Qualitative Spatial Reasoning using a N-Dimensional^{+} Projective Representation

Jorge Pais
Instituto Superior Engenharia Lisboa
Instituto de Sistemas e Robótica
Instituto Superior Técnico*
email: jpais@isel.pt

Carlos Pinto-Ferreira
Instituto de Sistemas e Robótica
Instituto Superior Técnico*
email: cpf@isr.ist.utl.pt

Abstract

A N-Dimensional^{+}(ND^{+}) projective representation to support qualitative spatial reasoning is presented. The principal features of this representation are, the translation from a N-dimensional Euclidean space into a ND^{+} world without lost of its spatial properties, the compression of data on qualitative representation of elements, the representation of change using a minimal set of operators, the completion and consistency of all world topological descriptions, and the automatic definition of all topological relationships among elements.
Consequently the resulting ND^{+} reasoning process is very effective. This offers four significant advantages over other multi-dimensional spatial reasoning approaches. Firstly, the topological relations among elements are intrinsic to the representation and therefore the exponential growth in the number of relations w.r.t. the number of elements and the dimensionality of space does not happen. Secondly, there is no generation of inconsistent topological descriptions and consequently the computational resources demanded to generate, test and purge inconsistent descriptions are no longer needed. Thirty, there is no place of incomplete topological definitions and then, also, in this aspect no computational resources are expended. Finally, the computational cost of manipulating a ND^{+} space increases linearly N times when compared with a 1D^{+} space.
For the present, as this representation is completely pictorial, which means it does not incorporate verbal definitions, then all world elements with any shape are qualitatively transformed into world^{+} regions that are bodies with boundaries parallel to the projective axis. Notwithstanding this drawback, the spatial properties of the real world will not be lost in the ND^{+} world as suggested by the results obtained by the implemented system.

1 Introduction

Two of the most challenging and hard problems in A.I. is how an artificial system performs automatically reasoning about an unknown domain or about a known NP complete domain where the complexity and the computational resources needed to solve problems grow up exponentially. Particularly in spatial reasoning, there is a lot of research work concerned with one [Allen, 1991],[Detcher *et al.*, 1991] and two dimensional spaces [Hernandez, 1991],[Freksa, 1991], [Retz-Schmidt, 1998] but with respect to spaces with three or more dimensions a few successful approaches have been developed (to our knowledge) [Coenen *et al.*, 1998]. Scaling up spatial approaches developed to solve one or two dimensional domains into three or more dimensions almost generates a combinatorial explosion in the number of relations needed to representing the domain relationships among elements and in the computational resources to solve problems.
The question how to model an artificial system and its environment is a question stressed in [Albus, 1984] and his opinion is completely shared by us because we are convinced that the way of representation knowledge about the world can make the reasoning process simpler or harder in solving problems. And the most important part of the world representation done by the human beings in every day life seems to be almost qualitative.
As the real world space is hard to model a simpler model of space named 2D^{+} world was proposed in [Backstrom, 1990]. It is based on a geometrical logic that is a first order predicate calculus composed by primitive and derivable predicates. The main objective of 2D^{+} world was to model simplified geometrical objects and mechanical assembly processes with real effectiveness, in terms of results and in the aim of pre-

* Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal.

serving the real word properties.

The projective representation here presented shares one characteristic with $2D^+$ world that is the shape of elements, which have parallel boundaries with the projective axis. However, excluding the referred aspect, the ND^+ projective representation is completely pictorial whereas the $2D^+$ approach is based on verbal characterization (predicates).

The ND^+ projective representation (ND^+PR) is based on fundamental geometric principles [Ayres, 1967] and has N projective axis. A region is a body with boundaries parallel to all projective axis and in each projective axis an existing region is defined by two projective region vertex, the start and the end vertex projections. Consequently the ND^+PR is a completely pictorial approach and its principal features can be summarized as follows:

- Data compression because each region is only identified in each projective axis by its start and end vertex projections.
- All topological descriptions are complete and consistent.
- All topological relationships among elements are automatically defined.
- Just two movement operators *left* and *right* generate change in each projective axis.
- The one-dimensional reasoning process is effective as consequence of the mentioned features.
- The extensibility from a $1D^+$ to a ND^+ reasoning process just requires an N times repetition of the $1D^+$ algorithm.

