To appear in the Proceedings of the 2nd ICLP Workshop on Deductive Databases, Italy. Publication Series of GMD Society. 1994.

Constraint-Based Reasoning in Geographic Databases: the Case of Symbolic Arrays

Dimitris Papadias[1] Andrew Frank[1] Manolis Koubarakis[2]

[1]Department of Geoinformation,
Technical University of Vienna,
Gusshausstrasse 27-29, Austria A-1040
dp@ithaca.dbnet.ece.ntua.gr
afrank@fbgeo1.tuwien.ac.at

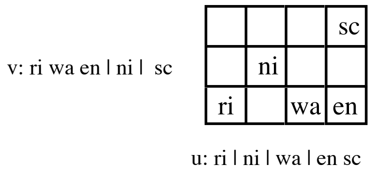
[2]IC-Parc,
Imperial College,
London SW7 2BZ, United Kingdom
msk@doc.ic.ac.uk

Abstract Symbolic arrays are hierarchical constraint-based representations that preserve direction relations (e.g., north, northeast) among the distinct components of complex spatial entities. They have been used in problems involving pattern matching and spatial information retrieval. In this paper we demonstrate how inference can be achieved in geographic databases of symbolic arrays using composition of direction relations. In particular, we distinguish two types of spatial inference: the first is concerned with the inference of constraints between objects that exist at different levels in the hierarchy, while the second type involves the inference of constraints between objects that exist at the same level but in different arrays.

1. Introduction

Constraint-based reasoning has recently received a lot of attention in the areas of Image Databases, Spatial Databases and GIS (Papadias and Sellis, 1992), (Freksa, 1992), (Egenhofer and Sharma 1993), (Hernandez, 1993). In this paper we concentrate on constraint-based spatial representations which derive from research on *analog* array structures, like *symbolic images* (Chang et al., 1987) and *symbolic arrays* (Papadias and Glasgow, 1991). A symbolic image is an array representing a spatial entity where each component (object) of the entity is denoted by one or more symbols.

Figure 1 illustrates a symbolic image representing a conceptual map of Britain and Ireland (*ri* stands for the Republic of Ireland, *ni* for Northern Ireland, *sc* for Scotland, *wa* for Wales and *en* for England). The symbolic image preserves direction constraints among its components (e.g., NorthEast(sc, ri), SouthWest(ri, sc), SouthEast(wa, ni)). Chang et al., (1987) developed the two-dimensional string (*2D string*) representation for encoding symbolic images. A 2D string is a pair of one dimensional strings (u, v) where u represents the symbolic projections of the objects on the x axis, and v represents the projections on the y axis. Figure 1 also illustrates strings u and v.



			sc
	ni		
ri		wa	en

Fig. 1 Symbolic image example

Symbolic arrays are hierarchical symbolic images that represent complex spatial entities in different levels (i.e., *aggregation hierarchies*). In this paper we deal with symbolic images and arrays where each object is represented by exactly one symbol per array. Arrays where objects are represented by one symbol have restricted expressive power that renders them inadequate for some practical applications. For instance, using the array of Figure 1 we cannot answer whether "*there are parts of England, which are North of some parts of Wales*". In order to have the expressive power to answer such queries we need arrays that use multiple symbols per object. Extensions to the expressive power of symbolic arrays to higher direction and topological resolution can be found in (Papadias and Sellis, 1993).

This paper concentrates on the inference of spatial information not explicitly stored in geographic databases of symbolic arrays using composition of direction relations. Section 2 discusses the representational properties of symbolic arrays and describes how direction constraints are represented. Section 3 is concerned with composition of spatial relations in hierarchical representations while Section 4 demonstrates how composition can be applied to infer direction relations between objects that exist in different arrays. Section 5 discusses the advantages of hierarchical spatial representations and Section 6 concludes with suggestions for further extensions.

2. Geographic Databases of Symbolic Arrays

A *symbolic array* is a hierarchical array of symbols that represents direction constraints among the distinct objects that comprise a spatial entity. In the context of this paper, a *geographic database* is a collection of symbolic arrays each corresponding to a conceptual map of a geographic region. The hierarchical structure is very important in several application domains, like geographic applications, where spatial entities are combined to form more general ones. In our example database, for instance, we have a symbolic array representing the map of continents at the top level, then each continent is subdivided in symbolic arrays representing sub-continents; each sub-continent is further decomposed in sub-arrays representing countries, each country in arrays representing states and so on.

