\ C_3 \subseteq (C_1 \cup C_2) \backslash x$.

A fundamental aspect of matroid theory is that of duality. A **hyperplane** is any closed set $H$ such that $r(H) + 1 = r(E)$, and the complement of a hyperplane is called a **cocircuit**. The set of all cocircuits satisfy the matroid circuit axioms, and so form another matroid on the same ground set as the original matroid. This is called the **dual matroid**. Every matroid is the dual of its dual.

## 4.2 Affine Matroids

One class of examples of matroids that is particularly important for our purposes is that based on the notion of affine independence. In the plane

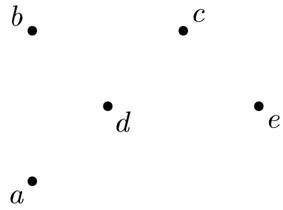


Figure 4: Five points in the plane.

$\mathbb{R}^2$, the **affine sets** (also called **flats** or **closed sets**) are: the empty set, the points themselves, the straight lines, and the plane itself. The **affine hull** of any set is the smallest affine set that contains it. A set is **affine independent** iff its affine hull is not the affine hull of any of its proper subsets. Given a set $E$ of points in the plane, we can construct a matroid, called the **affine matroid** on $E$, by taking the independent sets to be the sets of points which are affine independent.

For example, consider the set of points in the plane shown in figure 4. The independent sets in the inherited matroid are all the ($\leq$ 3)-element subsets except $\{a, c, d\}$. The circuits are $\{a, c, d\}$, $\{a, b, d, e\}$, $\{b, c, d, e\}$, and the hyperplanes are $\{a, b\}$, $\{a, c, d\}$, $\{a, e\}$, $\{b, c\}$, $\{b, d\}$, $\{b, e\}$, $\{c, e\}$, $\{c, d\}$. The bases are all the 3-element subsets except $\{a, c, d\}$. The affine sets are: $\varnothing$; all the singleton subsets; $\{a, c, d\}$ and all the doubleton subsets except $\{a, c\}$, $\{a, d\}$, $\{c, d\}$; $\{a, b, c, d, e\}$.

### 4.3 Graphic Matroids

A non-empty collection of edges in an undirected graph is a **circuit** if it can be ordered $e_0, \ldots, e_k$ such that there exist distinct vertices $v_0, \ldots, v_k$ such that $e_i$ is an edge between $v_i$ and $v_{i+1}$ (indices considered $mod\ k+1$). A set of edges is independent if it contains no circuit (these sets are called **forests** in graph theory; a connected forest is a **tree**), and these independent sets satisfy the matroid axioms. In the example shown in figure 5, the circuits are $e$, $ab$, $adf$, $bdf$. The cocircuits of a graphic matroid are its minimal cut-sets, which in the example are $c$, $df$, $abd$, $abf$.

The bases of a graphic matroid of a connected graph are its spanning trees. This is one example of the (quite extensive) use of matroids in computer science and spatial computation, particularly in the area of combinatorial optimization. A good basic reference is [CC$^+$98].

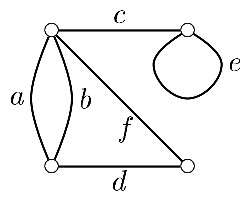


Figure 5: Graph to illustrate construction of graphic matroid

# 5 Oriented matroids

In this section we sketch the main concepts for oriented matroids. For a detailed study, the book [BVS+99] is a valuable source.

Vector spaces do not in general have enough structure to support a theory of convexity; it is only vector spaces over ordered fields that do. We can therefore hardly expect matroids to have any theory of convexity. One view is that an oriented matroid is a matroid plus convexity.

## 5.1 Definition and Key Concepts

To define oriented matroids we need the concept of a signed set. A **signed set** is any pair $X = (X^-, X^+)$ of disjoint sets, which we will call its **parts**. Its **opposite** is the signed set $-X = (X^+, X^-)$, and its **support** is the set $\underline{X} = X^- \cup X^+$. A signed set is said to be a **signature** of its support. A **circuit signature** of a matroid is a collection of signatures of its circuits that contains exactly two signatures of each circuit, and these two are opposite.

An **oriented matroid** is a matroid together with a circuit signature such that, for any signed circuits $X \neq -Y$, and any $e \in X^+ \cap Y^-$, there is a signed circuit $Z$ such that $Z^+ \subseteq (X^+ \cup Y^+)\backslash e$ and $Z^- \subseteq (X^- \cup Y^-)\backslash e$.