A more detailed discussion of the ND^+ projective representation will be presented in section 2. A definition of $1D^+$ operators will be done in section 3 and its extensibility to a multidimensional space. In section 4 will be presented the algorithm that supports the one-dimensional reasoning process of our system and its extensibility to a multidimensional space. Results in different dimension space will be showed in section 5. Finally, in section 6 we discuss the drawback of a complete pictorial approach and how in future work it can be solved.

2 The One-Dimensional$^+$ Projective Representation

The foundations of this representation are based on three geometrical concepts that are by hierarchical order the *projective axis*, the *projective axis vertex,* and the *projective region vertex*.

2.1 Projective Axis

A projective axis is a straight line that defines a $1D^+$ representation in the Euclidean Space.

The representation of an ND^+ space is done using N projective axis that can be defined as $S_n = \{A_1, \ldots, A_n\}$, where A_i represents the ith projective axis. Particularly, the $3D^+$ space can be represented as $S_3 = \{A_1, A_2, A_3\}$ that defines the space showed in figure 1.

2.2 Projective Region Vertex

A projective region vertex is a form to representing start and end points of a region projection over a certain projective axis. A region is a body with boundaries parallel with all projective axis and it can be considered as a result of a geometrical transformation applied to a physical element with any shape. For instance, figure 1 showed a geometrical transformation from a physical element to a region into a $3D^+$ space.

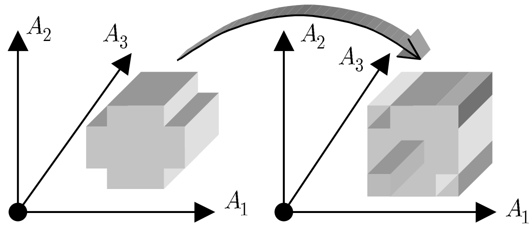

Figure 1: The 3D+ geometrical transformation.

The existing regions into a domain can be represented by a set $R_k = \{r_1, r_2, \ldots, r_k\}$, such that r_j represents jth region into a domain.

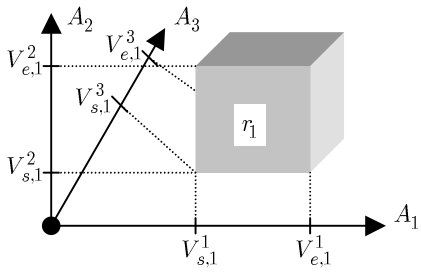

Figure 2: Projective region vertex.

A projective region is represented in each projective axis by two vertex, $V^a_{s,j}$ - the character 's' identifies the projective *start vertex* of region j over an axis A_a, and $V^a_{e,j}$ - the character 'e' defines the projective *end vertex* of region j over an axis A_a (see figure 2)

2.3 Projective Axis Vertex

A projective axis vertex is an entity V^a_m that represents the mth vertex axis into a projective axis A_a, and it exists whether it always holds at least one projective region vertex. This means that a projective axis vertex is a non-empty set of projective region vertex.

An example is showed in figure 3, where is defined a three dimensional space with two regions, which produce four projective region vertex in each projective axis but the number of projective axis vertex can vary between two and four depending from the spatial position of regions.

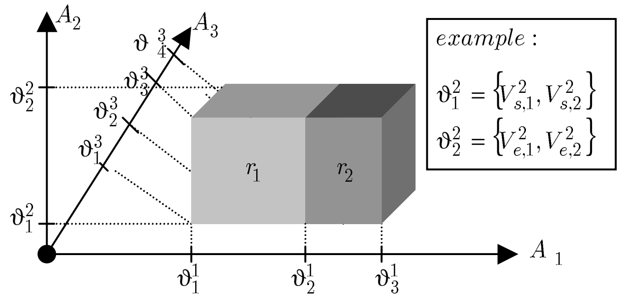

Figure 3: Projective axis vertex.

2.4 Shape of Regions

In $1D^+$ space a region is an interval between the start and the end projective region vertex. In $2D^+$ space a region is a rectangle or a square defined by four projective region vertex, two in each projective axis. In $3D^+$ spaces a region is a cube and, in spaces greater than $3D^+$ regions are hypercubes.

3 Projective Operators

3.1 $1D^+$ Positional Operators

All desired information about the world should be derivable from the *primitive positional operators*, which themselves should not be derivable from each other. The primitive positional operators define a minimal model of the world and are just defined by the set $\{<<, \equiv\}$.

