Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

Research Report
RR-92-21

Representing Spatial Relations (Part II)

-The Geometrical Approach-

Jörg-Peter Mohren
Jürgen Müller

April 1992

Deutsches Forschungszentrum für Künstliche Intelligenz GmbH

Postfach 20 80
D-6750 Kaiserslautern, FRG
Tel.: (+49 631) 205-3211/13
Fax: (+49 631) 205-3210

Stuhlsatzenhausweg 3
D-6600 Saarbrücken 11, FRG
Tel.: (+49 681) 302-5252
Fax: (+49 681) 302-5341

Deutsches Forschungszentrum für Künstliche Intelligenz

The German Research Center for Artificial Intelligence (Deutsches Forschungszentrum für Künstliche Intelligenz, DFKI) with sites in Kaiserslautern and Saarbrücken is a non-profit organization which was founded in 1988. The shareholder companies are Atlas Elektronik, Daimler Benz, Fraunhofer Gesellschaft, GMD, IBM, Insiders, Mannesmann-Kienzle, Philips, SEMA Group Systems, Siemens and Siemens-Nixdorf. Research projects conducted at the DFKI are funded by the German Ministry for Research and Technology, by the shareholder companies, or by other industrial contracts.

The DFKI conducts application-oriented basic research in the field of artificial intelligence and other related subfields of computer science. The overall goal is to construct *systems with technical knowledge and common sense* which - by using AI methods - implement a problem solution for a selected application area. Currently, there are the following research areas at the DFKI:

- ❏ Intelligent Engineering Systems
- ❏ Intelligent User Interfaces
- ❏ Intelligent Communication Networks
- ❏ Intelligent Cooperative Systems.

The DFKI strives at making its research results available to the scientific community. There exist many contacts to domestic and foreign research institutions, both in academy and industry. The DFKI hosts technology transfer workshops for shareholders and other interested groups in order to inform about the current state of research.

From its beginning, the DFKI has provided an attractive working environment for AI researchers from Germany and from all over the world. The goal is to have a staff of about 100 researchers at the end of the building-up phase.

Prof. Dr. Gerhard Barth
Director

Representating Spatial Relations (Part II) -The Geometrical Approach-

Jörg-Peter Mohren, Jürgen Müller

DFKI-RR-92-21

Parts of this report will be published at

the European Conference on AI (ECAI-92), Vienna, August 1992, under the title: *A Geometrical Approach to the Depictional Representation of Spatial Relations*

This work has been supported by a grant from The Federal Ministry for Research and Technology (FKZ ITW-9104).

Representing Spatial Relations (Part II)

-The Geometrical Approach-

Jörg-Peter Mohren
Jürgen Müller

DFKI - The German Research Center for AI
Stuhlsatzenhausweg 3
D-6600 Saarbrücken
Germany
Tel.: ++49 681 302 5322
e-mail : mueller@dfki.uni-sb.de

Abstract

The representation and analysis of spatial relations is a tough problem in AI and Cognitive Science and is hence heavily discussed in the literature. Our general approach to this problem is to use a two-level representation where the relations may either be defined on a logical/propositional level or in terms of a three dimensional model of co-ordinates. Here we occupy ourselves with an approach to analyze spatial relations on the depictional level, i.e. on a representation of spatial scenes by space co-ordinates.

First we describe a representation formalism for spatial objects, based on boundary representations. Coming from that, we introduce a method for testing the applicability of spatial relations between two or more objects. The degree of applicability of a spatial relation results from the deviation of the object to be located from an 'ideal position' which is specified by the reference object(s) and various influences by spatial properties of the regarded objects like size or shape, where the deviation results from trigonometrical computations.

Key words : Spatial Relation, Knowledge Representation

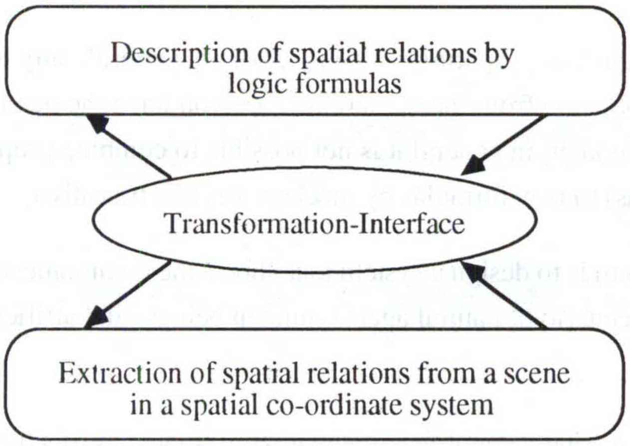
Area : Knowledge Representation

Index

1. Introduction

The ability of processing spatial knowledge is a fundamental characteristic of intelligence as we see it. This aspect of intelligence cannot only be found amongst human beings, but also amongst other, often much lower developed creatures. Also the ability of describing the positions of objects in space relatively to others is not reserved to man. E.g., bees can describe the location of food to members of their beehive by dance-like movements ([DS82]). This ability allows beings to communicate about real world events, even if one of the communicating persons only partially knows the scene.