Figure 2 illustrates a symbolic array (we call it *we*) representing a conceptual map of Western Europe (*ir* is for Ireland, *br* for Britain, *fr* for France, *sp* for Spain and *po* is for Portugal). Each of the objects that exist in *we* may be decomposed in simpler arrays (Britain for instance is decomposed in Northern Ireland, Scotland, Wales and England, while Ireland is decomposed in Republic of Ireland and Northern Ireland). Furthermore *we* itself is an object in a symbolic array representing the European continent. It is allowed to have *common objects* that exist in more than one arrays (multiple hierarchies). For instance, Northern Ireland exists in both symbolic arrays representing Britain and Ireland. In general, although we will not discuss the construction of symbolic arrays in this paper, we assume that the structure of symbolic arrays corresponds to the actual structure of the domain to be modeled (Papadias and Glasgow, 1991). Furthermore as we will see in the next sections, the choice of common objects is also important for the expressive power and the inferential adequacy of the system.

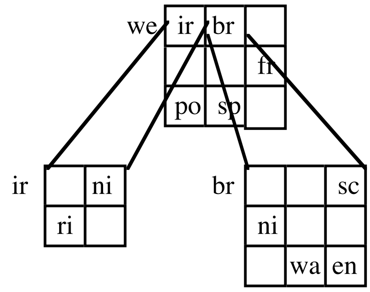


Fig. 2 Symbolic array example

The individual cells of a symbolic array S are denoted with subscripts; S_{ij} denotes the cell at row i and column j (S_{ij} and S_{kl} refer to the same cell iff i=k and j=l). Each cell of a symbolic array can be empty, or it can be occupied by one symbol denoting an object. The predicate $S_{ij}(P)$ denotes that cell S_{ij} contains object P. R(S,P,Q) denotes that the constraint R between symbolic object representations P and Q is satisfied in the symbolic array representing object S. For instance, NorthWest(we, br, fr) denotes that Britain is NorthWest of France in the symbolic array of West Europe.

Arrays that use one symbol per object, such as the ones in Figure 2, represent a set $\mathbf{D_1}$ of mutually exhaustive and pair-wise disjoint binary constraints. These constraints have been called *primitive direction relations* (Papadias and Sellis, 1994) and correspond to the *cardinal directions with neutral area* defined in (Frank, 1991).

NorthWest(S,P,Q) ... $\exists S_{ij}\exists S_{kl}\ (S_{ij}(P) \,|\, S_{kl}(Q) \,|\, i>k \,|\, j<l)$

RestrictedNorth(S,P,Q) ... $\exists S_{ij}\exists S_{kl}\ (S_{ij}(P) \,|\, S_{kl}(Q) \,|\, i>k \,|\, j=l)$

NorthEast(S,P,Q) ... $\exists S_{ij}\exists S_{kl}\ (S_{ij}(P) \,|\, S_{kl}(Q) \,|\, i>k \,|\, j>l)$

RestrictedWest(S,P,Q) ... $\exists S_{ij}\exists S_{kl}\ (S_{ij}(P) \,|\, S_{kl}(Q) \,|\, i=k \,|\, j<l)$

RestrictedEast(S,P,Q) ... $\exists S_{ij}\exists S_{kl}\ (S_{ij}(P) \,|\, S_{kl}(Q) \,|\, i=k \,|\, j>l)$

SouthWest(S,P,Q) … $\exists S_{ij}\exists S_{kl}\ (S_{ij}(P) \mid S_{kl}(Q) \mid\ i<k\ \mid\ j<l)$

RestrictedSouth(S,P,Q) … $\exists S_{ij}\exists S_{kl}\ (S_{ij}(P) \mid S_{kl}(Q) \mid\ i<k\ \mid\ j=l)$

SouthEast(S,P,Q) … $\exists S_{ij}\exists S_{kl}\ (S_{ij}(P) \mid S_{kl}(Q) \mid\ i<k\ \mid\ j>l)$

Figure 3 illustrates the primitive direction constraints among object symbols in a symbolic array. The symbol Q in Figure 3 denotes the *reference*[1] object and the other symbols refer to the direction constraint depending on the position of the *primary* object in the symbolic array (in some cases instead of the full name of the relation we use only the capital letters e.g., NW instead of NorthWest).