As with matroids, duality is a fundamental aspect of the theory. Two signed sets are **orthogonal** if either their supports are disjoint or one part of one intersects both parts of the other. Given any oriented matroid and any cocircuit of its underlying matroid, there is, up to opposites, a unique signature of the cocircuit that is orthogonal to every signed circuit. The resulting collection of cocircuit signatures is an orientation of the dual matroid, and so we have another oriented matroid, called the **dual** of the original. Every oriented matroid is the dual of its dual.

## 5.2 Affine Oriented Matroids

Consider the affine matroid on a finite subset $E$ of Euclidean space (or any vector space over an ordered field). Any circuit admits a unique partition, called a **Radon partition**, into two sets such that the Euclidean convex hulls of these sets intersect. Taking, for each circuit, the pair of opposite

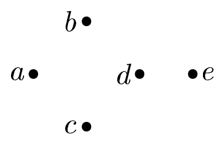


Figure 6: Illustrating construction of affine oriented matroid

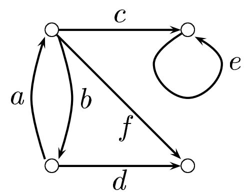


Figure 7: Illustrating construction of graphic oriented matroid

signed sets determined by its Radon partition gives an oriented matroid, called the **affine oriented matroid** [1]

In figure 6, the signed circuits in the affine oriented matroid are $(ae, d)$, $(ad, bc)$, $(ae, bc)$, $(bce, d)$ and their opposites. The signed cocircuits are $(abd, \emptyset)$, $(acd, \emptyset)$, $(ab, e)$, $(a, de)$, $(ac, e)$, $(b, c)$, $(bde, \emptyset)$, $(cde, \emptyset)$ and their opposites.

### 5.3 Graphic oriented matroids

A non-empty collection of edges in a directed graph is a circuit if it can be ordered $e_0, \ldots, e_k$ such that there exist distinct vertices $v_0, \ldots, v_k$ such that $e_i$ is an edge from $v_i$ to $v_{i+1}$ or from $v_{i+1}$ to $v_i$ (indices considered $mod\ k+1$). For each circuit, consider the signature $(\{e_i \mid e_i$ an edge from $v_i$ to $v_{i+1}\}$, $\{e_i \mid e_i$ an edge from $v_{i+1}$ to $v_i\})$ and its opposite. This gives an oriented matroid.

In figure 7, the signed circuits in the graphic oriented matroid are $(e, \emptyset)$, $(ab, \emptyset)$, $(af, d)$, $(bd, f)$ and their opposites. The signed cocircuits are $(c, \emptyset)$, $(df, \emptyset)$, $(ad, b)$, $(a, bf)$ and their opposites.

## 6 Applying Oriented Matroids to GIS

In this section we identify ways in which oriented matroids might be applied in GIS. In each case further work will be required before the full details of the oriented matroid approach are available and can be evaluated. However,

[1] The term **affine oriented matroid** is used in the literature for a quite different structure on $E$.

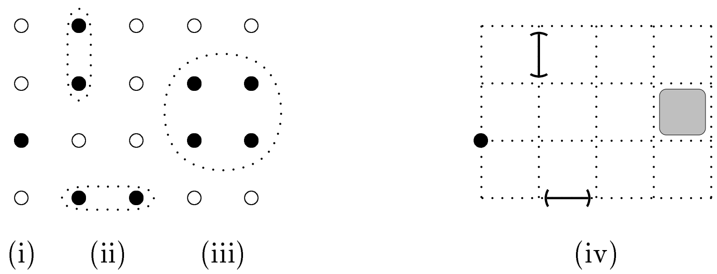

Figure 8: Continuous Part

there is sufficient evidence to justify the claim that these are important research directions which should be pursued.

### 6.1 Conventional Background Space

The most straightforward application of oriented matroids in GIS would be to represent the geometrical structure of a collection of discrete points in a similar way to the grid of points in a realm. Geographical objects having point locations could then be referenced to elements of the matroid. Geographical entities corresponding to regions or linear features could also be represented, but to do so we need to show how certain subsets of (certain kinds of) oriented matroids can be thought of as standing for regions in continuous space.

Using an oriented matroid to encode the affine structure of a discrete grid, we can think of the elements of the matroid as points. The open interval between a pair of adjacent points can then be represented by the pair of corresponding elements of the matroid, and the interior of a square bounded by four pairs of adjacent points can be represented by the four elements representing these points. To give a detailed treatment here we would need to introduce cell complexes, but the picture in figure 8 coveys the main idea. The cells in the grid of points on the left are: (i) every point; (ii) every vertically or horizontally adjacent pair of points; (iii) every square of four points as shown. Diagram (iv) shows the **continuous part** of each set, that is, the subset of continuous space which it represents.