Primarily, man uses language to describe spatial affairs, supporting his statements by gestures. On the other hand, he receives a picture of his surroundings by his eyesight. We see that he uses language-like or **propositional** representations of spatial relations on the one hand, and image-like or **depictional** representations on the other (see also section 2). So we come to the question, how one can be transformed to the other. What we need, is a **transformation interface** between the two representation formalisms, as the following graphic explains :



We so obtain a **hybrid system** for the representation of spatial knowledge (Fig.1). The idea of a hybrid representation formalism comes from cognitive science ([Ko80]). The so-called **Imagery Debate** at last came to the result that both representation formalisms are used at the same time, dependant from the work that is to be done. E.g., you know that your house is left of your neighbor's house, because you see the houses everyday. This fact is stored up propositionally. On the other hand, scenes you have only seen once are stored depictionally, already because this 'mind picture' contains much more information than the single relation *left_of(my_house, neighbour_house)*. When needed, the relation can be deducted from the depiction.

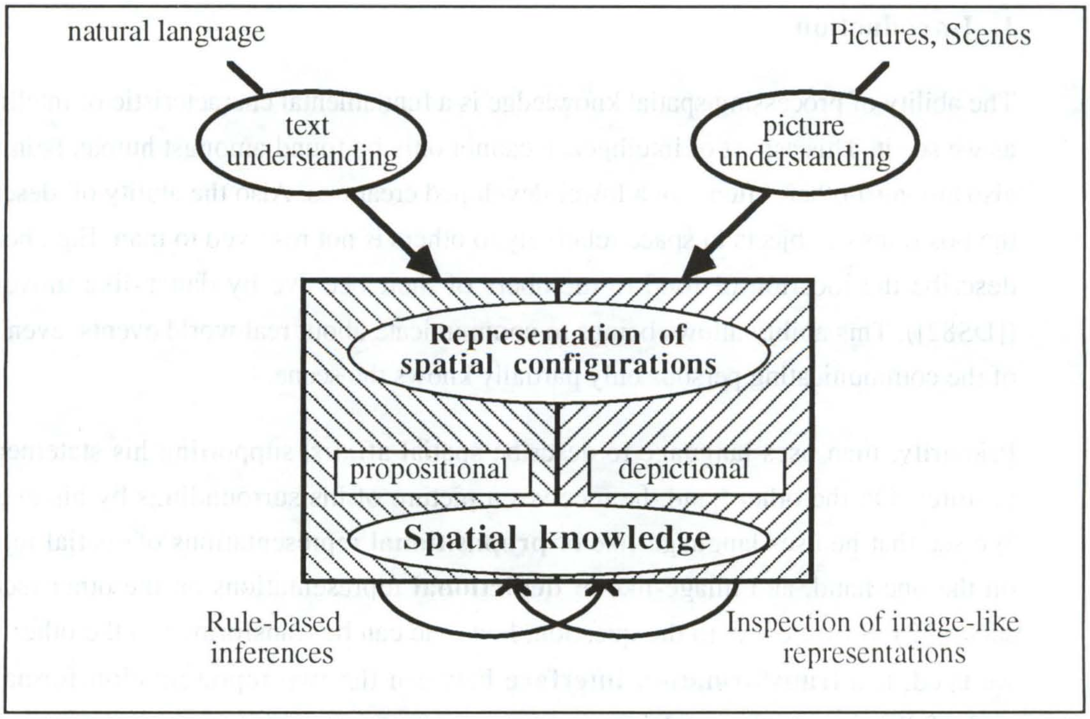


Fig.1 : Hybrid systems for space representation

You can deduct any spatial relation (and, in general, any spatial property of the represented objects) from the depiction when you have the needed deduction algorithm. As we will see later, in general it is not possible to combine propositional formulas (e.g., logic formulas) to new formulas by mechanisms like transitivity.

The general aim is to design a system that allows the communication on spatial affairs in a multi-agent scenario of natural agents (human beings) and artificial agents (robots, expert systems etc.).

Example : Imagine a depot where a warehouseman controls the work of several robots. An XPS administrates the goods. Especially it knows where all the things are stored. So it could describe the position of some object to the warehouseman. On the other hand, the warehouseman could give orders like : 'Bring me the box behind the pillar' to one of the robots.

This little example already shows that both, the XPS and the robots, should have the ability of understanding human space descriptions and generating own descriptions in a way that is understandable to man. Therefore, our aim will be to develop a system that is capable to extract spatial relations from a given scene description. The scene description may be given to the robots e.g. by laser scanning of the surroundings, or by incremental inserting and removing of objects in a given initial scene (e.g., the description of the empty depot). The underlying idea of the developed method is independent of the data structure by which the scene is represented, though, of course, certain representation formalisms support the algorithms in a better way.