NorthWest	RestrictedNorth	NorthEast
RestrictedWest	**Q**	RestrictedEast
SouthWest	RestrictedSouth	SouthEast

Fig. 3 Direction constraints encoded in symbolic arrays

In addition to direction constraints, symbolic arrays capture the *inclusion* and *containment* relations i.e., *IN(Q,P)… ∃* $P_{ij}(P_{ij}(Q))$ and *ContaiNs(Q,P)… IN(P,Q)*. Although we used a hierarchical illustration for visualization purposes in the example of Figure 2, each symbolic array is in fact an object whose name appears in the array representing the parent object, and whose components are names of symbolic arrays in a lower level of aggregation. Symbolic arrays, symbolic images and equivalent 2D string encodings have been used to answer efficiently queries regarding the constraints between objects that exist in the same array (e.g., Chang et al., 1987). In the rest of the paper we will show how we can compute direction constraints between objects that exist in different arrays of the database.

3. Composition of Direction Relations

In order to achieve spatial constraint propagation we will use composition of spatial relations. The problem of composition can be defined as "if the spatial relation between objects X and Z, and between Z and Y is known what are the possible relations between X and Y?". *Composition tables* are usually used to describe the results of composition; Freksa (1991), for instance, illustrates a composition table for relations in 1D space and Egenhofer (1991) a composition table for topological relations in 2D space.

Composition of two primitive constraints does not always yield a constraint of $\mathbf{D_1}$, but may result in a constraint of lower resolution. We will define a set $\mathbf{D_2}$ of low resolution directions using disjunctions of primitive constraints:

North(S,P,Q) … NW(S,P,Q)ʃRN(S,P,Q)ʃNE(S,P,Q)

East(S,P,Q) … NE(S,P,Q)ʃRE(S,P,Q)ʃSE(S,P,Q)

[1] The term primary object denotes the object to be located and the term reference object denotes the object in relation to which the primary object is located.

South(S,P,Q) … SW(S,P,Q)∫RS(S,P,Q)∫SE(S,P,Q)
West(S,P,Q) … NW(S,P,Q)∫RW(S,P,Q)∫SW(S,P,Q)
SameLevel(S,P,Q) … RW(S,P,Q)∫RE(S,P,Q)
SamewidtH(S,P,Q) … RN(S,P,Q)∫RS(S,P,Q)

The directions SameLevel, SamewidtH of $\mathbf{D_2}$ and the directions RestrictedNorth, RestrictedEast, RestrictedSouth, RestrictedSouth of $\mathbf{D_1}$ are called *restricted directions*. Let **D** be the union of $\mathbf{D_1}$ and $\mathbf{D_2}$; the following lattices represent the direction constraints of **D**. The first lattice represents north-south direction, while the second one represents west-east direction.

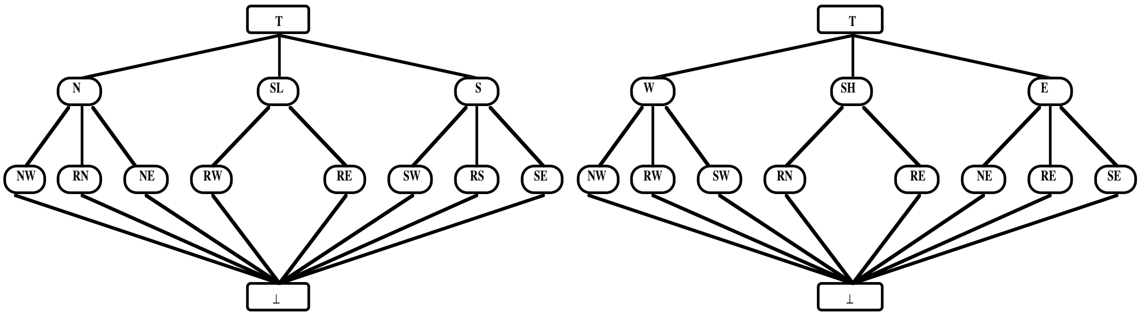


Fig. 4 Lattices representing direction constraints

R(P,Q) denotes that the direction constraint R (R ⌊ **D**) between objects P and Q is satisfied in the database:

R(P,Q) … [∃S(R(S,P,Q)]∫

[∃O(R_1(P,O) R_2(O,Q)) | (R_1(X,Z) ∗p (R_2(Z,Y) = R(X,Y))] [2]

That is, the database satisfies the direction constraint R between P and Q if:

1. there is an object S such that the direction R between objects P and Q is satisfied in the symbolic array representing S or,
2. there is an object O such that the direction R_1 between objects P and O, and the direction R_2 between O and Q is satisfied in the database, and there is a composition rule R_1(X,Z) $*_p$ R_2(Z,Y) = R(X,Y).