- $V_{s,k}^a << V_{e,k}^a$: means that the projective region start vertex $V_{s,k}^a$ is closer than the end vertex $V_{e,k}^a$ from the projective axis origin of a.
- $V_{s,k}^a \equiv V_{s,h}^a$: signifies that these two projective region vertex are equidistant from the projective axis origin.

The *derivable positional operators* are useful to defining topological relationships among projective region vertex. These operators are asserted using primitive positional operators, as follows:

- $Left(V_{i,r}^a) = \varphi_{i,r}^a : \forall V_{x,m}^a \in \varphi_{i,r}^a (V_{x,m}^a << V_{i,r}^a)$
- $Right(V_{i,r}^a) = \delta_{i,r}^a : \forall V_{x,m}^a \in \delta_{i,r}^a (V_{i,r}^a << V_{x,m}^a)$
- $Coincident(V_{i,r}^a) = \gamma_{i,r}^a : \forall V_{x,m}^a \in \gamma_{i,r}^a (V_{x,m}^a \equiv V_{i,r}^a)$

3.2 $1D^+$ Relational Operators

Relational operators are used to defining a set of primitive topological relationships among regions into a projective axis A_1, which are defined as follows:

- $OutsideLeft(A_1, r_i) = \alpha_i^1 \Rightarrow \forall r_x \in \alpha_i^1 : V_{e,x}^1 \in Left(V_{s,i}^1)$
- $OutsideRight(A_1, r_i) = \chi_i^1 \Rightarrow \forall r_x \in \chi_i^1 : V_{s,x}^1 \in Right(V_{e,i}^1)$
- $OutsideLeftCoincident(A_1, r_i) = \phi_i^1 \Rightarrow \forall r_x \in \phi_i^1 : V_{e,x}^1 \in Coincident(V_{s,i}^1)$
- $OutsideRightCoincident(A_1, r_i) = \delta_i^1 \Rightarrow \forall r_x \in \delta_i^1 : V_{s,x}^1 \in Coincident(V_{e,i}^1)$
- $CompletelyCoincident(A_1, r_i) = \varepsilon_i^1 \Rightarrow \forall r_x \in \varepsilon_i^1 : V_{s,x}^1 \in Coincident(V_{s,i}^1) \wedge V_{e,x}^1 \in Coincident(V_{e,i}^1)$
- $CompletelyInside(A_1, r_i) = \varphi_i^1 \Rightarrow \forall r_x \in \varphi_i^1 : V_{s,x}^1 \in Right(V_{s,i}^1) \wedge V_{e,x}^1 \in Left(V_{e,i}^1)$
- $InsideLeftCoincident(A_1, r_i) = \gamma_i^1 \Rightarrow \forall r_x \in \gamma_i^1 : V_{s,x}^1 \in Coincident(V_{s,i}^1) \wedge V_{e,x}^1 \in Left(V_{e,i}^1)$
- $InsideRightCoincident(A_1, r_i) = \eta_i^1 \Rightarrow \forall r_x \in \eta_i^1 : V_{s,x}^1 \in Right(V_{s,i}^1) \wedge V_{e,x}^1 \in Coincident(V_{e,i}^1)$
- $OverlappedRight(A_1, r_i) = \lambda_i^1 \Rightarrow \forall r_x \in \lambda_i^1 : V_{s,x}^1 \in (Right(V_{s,i}^1) \cap Left(V_{e,i}^1)) \wedge V_{e,x}^1 \in Right(V_{e,i}^1)$
- $OverlappedLeft(A_1, r_i) = \omega_i^1 \Rightarrow \forall r_x \in \omega_i^1 : V_{s,x}^1 \in Left(V_{s,i}^1) \wedge V_{e,x}^1 \in (Right(V_{s,i}^1) \cap Left(V_{e,i}^1))$

3.3 ND^+ Operators

In a ND^+ world, scaling up positional operators from $1D^+$ to ND^+ just requires to substituting the symbol A_1 that represents the unique projective axis by the current projective axis A_i.

Analyzing figure 3 that represents a region r_1 into a $3D^+$ space and considering the $1D^+$ positional operator *Left* applied over all projective axis, then the resulting positional operators are $Left(V_{s,1}^1)$, $Left(V_{s,1}^2)$ and $Left(V_{s,1}^3)$ that can be usually intended in common sense as *left*, *backward* and *below* operators. Thus, a correct interpretation of our relational operands depends on the distribution of the projective axis over the Euclidean Space.