2. Representation of Spatial Objects

In this section, we will present a formalism for representing spatial objects in a three-dimensional co-ordinate system. We presume that the spatial objects are completely known, i.e., we know the exact co-ordinates of all their points or can deduct them easily. The formalism bases on the idea of boundary representation of spatial objects (comp. [Mo91], chapter 2). The basic idea is to build up n-dimensional objects from (n-1)-dimensional, i.e., lines from points, planes from lines and 3D-objects from planes. We will restrict ourselves to a BNF-like introduction of the formalism, including short explanations where needed :

Points

<point> ::= (REAL REAL REAL)

→ The three components represent the co-ordinates of the point in a given three-dimensional co-ordinate system.

Lines and arcs

<line> ::= (<starting_point> <end_point>)

→ A line is represented by his starting and end point.

<circular_arc> ::= (<start_point> <end_point> <center> <arc_point>)

→ A circular arc is represented by his starting and end point, the center of the circle it is lying on, and the arc-point which represents a point on the arc that is neither the starting not the end point. This point serves for the orientation of the arc, as Fig.2 makes clear.

<elliptical_arc> ::= (<start_point> <end_point> <center1> <center2> <arc_point>)

→ The representation of elliptical arcs is the same as for circular arcs, except the fact that there are two center points needed instead of one.

Planes

<closed_trail> ::= ({ <line> | <circular_arc> | <elliptical_arc> }$^+$)

→ A closed trail is an ordered list of arcs and lines where the starting point of the first line (arc) is the end point of the last one.

<even_plane> ::= <closed_trail>

→ An even plane is a coherent set of points that are all lying in one plane, that means, they can all be described by a function $ax+by+cz+d = 0$ ($a,b,c \in \mathbb{R}$). This function can easily be deducted from three points of the plane that are not lying on the same line. The lines and arcs describing the closed trail delimit the plane.

<spherical_plane> ::= (<closed_trail> <sphere_center> <surface_point>)

→ A spherical plane is the surface or a coherent part of the surface of a sphere. The closed trail delimits the plane if it is only a part of the surface, otherwise it is only a point.

<cylindrical_plane> ::= ({<closed_trail>}^{1,2} <center_line>)

→ A cylindrical plane is the mantle surface or a coherent part of the mantle surface of a spheric cylinder represented by the center line. The plane is delimited by one or two closed trails.

3D-objects

<3D_object> ::= ({<even_plane> | <spherical_plane> | <cylindrical_plane>}+)

→ A three-dimensional object is built up from a list of planes. It must be assured that these planes build a completely closed object because the developed algorithms presume this.

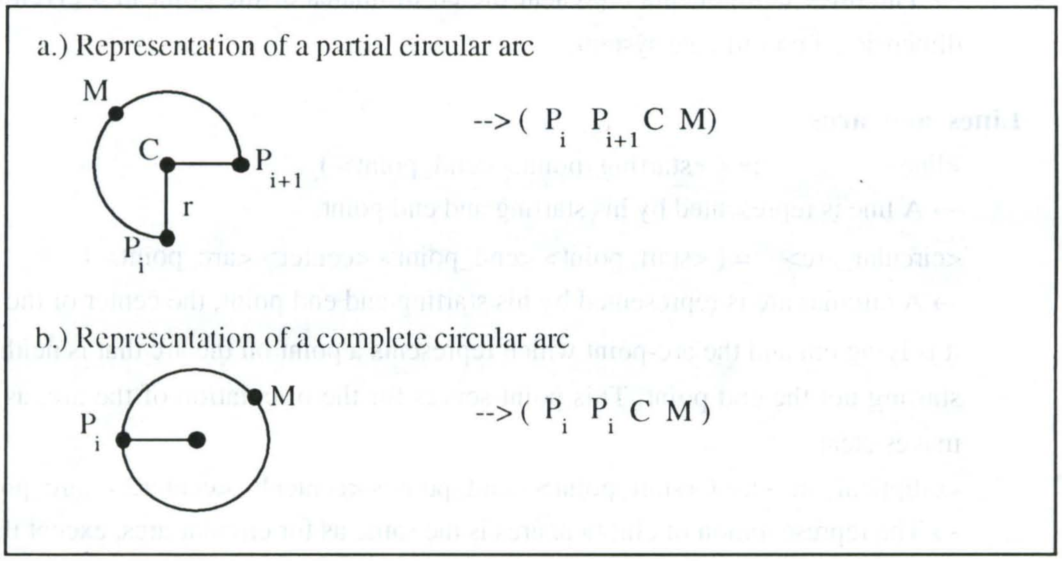


Fig.2 : Representation of arcs

It is obvious that this formalism will not do to represent all possible objects (cf. [Mo91], sect.6). But on the other hand, one must consider that it is often adequate to reduce the real world objects to simplified representations. Therefore, this method shall be sufficient for our purpose.