The direction constraints between objects that exist in the same array are directly extracted using operations that scan the array elements and retrieve the corresponding relation. Composition is aimed only at the retrieval of directions between objects that exist in different maps. Table 1 describes the composition rules that can be applied in order to produce the possible direction constraints between objects that exist in different arrays. The table extends the composition table for directions with neutral area in (Frank, 1992) by including the constraints of $\mathbf{D_2}$, as well as, "hierarchical composition" using aggregation hierarchies. Furthermore, unlike Frank, who uses the notion of *Euclidean approximation* in cases where

[2] The symbol $*_p$ denotes *path composition*; for details see (Frank, 1992).

there is uncertainty, we use relations of lower direction resolution (i.e., disjunctions of primitive relations). A similar approach was taken by Egenhofer (1991) in defining composition for topological relations in 2D space.

		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
		NW	**RN**	**NE**	**RW**	**RE**	**SW**	**RS**	**SE**	**N**	**E**	**S**	**W**	**SL**	**SH**	**CN**	**IN**
1	**NW**	NW	NW	N	NW	N	W	W	T	N	T	T	W	N	W	NW	T
2	**RN**	NW	RN	NE	NW	NE	W	SH	E	N	E	T	W	N	SH	N	T
3	**NE**	N	NE	NE	N	NE	T	E	E	N	E	T	T	N	E	NE	T
4	**RW**	NW	NW	N	RW	SL	SW	SW	S	N	T	S	W	SL	W	W	T
5	**RE**	N	NE	NE	SL	RE	S	SE	SE	N	E	S	T	SL	E	E	T
6	**SW**	W	W	T	SW	S	SW	SW	S	T	T	S	W	S	W	SW	T
7	**RS**	W	SH	E	SW	SE	SW	RS	SE	T	E	S	W	S	SH	S	T
8	**SE**	T	E	E	S	SE	S	SE	SE	T	E	S	T	S	E	SE	T
9	**N**	N	N	N	N	N	T	T	T	N	T	T	T	N	T	N	T
10	**E**	T	E	E	T	E	T	E	E	T	E	T	T	T	E	E	T
11	**S**	T	T	T	S	S	S	S	S	T	T	S	T	S	T	S	T
12	**W**	W	W	T	W	T	W	W	T	T	W	T	T	T	W	W	T
13	**SL**	N	N	N	SL	SL	S	S	S	S	T	N	T	SL	T	T	T
14	**SH**	E	T	W	E	W	E	T	W	T	E	T	W	T	SH	T	T
15	**CN**	T	T	T	T	T	T	T	T	T	T	T	T	T	T	CN	T
16	**IN**	NW	N	NE	W	E	SW	S	SE	N	E	S	W	T	T	T	IN

Table. 1 Composition Table

Composition can be divided in two cases. The first case involves composition of constraints between objects that exist at different levels in the hierarchy through *inclusion* and *containment* relations (i.e., the last two rows and columns of the composition table). The second case involves composition of constraints between objects that are in the same level of aggregation and are connected through a chain of common objects. In the next section we will demonstrate examples of composition in geographic databases of symbolic arrays and we will discuss potential problems that could arise from the improper use of aggregation hierarchies.

4. Spatial Constraint Propagation in Symbolic Arrays

Consider the query “*What is the relation between Scotland and France*Ó? Since Scotland and France do not belong to the same array the direction constraint between them cannot be immediately retrieved. From the facts that Scotland is IN Britain and Britain is NorthWest of France and the entry (16, 1) of the composition table we can conclude that Scotland is NorthWest of France. In general, each object inherits the constraints of the corresponding ancestor with respect to the objects at the ancestor's level in the hierarchy. In case of the restricted directions though (i.e., RN, RE, RS, RW, SL, SH) the object inherits the direction constraint of the next level of resolution:

IN(X,Z) $*_p$ RE(Z,Y) = E(X,Y), RE(X,Z) $*_p$ CN(Z,Y) = E(X,Y),
IN(X,Z) $*_p$ RS(Z,Y) = S(X,Y), RS(X,Z) $*_p$ CN(Z,Y) = S(X,Y),
IN(X,Z) $*_p$ RW(Z,Y) = W(X,Y), RW(X,Z) $*_p$ CN(Z,Y) = W(X,Y),
IN(X,Z) $*_p$ RN(Z,Y) = N(X,Y), RN(X,Z) $*_p$ CN(Z,Y) = N(X,Y),
IN(X,Z) $*_p$ SL(Z,Y) = T(X,Y), SL(X,Z) $*_p$ CN(Z,Y) = T(X,Y),
IN(X,Z) $*_p$ SH(Z,Y) = T(X,Y), SH(X,Z) $*_p$ CN(Z,Y) = T(X,Y).

This approach in addition to being more intuitive, also avoids possible inconsistencies. For instance, if we allowed the inheritance of restricted constraints we could reach the conclusion that Northern Ireland is RestrictedEast of Scotland, by following the inference chain: (IN(ni, ir) $*_p$ RE(ir,br)) $*_p$ CN(br,sc) = RE(ni, br) $*_p$ CN(br,sc) = RE(ni, sc).

Hierarchical reasoning is not the only way to achieve spatial inference in symbolic arrays. Consider, for example, that we want to find the constraint between the Republic of Ireland and Scotland. Using hierarchical constraint propagation, we can conclude that Scotland is East of the Republic of Ireland because its parent (i.e., Britain) is RestrictedEast from the parent of the Republic of Ireland (i.e., Ireland). On the other hand, if we take advantage of the common object (i.e., Northern Ireland) we can infer that Scotland is NorthEast of the Republic of Ireland using NE(sc, ni) $*_p$ NE(ni, ir) (i.e., entry (3,3) of the composition table). That is, the proper use of common objects results in an increase of the direction resolution for the composition direction (i.e., NorthEast instead of East).

Similarly we can infer that England is East of the Republic of Ireland using SE(en, ni) $*_p$ NE(ni, ir) (i.e., entry (3,3) of the composition table). The result of the composition using common objects may consist of a constraint of $\mathbf{D_1}$ as in the first inference (i.e., NE(sc, ri)), of a constraint of $\mathbf{D_2}$ as in the second inference (i.e., E(en, ri)), or of the disjunction of all primitive constraints (i.e., the composition does not yield new information). The entries of the table involving composition of constraints of $\mathbf{D_2}$ are needed for the cases that the symbolic images containing the initial objects are not connected through a single object, but through a chain of common objects and an intermediate composition yields a constraint of lower resolution.

Potential problems could arise by the improper specification of multiple hierarchies. Since we allow an object to have more than one parents (e.g., Northern Ireland), the object may inherit some inconsistent constraints with respect to the objects at its level, or the objects at the parent level. In the structure of Figure 2, if we use Ireland as the parent array for Northern Ireland, we reach the conclusion that Northern Ireland is North of Portugal, while if we use

Britain as the parent we reach the conclusion that Northern Ireland is NorthEast of Portugal. Although the constraints in this case are not inconsistent (NorthEast North) we could reach inconsistent conclusions in other arrays. In the array structure of Figure 5, for instance, if we use Ireland as the parent array for Northern Ireland, we reach the conclusion that Northern Ireland is NorthWest of Portugal, while if we use Britain as the parent we reach the conclusion that Northern Ireland is NorthEast of Portugal (the relations NorthEast and NorthWest are inconsistent).

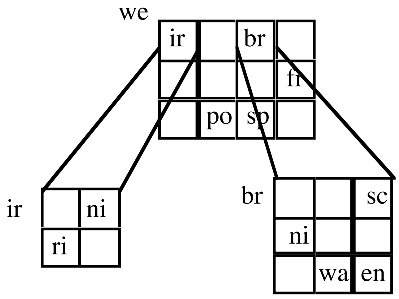


Fig. 5 Symbolic array that yields inconsistent relations

The problems with multiple hierarchies can be avoided by not permitting the existence of common objects during the construction of symbolic arrays. On the other hand, despite the potential problems, common objects are important because they allow the retrieval of high resolution constraints between objects that exist at the same level of aggregation. This feature is important to most geographical queries since they usually refer to objects in the same level at the hierarchy; it is for instance, more probable to have queries of the form "*is Scotland northeast of the Republic of Ireland*", than queries of the form "*is Glasgow north of Africa*". If we did not include Northern Ireland in the symbolic image of Britain, then we would loose the relations of $\mathbf{D_1}$ between Northern Ireland and the other states of Britain. Furthermore we would not be able reach the conclusion that Scotland is NorthEast of the Republic of Ireland, i.e., the resolution would be restricted to the constraints of $\mathbf{D_2}$ (and in particular the constraint East in this case). Instead of completely disallowing common objects during the construction of symbolic arrays, the common objects should be chosen in a way that avoids errors and maximizes resolution for the queries of interest.