The previous example illustrates how to expanding all our operands from a $1D^+$ world to a ND^+ world. In fact, it just consists of substituting the parameter A_1 in formulae presented in sections 3.1 and 3.2 by the parameter A_i, which is the i^{th} projective axis among N.

4 ND^+ Projective Reasoning Process

4.1 Movement Operators

Just two atomic movement operators are able to generate change over each projective region vertex along each projective axis.

- $MoveVertexLeft(V^i_{k,r}, \vartheta^i_j)$: moves a projective region vertex $V^i_{k,r}$ from the current projective axis vertex ϑ^i_j to its left projective axis vertex.
- $MoveVertexRight(V^i_{k,r}, \vartheta^i_j)$: changes a projective region vertex $V^i_{k,r}$ from the current projective axis vertex ϑ^i_j to another projective axis vertex on its right.

MoveVertexLeft($V^i_{s,r}$, ϑ^i_j)
  RemoveRegionVertex $V^i_{s,r}$ from ϑ^i_j
  IF $(\exists V^i_{s,b}, V^i_{e,b} : V^i_{s,b} \in \vartheta^i_{j-1} \wedge V^i_{e,b} \in \vartheta^i_j)$
    CreateProjectiveVertex $\vartheta^i_k : \vartheta^i_{j-1} << \vartheta^i_k << \vartheta^i_j$
    InsertRegionVertex $V^i_{s,r}$ into V^i_k
  ELSE
    InsertRegionVertex $V^i_{s,r}$ into ϑ^i_{j-1}
  IF $(\vartheta^i_j = \Phi)$ RemoveAxisVertex(ϑ^i_j)

Figure 4: Move *left* a *start region vertex.*

If none of these operators are applied another movement operator that generates no change is defined as $NoMoveVertex(V^i_{k,r}, \vartheta^i_j)$.

MoveVertexLeft($V^i_{e,r}$, ϑ^i_j)
  RemoveRegionVertex $V^i_{e,r}$ from ϑ^i_j
  IF $(V^i_{s,r} \in \vartheta^i_{j-1}) \vee (\exists V^i_{s,b}, V^i_{e,b} : V^i_{s,b} \in \vartheta^i_{j-1} \wedge V^i_{e,b} \in \vartheta^i_j)$
    CreateProjectiveVertex $\vartheta^i_k : \vartheta^i_{j-1} << \vartheta^i_k << \vartheta^i_j$
    InsertRegionVertex $V^i_{e,r}$ into ϑ^i_k
  ELSE
    InsertRegionVertex $V^i_{e,r}$ into ϑ^i_{j-1}
  IF $(\vartheta^i_j = \Phi)$ RemoveAxisVertex(ϑ^i_j)

Figure 5: Move *left* an *end region vertex.*

MoveVertexRight($V^i_{s,r}$, ϑ^i_j)
  RemoveRegionVertex $V^i_{s,r}$ from ϑ^i_j
  IF $(V^i_{e,r} \in \vartheta^i_{j+1}) \vee (\exists V^i_{s,b}, V^i_{e,b} : V^i_{s,b} \in \vartheta^i_j \wedge V^i_{e,b} \in \vartheta^i_{j+1})$
    CreateProjectiveVertex $\vartheta^i_k : \vartheta^i_j << \vartheta^i_k << \vartheta^i_{j+1}$
    InsertRegionVertex $V^i_{s,r}$ into ϑ^i_k
  ELSE
    InsertRegionVertex $V^i_{s,r}$ into ϑ^i_{j+1}
  IF $(\vartheta^i_j = \Phi)$ RemoveAxisVertex(ϑ^i_j)

Figure 6:Move *Right* a *start region vertex.*

The movement operators must guarantee continuity between vertex regions and regions. Thus, when a vertex region axis $V^i_{k,r}$ moves along any domains region movement operators must guarantee change across spatial relationships from *Left* to *Right* or the opposite, depending of the movement direction. Considering such rules, the algorithms to describing these operators are presented in figures 4, 5, 6 and 7. Note that in these figures the projective region vertex $V^i_{s,r}$ and $V^i_{e,r}$ are considered into the projective vertex ϑ^i_j.