3. Analysis of Spatial Relations

3.1 Introduction to Spatial Relations

A spatial relation R describes to position of a spatial object LE (located entity) relatively to other objects RO_i (reference objects). The number of reference objects depends on the relation R and can (in principle) be any number in $\mathbb{N}_0$.

Example : here(box_1) → no reference object
in_front_of(box1, box2) → one reference object
between(box_1, box_2, box_3) → more than one reference object

Our aim is to develop a system that is capable of comparing spatial objects with respect to their degree of acceptability of a spatial relation, as the following example makes clear :

Example : Imagine the depot outlined in Fig.3. The warehouseman asks the robot to get the box in front of the pillar. The robot sees that there are two boxes standing in front of the pillar, but he remarks that the right box is 'more in front of the pillar' than the left one, so he chooses the right one.

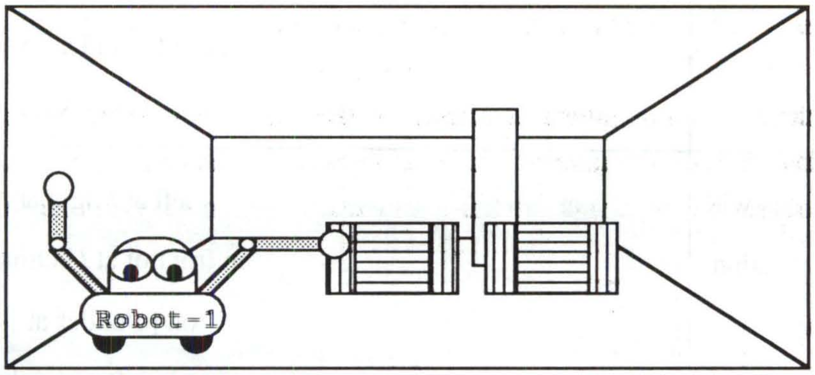


Fig.3 : Different degrees of applicability of the relation 'in_front_of(pillar, x)

Therefore, we interpret spatial relations as "fuzzy" relations, i.e., as functions with values in the real interval [0,1]. We shall call the value of an instantiated spatial relation R (LE, RO_1,..., RO_n) the **degree of applicability of R to LE and RO_1,...,RO_n.**

We separate the spatial relations in several classes (cf. Fig. 4). First, we differ between **topological** and **directional relations**. We call a spatial relation topological if it only refers to topological aspects of LE and RO_1,...,RO_n. Especially, they are independent of the viewer's position. Basic examples are the relations *in, near* and *between*. The opposite of the topological relations are the directional relations which describe the direction in which the located object is positioned in respect to the reference object. This direction is a non-topological property and can only be specified relatively to a given reference system. Examples for directional relations in three-dimensional space are *left/right of, in front of /behind* and *above/below*.

Further we subdivide spatial relations in **basic relations, composed relations** and **relations with refined specification**. A basic relation is elementary within the meaning that their analysis cannot be reduced to other relations (cf. Fig.4). Composed relations result from the combination of two or more basic relations by the boolean operators *and* and *or*. Examples are combinations like *left behind* (combined by *and*) or *besides* (combining *left of* and *right of* by *or*).

	Topological relations	Directional relations
basic relations	inside / outside of between near	in front of / behind left of / right of above / below
composed relations		left/right in front of/behind left/right above/below [in front of/behind above/below] besides
relations with refined specification	at	left at / right at in front of at / behind at on [/ under at] directly left of / right of directly in front of / behind directly above / below <spatial adverbial specification> <spatial relation>

Fig.4 : Classification of spatial relations

Spatial relations can be refined by adverbial definitions which specify certain spatial properties like the distance of located and reference object. We obtain spatial relations like *on* (*above* where the distance should be zero) or *50 yards left of*.

Up to here, we only introduced spatial relations which describe the spatial relation between fix positions of objects. But beyond this, spatial relations can also describe the movement of an object in space by relations like *along*, *into* or *out of*. We so get a further distinction in **static** and **dynamic relations**. In this paper, the analysis of dynamic relations shall not be handled. For those who are interested in this aspect we refer to [Mo91] and [ABHR86a].

3.2 External Influences to Spatial Relation Analysis

The degree of applicability of a spatial relation depends on numerous factors. The first factor is the **view** which is the decisive influence for the directional relations. The view determines the **reference frame** of the reference object RO, i.e., which side of RO is assumed to be the front (back, left, right, upper, lower) side.