5. On the Efficiency of Hierarchical Spatial Representations

The importance of analog spatial representations, due to the large number of implicit constraints in relatively compact representational structures, has been pointed out by several researchers with different perspectives on spatial knowledge representation e.g., (Lindsay, 1988), (Myers and Konolige, 1992). If n is the number of objects in a symbolic array, the maximum size of the array is n^2 cells, when there is exactly one object in each line and column. If we encode the array using 2D strings, i.e., one dimensional encodings as in Figure 1, then the maximum size becomes *2n*. On the other hand, if we used propositional representations for the same set of constraints we would need *n(n-1)* binary predicates. Furthermore the ordered structure of information in analog representations facilitates the retrieval of direction constraints.

The fragmentation of large "flat" arrays in smaller, hierarchical ones, reduces the overall storage requirements and facilitates efficiency in the retrieval of direction constraints within one array. If we decompose a flat array containing n objects (maximum size is *n2*) in m arrays, each containing *n/m* objects then the size of each array is $(n/m)^2$ and the total size is $n^2/m + n/m$. This number consists of the sum of the m arrays that are the result of the decomposition plus the size of the parent array (*we* in our example). The previous numbers refer to the worst case (exactly one object in each line and each column) and when we do not have redundancy (the existence of common objects). Although the addition of common objects during decomposition increases storage requirements, since decomposition can be continued for each of the resulting sub-arrays the efficiency is higher.

As an example consider the gazetteer of the United States. There are 128.000 *populated places*, grouped in 3700 counties that belong to 50 states. If we used a flat map to encode the direction constraints between the populated places we would need an array of 128.0002 = 16384x106 cells (or alternatively a 2D string of size 2x128.000 = 256000). On the other hand, we can use the hierarchical representation of Figure 6. Using this structure we have one array of 502 cells with the states at the top level. At the second level each of the 50 states is decomposed in an array of size 742 that contains its counties (on the average each state contains 74 counties). At the third level each county is decomposed in an array of size 352 that represents its populated places (on the average, each county contains 35 populated places). The total size of this structure is: 502 + (50x742) + (74x352) = 366950 cells (or alternatively a 2D string of size 2x50 + 50 x (2x74) + 74 x (2x35) = 11880).

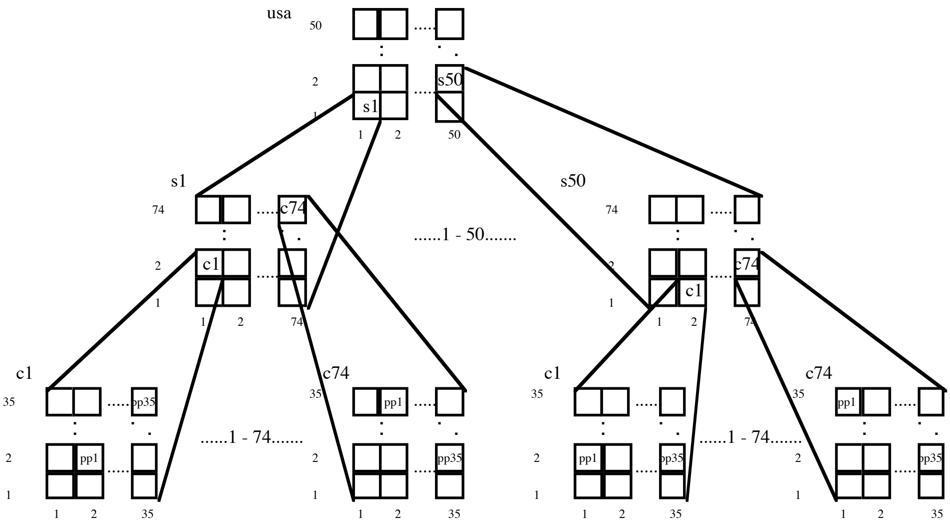


Fig. 6 A practical application

In addition to storage efficiency the hierarchical structure facilitates efficiency in information retrieval regarding the constraints between objects that exist in the same array. For example, queries of the form "*find all counties RestrictedNorth of Peuobscot County in the state of Maine*" involve time O(*n*) in 2D string implementations where n is the size of the string. Since after decomposition *n* becomes significantly smaller, spatial information retrieval involving objects that belong to the same geographic entity becomes much faster.