Equipped with this notion of continuous part, we can represent a region by a set of cells, each cell being represented by a set of 1, 2, or 4 elements of the oriented matroid. Such a set of cells need not be a crisp subset; we could use a rough description making the distinction between cells lying wholly within a continuous region and those which lie partly in and partly outside the region. This suggests that concepts from rough set theory already used

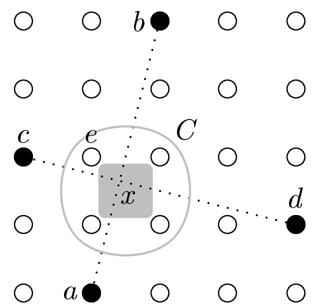


Figure 9: Handling incomplete information

in GIS [BS01] can be adapted to the oriented matroid setting.

## 6.2 Relational Space

Oriented matroids need not be used to support a kind of space independent from the objects about which we record geographical information. An alternative is to identify the elements of the matroid not with idealized locations, but with particular entities, such as houses, trees, and fish markets. Such an approach would encode certain geometrical relationships between the entities without needing to situate them first within some space. This could provide a relational representation of space, as discussed in the introduction to this paper.

## 6.3 Handling Incomplete Information

Geographical data is inherently inaccurate and incomplete [Wor98, Goo01]. Goodchild [Goo01, p3] advocates an *'object-centered'* approach in which point locations are *'conceptualized as located at the center of a circle of possibility in continuous space'* rather than identified with the nearest point in a discrete grid. We argue that using oriented matroids allows an approach which is intermediate between what Goodchild advocates and the method of realms championed by Schneider [Sch97].

We agree that realms are too rigid – forcing locations to fit into a predetermined grid – but argue that implemented space cannot be the ideal continuous space imagined by Goodchild. The solution is to replace continuous space by an oriented matroid, but to allow regions of possibility, represented by subsets of the matroid. An example of how we can avoid some of the problems which realms encounter, is given in figure 9 which concerns the intersection of two lines (i.e. affine sets of rank 2) in the inherited oriented matroid on a grid of points in $\mathbb{R}^2$.

Figure 9 shows the lines $\{a, b\}$ and $\{c, d\}$. Now these lines "cross", but there is no point in the oriented matroid that witnesses this: the Euclidean point of intersection $x$ does not exist in the oriented matroid. Now consider the set $C$ of 4 points outlined: $x$ lies in the continuous part of $C$, which is the shaded open square indicated. Rather than round $x$ to its nearest point in the oriented matroid, which is $e$, and call *this* the point of intersection, we might instead say that the lines "intersect" at $C$. Instead of giving erroneous information, namely that the lines intersect at $e$, we are choosing to give *incomplete* information, namely that the point of intersection lies somewhere in the continuous part of $C$.

### 6.4 Triangulations

We have already indicated that regarding elements of an oriented matroid as points, it is possible to represent regions by sets of points. To develop this approach further, it will be necessary to investigate the geometry of such regions. One way of developing the theory of regions within oriented matroids is via triangulations of oriented matroids [San02]. Some initial steps to providing a 'geometry of triangulations' for applications to GIS have already been taken by Webster [Web02].

Further motivation for the study of triangulations of oriented matroids is provided by the use of triangulated irregular networks in GIS [Jon97, p200] and by the possibility of using refinements of triangulations to handle multiresolution spatial data [FMP99].

### 6.5 Unified Data Model

Winter [Win98] has already noted the possibility of using cell complexes to provide a unified data model encompassing the merits of both the raster and vector approaches. The possibility of using oriented matroids to handle not only points but also regions of continuous space should allow a space built from matroids to function in a similar way.

## 7 Conclusions and Further Work

In this paper we have considered oriented matroids from the viewpoint of the spatial needs of geographical information science (GIS). In providing a brief tutorial introduction to theory of oriented matroids (sections 3–5) we have only been able to give the most basic concepts and examples. The main contribution of the paper is to have identified, in section 6, particular GIS issues which can be addressed by oriented matroids. All of these application areas need additional research before the potential of oriented matroids for GIS can be fulfilled – the development of all these areas constitutes further work that the authors are currently undertaking. It would be possible to identify extra application areas, but we have restricted ourselves in section 6 to those where there there is already sufficiently good evidence for the viability of an oriented matroid approach. One more speculative topic that deserves mention is the variation of geographical data over time. It would be worthwhile investigating whether the idea of a set of oriented matroids varying over a space, as in the theories of combinatorial differentiable manifolds and matroid bundles [And99], has the appropriate structure to address the challenges of this topic.