A directional relation R_d can be used in several modes (cf. also [Re88], [Mo91] and [ABHR86b]) :

- **Intrinsic use**
  In intrinsic use the reference frame is determined by intrinsic properties of RO. E.g., the front side of a house could be the side with the main entrance.
- **Extrinsic use**
  In extrinsic use, influences of the situational context are used to determine the reference frame. Such an influence is for example the actual direction of movement of the object. We also use earth gravitation for determining the upper and lower side of an object.
- **Deictic use**
  In deictic use, the reference frame is determined by the viewer's position. The front side of the object is the side he directly looks at when staring at the object. In a dialogue, the viewer can either be the speaker, the listener or any other person.
  Example : Seen from my position, the box is left of the pillar
  Looking from the door, the box is right of the pillar.

We don't want to deepen the problematic nature of different views in here, and therefore refer to the literature as enumerated above.

Further influences to the degree of applicability are intrinsic properties of the located object LE and the reference object RO ([HP88]). Aspects like size or contextual importance in general increase the degree of applicability. On the other hand, the degree can be decreased by competing reference objects, i.e., objects of great size or importance in the neighborhood of the reference object diminuate its influence range. Just so disturbing influences like e.g. walls between reference and located object can decrease the degree.

This short introduction to influences shall do in here. A good treatise on these aspects can be found in [HP88].

3.3 The Method of Referred Fronts for Spatial Relation Analysis

In this chapter we develop a new method for spatial relation analysis. The method bases on following ideas :

1.) The degree of applicability of a spatial relation R depends only on the position of the **referred front** of the located object LE which is determined by the reference object RO and the relation R to RO.

2.) The degree of applicability results primarily from the deviation of LE from an 'ideal position'.

3.) Little movements of the objects LE and RO should cause only little changes in the degree of applicability of the relation R (LE, RO).

We first should define the term 'referred front'. All the directional relations define a direction vector in $\mathbb{R}^3$, and that is the vector that points to the front of LE that is related by the relation. E.g., for the relation *in front of* the direction vector points to the front side of LE.

Def.: If LE is an object in three-dimensional space, and d a direction vector in $\mathbb{R}^3$, then the **referred front of LE identified by d** is defined as :

$$RF(LE,d) = \{ x \in \partial LE \mid \neg\exists\, y \in LE \text{ and } \neg\exists\, \delta > 0 : x = y + \delta d \}$$

The set of points RF(LE, d) can be interpreted as a set of planes (planes in the sense of the representation formalism of §2).

The **referring front** of a reference object RO identified by d is defined as

$$RF^{-1}(RO, d) = RF(RO, -d)$$

E.g., the referred front of LE relative to the relation *in front of* is the front side of LE, the referring front of RO is the back side of RO.

In this definition, ∂LE is the rim of LE, i.e., the set of points on a border plane of LE (for exact definitions, cf. to [Mo91] or standard literature on mathematical analysis. We don't want to deepen the difficulties of computing the referred front of an object LE, and instead refer to [Mo91], pages 49-52 and 81-84.

We can thus reduce the problem whether a spatial object LE stands in a certain relation *R* to another object RO to the question whether the referred front of LE stands in this relation to RO or, better said, to the referring front of RO. This refers only to the directional relations and the topological relations *between* and *near*. The relations *in* and *outside of* are handled separatedly, because their applicability depends on others than these factors. The degree in which a directional relation between the object LE and the referring front $RF^{-1}(RO, d)$ is fulfilled in general bases on two influences, that are

- difference between the **distance** of the objects and a 'nominal distance', and
- **lateral deviation** of the located object from an 'ideal deviation'

We won't refer to the computation of the distance of two objects (on interest, see [Mo91], p. 71-76). Instead, our aim will be to develop a method for computing the lateral deviation of a point from an 'ideal position' and combine these 'point deviations' to an 'object deviation'. We only regard the case of a 'default view', that means, that the direction of view runs in direction of the z-axis, parallel to the x-z-plane. This state can simply be achieved by rotating the scene. The total lateral deviation then results from two deviation values representing the lateral deviations in two planes that are orthogonal to the vector d defined by the relation. For illustration, using the relation *behind*, the total deviation results from the deviations to the left/right and to the top/bottom.

We use in the following the function

$$\text{extreme\_point}(c_1 : E_1, c2 : E_2, \text{Graph\_Obj}) \rightarrow \text{Point}$$

that computes the point of Graph_Obj with c_1-co-ordinate extreme 'in context E_1' and c_2-co-ordinate extreme in sense E_2, e.g., extreme_point(x:max, y:min, LE) computes the point of LE with maximal x-co-ordinate and minimal y-co-ordinate. We shall illustrate the computation of the average lateral deviation by the deviation to the left/right of an object LE using the relation *behind*. We shall use the following denotations :

RO_x := extreme_point (x:min, y:max, RO)
RO^x := extreme_point (x:max, y:max, RO)
LE_x := extreme_point (x:min, y:min, LE)
LE^x := extreme_point (x:max, y:min, LE)