The gain in storage efficiency using hierarchical spatial representations comes at the expense of expressive power. Using the array of Figure 1, for instance, we can retrieve the constraint RestrictedEast between England and the Republic of Ireland. On the other hand, using the symbolic array of Figure 2 and inference through aggregation or common objects we can only infer that East(en, ri), that is, the inferred constraint is one of lower resolution. According to the definitions in (Papadias and Sellis, 1994) hierarchical spatial representations are not *complete* since they loose information with respect to "flat" representations. The *construction processes* that create hierarchical spatial representations from non-hierarchical ones are responsible for the proper fragmentation of data in order to achieve maximum efficiency with the minimum information loss for the important queries.

6. Conclusion

This paper deals with constraint-based reasoning in geographic databases of symbolic arrays. Symbolic arrays and related structures, like *symbolic images* and 2D strings have been used in applications including image databases (e.g., Chang et al., 1989) and spatial pattern matching, i.e., matching where similarity depends on the spatial relations among distinct objects, and not on geometric properties such as shape (e.g., Glasgow et al., 1992). The paper demonstrates how spatial inference can be applied to infer constraints between objects that exist in different arrays using composition of direction relations.

The direction constraints that we assumed in the paper can be classified in two categories, namely, the primitive constraints of high resolution (e.g., the constraints between objects that exist in the same array) and constraints of low resolution that are sometimes the result of the composition of two primitive constraints. Composition itself can be divided in two cases:

- composition of constraints between objects that exist at different levels in the hierarchy,
- composition of constraints between objects that are at the same level of aggregation and are connected through a chain of common objects.

Our approach extends previous work in composition of direction relations, e.g., (Frank 1992), (Hernandez, 1993), (Freksa, 1991), by involving two levels of direction resolution and aggregation hierarchies. The previous concepts are directly applicable to 3D space and can be used in applications involving CAD/CAM, Computer Vision etc. In 3D space the number of primitive direction relations is 26 (Figure 7) and the hierarchical representation of space results in more significant efficiency because the storage requirements grow exponentially with the number of dimensions.

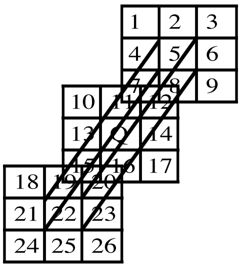


Fig. 7 Number of direction constraints in 3D space

One issue which we have not treated in depth in this paper is supplying a formal semantics for symbolic array structures and their operations. A formal theory of arrays provides a meta-language for specifying the symbolic array representation. Array theory is the mathematics of nested, rectangularly arranged data objects (More, 1981), while NIAL is a functional programming language based on array theory (Jenkins et al., 1986). Several functions that can be used to create symbolic arrays from other representations that store spatial knowledge (such as a frame database of complex objects), to modify symbolic arrays (e.g., rotate an array or move an object within an array) or extract information found in the array have been defined and implemented in NIAL (Glasgow and Papadias, 1992).

Further extensions can be made for reasoning with symbolic arrays that use more than one symbolic instances per object. Such arrays represent a large number of primitive direction constraints (Papadias and Sellis, 1994) and composition is more complicated. Another interesting topic is the composition of spatial constraints that involve both topological and direction information. Although these problems have been studied independently, to our knowledge there does not exist previous work that combines both approaches. Finally, the model of symbolic arrays can be extended with the ability to represent non-spatial information or can be integrated with an existing non-spatial database model. For the latter approach it may be beneficial to adopt the methodology of (Koubarakis, 1993) and (Koubarakis, 1994).

References

Chang, S.K, Shi, Q.Y.,Yan, C.W. (1987) Iconic Indexing by 2-D String. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-9, no 3, pp. 413-428.