## References


[And99] L. Anderson. Matroid bundles. In L. J. Billera, A. Björner, et al., editors, *New Perspectives in Algebraic Combinatorics*, pages 1–21. Cambridge University Press, 1999.

[BS01] T. Bittner and J. G. Stell. Rough sets in approximate spatial reasoning. In W. Ziarko and Y. Yao, editors, *Proceedings of Second International Conference on Rough Sets and Current Trends in Computing (RSCTC'2000)*, volume 2005 of *Lecture Notes in Computer Science*, pages 445–453. Springer-Verlag, 2001.

[BVS+99] A. Björner, M. Las Vergnas, B. Sturmfels, N. White, and G. Ziegler. *Oriented Matroids.* Cambridge University Press, 1999.

[CC+98] W. H. Cook, W. J. Cunningham, et al. *Combinatorial Optimization.* Wiley, 1998.

[Coh96] A. G. Cohn. Calculi for qualitative spatial reasoning. In J.A. Campbell J. Calmet and J. Pfalzgraf, editors, *Artificial Intelligence and Symbolic Mathematical Computation*, Lecture Notes in Computer Science 1138, pages 124–143. Springer-Verlag, 1996.

[FMP99] L. De Floriani, P. Magillo, and E. Puppo. Data structures for simplicial multi-complexes. In R. Gütting, D. Papadias, and F. Lochovsky, editors, *Advances in Spatial Databases*, volume 1651 of *Lecture Notes in Computer Science*, pages 33–51. Springer-Verlag, 1999.

[Fra99] A. U. Frank. One step up the abstraction ladder: Combining algebras – From functional pieces to a whole. In C. Freksa and D. Mark, editors, *Spatial Information Theory. COSIT'99 Proceedings*, volume 1661 of *Lecture Notes in Computer Science*, pages 95–107. Springer-Verlag, 1999.

[Gal99] A. Galton. The mereotopology of discrete space. In C. Freksa and D. Mark, editors, *Spatial Information Theory. COSIT'99*

*Proceedings*, volume 1661 of *Lecture Notes in Computer Science*, pages 251–266. Springer-Verlag, 1999.

[Goo01] M. F. Goodchild. A geographer looks at spatial information theory. In D. R. Montello, editor, *Spatial Information Theory. COSIT 2001 Proceedings*, volume 2205 of *Lecture Notes in Computer Science*, pages 1–13. Springer-Verlag, 2001.

[Jon97] C. B. Jones. *Geographical Information Systems and Computer Cartography.* Longman, 1997.

[Knu91] D. E. Knuth. *Axioms and Hulls.* Lecture Notes in Computer Science 606. Springer-Verlag, 1991.

[KR91] T. Y. Kong and A. Rosenfeld. Digital topology: A comparison of the graph-based and topological approaches. In G. M. Reed, A. W. Roscoe, and R. F. Wachter, editors, *Topology and Category Theory in Computer Science*, pages 273–289. Clarendon Press, Oxford, 1991.

[Nie02] J. A. Nieto. Searching for a connection between matroid theory and string theory. http://arXiv.org/abs/hep-th/0212100, 2002.

[Oxl92] J. Oxley. *Matroid Theory.* Oxford University Press, 1992.

[San02] F. Santos. Triangulations of oriented matroids. *Mem.Amer.Math.Soc*, 156(741):80 pp., 2002.

[Sch97] M. Schneider. *Spatial Data Types for Database Systems: Finite Resolution Geometries for Geographic Information Systems.* Lecture Notes in Computer Science 1288. Springer-Verlag, 1997.

[Smo00] L. Smolin. *Three Roads to Quantum Gravity.* Phoenix, 2000.

[Web02] J. Webster. Combinatorial properties of triangulations of oriented matroids. Submitted for publication, 2002.

[Win98] S. Winter. Bridging Vector and Raster Representations in GIS. In R. Laurini *et al*, editor, *Proceedings of the 6th International Symposium on Advances in GIS*, pages 57–62. ACM Press, 1998.

[Wis02] S. Wise. *GIS Basics.* Taylor and Francis, 2002.

[Wor98] M. F. Worboys. Imprecision in finite resolution spatial data. *GeoInformatica*, 2:257–279, 1998.