We first concentrate to the computation of the lateral deviation of a single point from the referring front of RO. Using the definitions introduced above, we define a function *rad* that computes the lateral deviation to the left/right of a single point by :

$$
rad(P, RO) = \begin{cases} \arctan \left| \dfrac{\text{x-val}(P) - \text{x-val}(RO_x)}{\text{y-val}(P) - \text{y-val}(RO_x)} \right| & \text{if x-val}(P) < \text{x-val}(RO_x) \\ \arctan \left| \dfrac{\text{x-val}(RO^x) - \text{x-val}(p)}{\text{y-val}(RO^x) - \text{y-val}(p)} \right| & \text{if x-val}(P) > \text{x-val}(RO^x) \\ \text{direct\_test}(P, RO_x, RO^x) & \text{otherwise} \end{cases}
$$

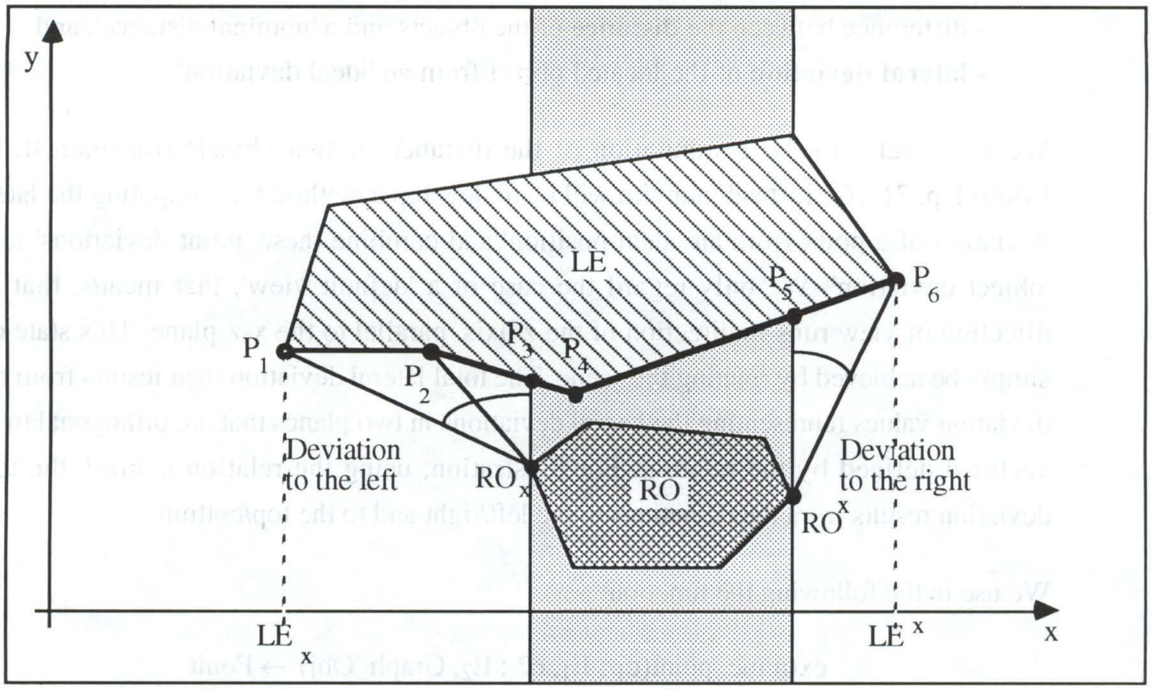


Fig.5 : Computation of the average lateral deviation

The third case in the definition of *rad* handles points P lying in the area that is marked by dots in fig.5, for this requires special treatment. In many cases it is sufficient to test whether P lies 'behind the line RO_x-RO^x'. Regarding only convex objects, this is a sufficient condition for objects lying 'behind RO'. Otherwise, it is neither a necessary nor a sufficient condition (cf. [Mo91], p. 54) so that we refine the function by the test if there is a plane RF_i in the referring front RF of RO so that P lies 'behind the plane', i.e., if f_i is a function of the form $f_i(x,y,z)=0$ describing the plane RF_i, and IP_{RO} is a point in the interior of RO, then P and IP_{RO} must be situated on different sides of the plane described by f_i, i.e., $f_i(P) * f_i(IP_{RO}) \leq 0$. We obtain the following function :

$$
\text{direct\_test}'(P, RO_x, RO^x) = \begin{cases} 0 & \text{if } \exists\, RF_i \in RF : \text{x\_val}(RF_{ix}) \leq \text{x\_val}(P) \leq \text{x\_val}(RF_i{}^x) \text{ and} \\ & \qquad f_i(P) * f_i(IP_{RO}) \leq 0 \\ \pi & \text{otherwise} \end{cases}
$$

Here, f_{y0} is the function describing the line RO_x-RO^x, that is

$$f_{y0}(x,y,z) = y - [(y\_val(RO^x) - y\_val(RO_x) / (x\_val(RO^x) - x\_val(RO_x)] * x$$

The value of *rad* is set to π for the case that P does not lie 'directly behind' RO because π represents the maximal deviation. A deviation of $\varphi > \pi$ in direction δ would correspond to a deviation of $2\pi-\varphi$ in direction $-\delta$. The test can be performed efficiently, because the number of planes in RF is finite.