Chang, S.K, Jungert, E., Li, Y. (1989) The Design of Pictorial Database upon the Theory of Symbolic Projections. In the Proceedings of the First Symposium on the Design and Implementation of Large Spatial Databases (SSD 89). Springer-Verlag.

Egenhofer, M. (1991) Reasoning about Binary Topological Relations. In the Proceedings of the Second Symposium on the Design and Implementation of Large Spatial Databases (SSD 91). Springer-Verlag.

Egenhofer, M., Sharma, J. (1993) Assessing the Consistency of Complete and Incomplete Topological Information. Geographical Systems, vol 1, 47-68.

Frank, A. U., (1991) Qualitative Spatial Reasoning about Cardinal Directions. In the Proceedings of the 10th International Symposium on Computer-assisted Cartography. Auto-Carto 10.

Frank, A. U., (1992) Qualitative Spatial Reasoning about Distances and Directions in Geographic Space. Journal of Visual Languages and Computing, 3, pp 343-371.

Freksa, C. (1991) Qualitative Spatial Reasoning. In Mark, D.M. and Frank, A.U (eds.) Cognitive and Linguistic Aspects of Geographic Space. Kluwer, Dordrecht.

Freksa, C. (1992) Using Orientation Information for Qualitative Spatial Reasoning. In Frank, A.U., Campari, I., Formentini, U. (eds.) Proceedings of the International Conference GIS - From Space to Territory: Theories and Methods of Spatio-Temporal Reasoning in Geographic Space. Springer-Verlag.

Glasgow, J.I, Papadias, D. (1992) Computational Imagery. Cognitive Science, 16, pp 355-394.

Glasgow, J.I., Fortier, S, Allen, F.H. (1992) Molecular Scene Analysis: Crystal Structure Determination through Imagery. In Hunter, L., (ed.) AI and Molecular Biology. AAAI Press.

Hernandez, D. (1993) Maintaining Qualitative Spatial Knowledge. In Frank, A.U., Campari, I. (eds.) Proceedings of the European Conference on Spatial Information Theory COSIT. Springer-Verlag.

Jenkins, M.A., Glasgow, J.I., McCrosky, C. (1986) Programming Styles in Nial. IEEE Software, 86, pp46-55.

Koubarakis, M. (1993) Representation and Querying in Temporal Databases: the Power of Temporal Constraints. Proceedings of the 9th International Conference on Data Engineering.

Koubarakis, M. (1994) Foundations of Indefinite Constraint Databases. Proceedings of the Workshop on Principles and Practice of Constraint Programming.

Larkin, J., Simon, H.A. (1987) Why a Diagram is (Sometimes) Worth Ten Thousand Words. Cognitive Science, 10, pp 65-99.

Levine, M. (1978) A Knowledge-Based Computer Vision System. In Hanson, A. and Riseman, E. (eds.) Computer Vision Systems. New York: Academic. pp 335-352.

Lindsay, R.K. (1988) Images and Inference. Cognition, 29, pp229-250.

More, T. (1981) Notes on the Diagrams, Logic and Operations of Array Theory. In Bjorke and Franksen (eds.) Structures and Operations in Engineering and Management Systems. Tapir Publishers.

Myers, K.,L., Konolige, K. (1992) Reasoning with Analogical Representations. In the Proceedings of the 3rd International Conference on Principles of Knowledge Representation and Reasoning, KR 92. Morgan Kaufman.

Papadias, D., Glasgow, J.I. (1991) A Knowledge Representation Scheme for Computational Imagery. Proceedings of the 13th Annual Conference of the Cognitive Science Society. Lawrence Erblaum Associates.

Papadias, D., Sellis, T. (1992) Spatial Reasoning using Symbolic Arrays. In Frank, A.U., Campari, I., Formentini, U. (eds.) Proceedings of the International Conference GIS - From Space to Territory: Theories and Methods of Spatio-Temporal Reasoning in Geographic Space. Springer Verlag.

Papadias, D., Sellis, T. (1993) The Semantics of Relations in 2D Space Using Representative Points: Spatial Indexes. In Frank, A.U., Campari, I. (eds.) Proceedings of the European Conference on Spatial Information Theory COSIT. Springer-Verlag.

Papadias, D., Sellis, T. (1994) On the Qualitative Representation of Spatial Knowledge in Two-Dimensional Space. To appear in the Very Large Data Bases Journal, VLDB Special Issue on Spatial Databases.