MoveVertexRight($V^i_{e,r}$, ϑ^i_j)
  RemoveRegionVertex $V^i_{e,r}$ from ϑ^i_j
  IF $(\exists V^i_{s,b}, V^i_{e,b} : V^i_{s,b} \in \vartheta^i_j \wedge V^i_{e,b} \in \vartheta^i_{j+1})$
    CreateProjectiveVertex $\vartheta^i_k : \vartheta^i_j << \vartheta^i_k << \vartheta^i_{j+1}$
    InsertRegionVertex $V^i_{e,r}$ into ϑ^i_k
  ELSE
    InsertRegionVertex $V^i_{e,r}$ into ϑ^i_{j+1}
  IF $(\vartheta^i_j = \Phi)$ RemoveAxisVertex(ϑ^i_j)

Figure 7: Move *right* an *end region vertex.*

4.2 ND$^+$ Movement Algorithms

Particularly, 1D$^+$ algorithm takes advantage of the relational and movement operators defined in previous sections. Considering C_1 and F_1 as being the current and final projections over the unique projective axis the 1D$^+$ algorithm is as showed in figure 8.

OneDimensionalSpatialMovement(C_1 , F_1)
  WHILE $(C_1 \neq F_1)$
    FOR each one of all axis vertex $\vartheta^1_j \in C_1$
      FOR each one of all region vertex $V^1_{i,r} \in \vartheta^1_j$
        IF $(V^1_{i,r} \in F_1)$
          IF $(Left(V^1_{i,r}) \in C_1) \subset (Left(V^1_{i,r}) \in F_1)$
            $MoveVertexRight(V^1_{i,r}, \vartheta^1_j)$
          ELSE
            IF $(Right(V^1_{i,r}) \in C_1) \subset (Right(V^1_{i,r}) \in F_1)$
              $MoveVertexLeft(V^1_{i,r}, \vartheta^1_j)$

Figure 8: The 1D$^+$ Movement Algorithm.

The 1D$^+$ algorithm complexity is quadratic in order to the number of projective region vertex into the projective axis. As the number of projective region vertex are only two per region and whether considering K regions into domain, then the complexity in worst case can be represented as $O(2^2 \times K^2)$.

Generally, ND$^+$ algorithm just requires executing 1D$^+$ algorithm for each one of the N projective axis, as illustrated in figure 9, consequently the resulting complexity of the ND$^+$ algorithm is $O(N \times 2^2 \times K^2)$. For instance, to find out a solution in a 3D$^+$ space is three times more expensive than in one-dimensional space, considering an equal number of regions in both spaces.

NDimensionalSpatialMovement()
WHILE $(C_N \neq F_N)$
  FOR each i of N projective axis
    IF $(C_i \neq F_i)$
      OneDimensionalSpatialMovement(C_i, F_i)

Figure 9: The ND+ Movement Algorithm.

Then, the complexity of this approach increases linearly in order to the dimensionality of the domain.

5 Results

The ND+PR behavior can best be illustrated by considering an example in a 3D+ world with 2 regions named *X* and *Y*, as showed in figure 10.

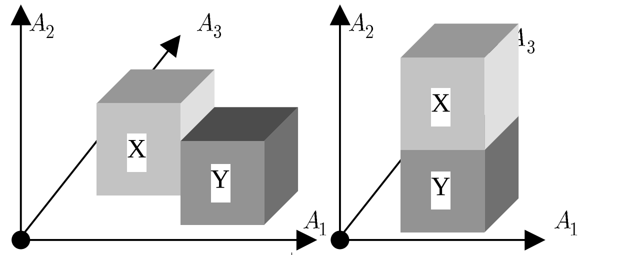

Figure 10: The 3D+ initial and final states.

The solution paths generated by the ND+PR approach to the problem described in figure 10 is showed in figure 11 respectively. Such solution path is an optimal solution path and it is the first one generated by the ND+PR approach.