Theoretically, we obtain as an average value for the total left/right deviation of the RF(LE, d) the value

$$md_{left/right}(LE,RO,behind) = \frac{\int_{RF(LE,\, d)} (rad(P, RO))\, dP}{area(RF(LE, d))}$$

It is obvious that for referring fronts which are built up a bit more complex, it will be nearly impossible to compute this exact value. Instead, we shall choose from RF(LE,d) a finite set of points which are representative for the front. These are especially points where the behavior of the function describing the referred front changes, e.g., points of intersection between edges of the front (cf. [Mo91], p. 81-84). Therefore, in the following we assume RF(LE, d) to be a finite set $\{P_1,..., P_n\}$ of points.

We thus combine the deviation values for the points of RF(LE, d) to a total deviation value by computing the arithmetic mean value of the points corresponding to the distance of them. For single deviation values $\varphi_i = rad\,(P_i, RO)$ we so obtain an average deviation value by

$$md_{left/right}(LE,RO,behind) = \frac{\sum_{i=1}^{n-1} l_x(P_i, P_{i+1}) \dfrac{(\varphi_i + \varphi_{i+1})}{2}}{P_n - P_1}$$

where $l_x(P_i, P_{i+1})$ is the x-distance of the points P_i and P_{i+1}, i.e.,

$$l_x(P_i, P_{i+1}) = |\, x\text{-}val(P_{i+1}) - x\text{-}val(P_i) \,|$$

The deviation to the top/bottom is computed the same way.

We so obtain three values, namely two values φ_{mean1}, φ_{mean2} representing the lateral deviation and one representing the difference between the distance of the objects and a nominal distance. These factors are to be valuated by functions with values in [0, 1]. These functions should also consider certain influences like the object size (cf. 3.2). We introduce functions ξ for valuating the object distance and δ for the lateral deviation. The graph of these functions resembles to the *Gauss bell-shaped graph,* and results from following definitions :

$$\xi : \text{Graph\_Obj} \times \text{Graph\_Obj} \times \text{Real} \times \text{Real} \rightarrow [0, 1]$$

$$\xi(LE, RO, d_0, \varepsilon) = \exp[-dp * \varepsilon * ((\text{distance}(LE, RO) - d_0) / S)^2]$$

where *distance* is a function computing the distance of two graphic objects and S represents the average size of LE and RO. d_0 is the nominal distance defaulting to zero and is set by statements like *50 yards left of* . ε is a factor for upsetting the graph when the relation is refined by strictness specifications like *exactly* or *about* . It causes the intensification or diminution of the influence of the object distance compared with the normal case. dp serves for the same purpose but can be set by the user for testing different valuation functions. Fig. 6a illustrates the qualitative graph behavior of ξ, using three different values for the parameter ε.

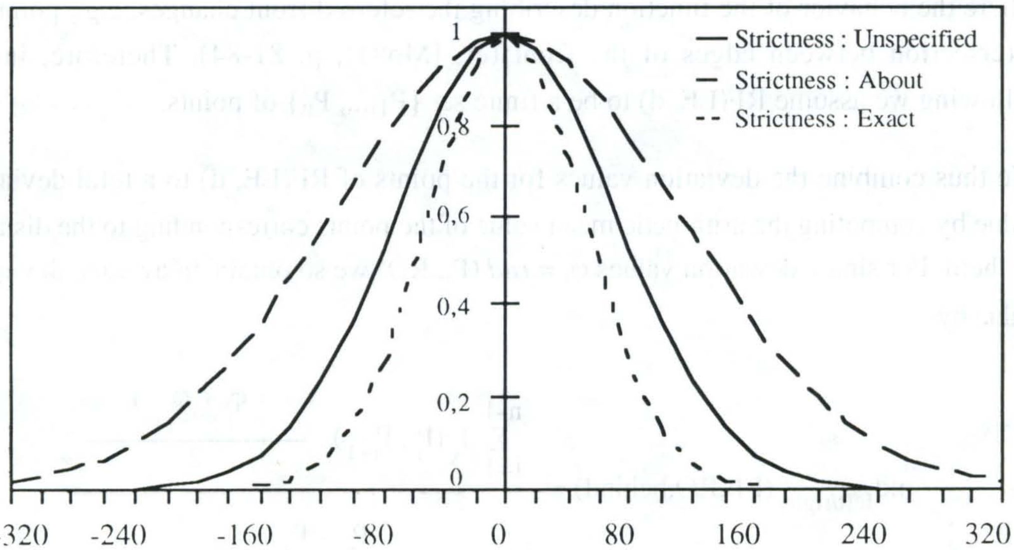


Fig 6a : Graphs for the valuation functions for the object distance

For valuating the lateral deviation we obtain

$$\delta : \text{Real} \rightarrow [0, 1]$$

$$\delta(\varphi) = \begin{cases} 0 & \text{if } |\varphi| \geq \pi/2 \\ \exp[-(1/\sqrt{2}) * ap * \varphi^2] & \text{otherwise} \end{cases}$$

ap is a factor that serves the same purpose like dp in the definition of ξ. Figure 6 illustrates the behavior of the function δ for several values of ap.