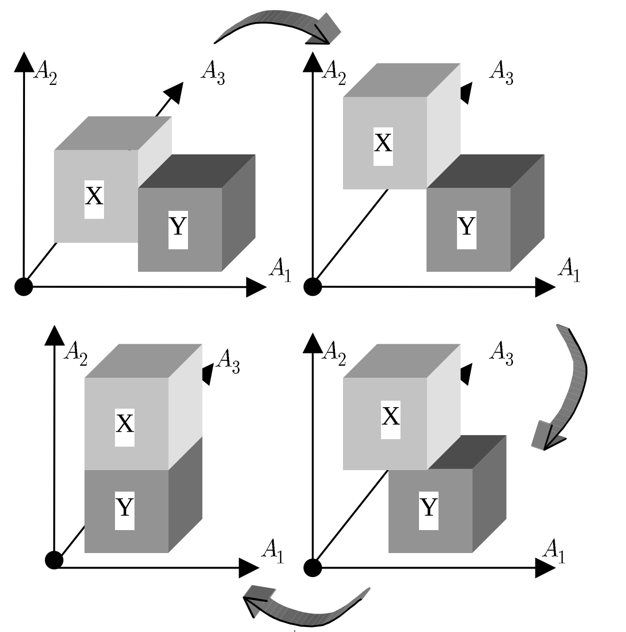

Figure 11: The 3D+ optimal solution path.

6 Conclusions and Future Work

As can be seen ND+PR is a completely pictorial approach in modeling N-dimensional domains requiring few computational resources to solve problems.

However this representation and its reasoning process is not able to dealing with spatial constraints that occur in real-world problems by the intrinsic physical properties of elements (e.g., impenetrability, rigidity, etc.). Also, the notion of the original shape of bodies is lost when the ND+ geometrical transformations are performed. For all that, the effectiveness of the implemented system encourages us to solve its drawbacks as quickly as possible.

Apparently the human brain works interleaving both pictorial and verbal knowledge to solving problems. The challenge is, therefore, to make possible an architecture, which incorporates verbal knowledge on the presented pictorial representation without degrading the system effectiveness.

References

[Allen, 1991] Allen J.F., "Time and Time Again: The Many Ways to Represent Time," in: *International Journal of Intelligent Systems* 6, 341-355.

[Detcher *et al.*, 1991] Detcher R., Meiri I., Pearl J., "Temporal Constraint Networks," in: *Artificial Intelligence* 49, 61-95..

[Hernandez, 1991] Hernandez D., "Relative Representation of Spatial Knowledge: The 2D Case," in*: Cognitive and Linguistic Aspects of Geographic Space*, 373-385, Kluwer Academic Publishers, Netherlands.

[Freksa, 1991] Freksa C., "Qualitative Spatial Reasoning," in*: Cognitive and Linguistic Aspects of Geographic Space*, 361-372, Kluwer Academic Publishers, Netherlands.

[Retz-Schmidt, 1988] Retz-Schmidt G., "Various Views on Spatial Proposition," in *A.I. Magazine*, 9(2), 95-105.

[Coenen *et al.*, 1998] Coenen F., Beattie B., Shave M., Bench-Capon T., Diaz B., "Spatial Reasoning Using the Quad Tesseral Representation," in: *Artificial Intelligence Review* 12, 321-243, Kluwer Academic Publishers, Netherlands.

[Albus, 1984] Albus J.S., Robotics, in: Robotics and Artificial Intelligence (NATO ASI Series, Vol. F11), M. Brady, L.A. Gerhardt, H.F. Davidson (Eds), Berlin: Springer Verlag.

[Backstrom, 1990] Backstrom C., Logical Modeling of Simplified Geometrical Objects and Mechanical Assembly Processes, in: Advances in Spatial Reasoning Vol.1, edited by Su-Shing Chen, ABLEX publishing corporation, USA.

[Ayres, 1967] Ayres F., *Theory and Problems of Projective Geometry*, in: Schaum's outline series in Mathematics, McGraw-Hill.

[Cui *et al.*, 1992] Cui Z., Cohn A., and Randell D., "Qualitative Simulation Based on a Logical Formalism of Space and Time", in: Proceedings of AAAI92, AAAI Press.

[Pais *et al.*,1998] Pais J., Pinto-Ferreira C., "Search Strategies for Reasoning about Spatial Ontologies," in: 10th IEEE International Conference On Tools with Artificial Intelligence, Taiwanese Association for Artificial Intelligence (ICTAI98), Taiwan.

[Beattie *et al.*,1996] Beattie B., Coenen F.P., Hough A., Bench-Capon T.J.M., Diaz B.M., Shave M.J.R., "Spatial Reasoning for Environmental Impact Assessment," in: Third International Conference on GIS and Environmental Modeling, National Centre for Geographic Information and Analysis, WWW and CD, Santa Barbara.