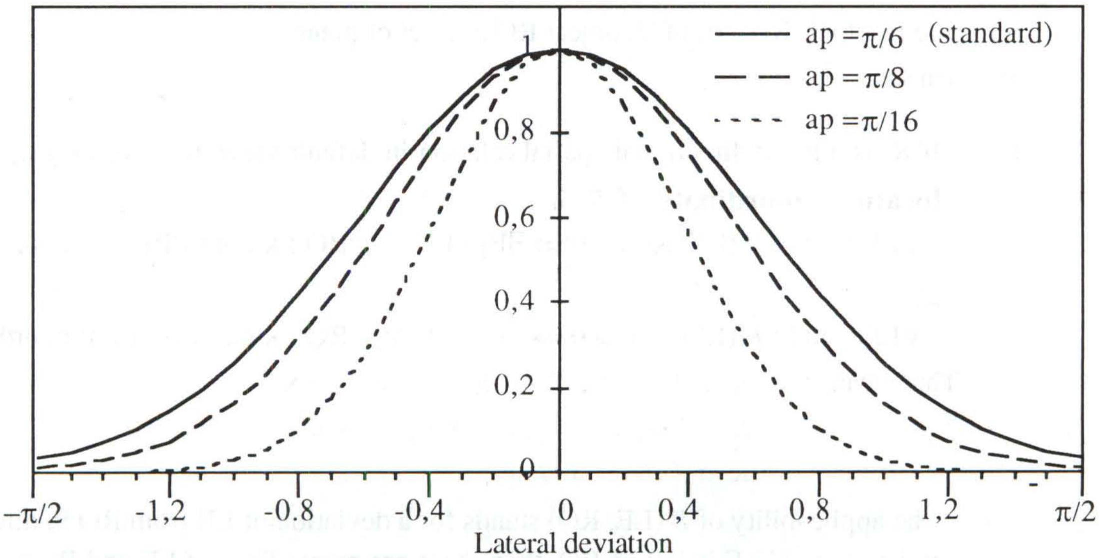


Fig.6b : Graphs of the valuation function for the lateral deviation with different values for the factor ap

Using these valuation functions, we obtain as total valuation for the relation *R* (LE, RO) the value

$$R\,(\mathrm{LE, RO}) = \delta(\varphi_{\mathrm{mean1}}) * \delta(\varphi_{\mathrm{mean2}}) * \xi(\mathrm{LE, RO}, d_0, \varepsilon)$$

The value range of this function can now be divided into intervals which correspond to discrete values (abstraction of value range), e.g. :

$$R_{\mathrm{discrete}}(\mathrm{LE, RO}) = \begin{cases} \text{true} & \text{if } R\,(\mathrm{LE, RO}) \in [0.7, 1] \\ \text{undefined} & \text{if } R\,(\mathrm{LE, RO}) \in [0.3, 0.7[\\ \text{false} & \text{otherwise} \end{cases}$$

In the following section, we shall illustrate how the method of referred fronts works for the six basic spatial directional relations, and introduce a method for handling composed directional relations.

4. Geometry Based Computation Functions for Spatial Directional Relations

4.1 Denotations

In the following, we assume the existence of a function *distance* for computing the minimal distance of two objects, and of the functions ξ and δ for valuating the distance and deviation values. Further, we suppose that we can compute the referred front RF(LE,d) of an object LE relative to a direction vector d as a finite set of points, and the referring front RF(RO, -d) of an object RO as a set of planes.

We then say :

Def.: If R is a basic directional spatial relation in default view, then $\kappa \in \{x,y,z\}$ is the **location co-ordinate** of R, if

$\forall LE\ \forall RO : R\,(LE, RO) > 0 \Rightarrow \exists P_1 \in LE,\ P_2 \in RO : \kappa\text{-coord}(P_1) < \kappa\text{-coord}(P_2)$

or

$\forall LE\ \forall RO : R\,(LE, RO) > 0 \Rightarrow \exists P_1 \in LE,\ P_2 \in RO : \kappa\text{-coord}(P_1) > \kappa\text{-coord}(P_2)$

That means : $R \in \{left\_of,\ right\_of\} \Rightarrow \kappa = x$

$R \in \{in\_front\_of,\ behind\} \Rightarrow \kappa = z$

$R \in \{above,\ below\} \Rightarrow \kappa = y$

The applicability of R (LE, RO) stands for a deviation of LE from RO in direction κ, i.e., e.g., if LE is left of RO, then there are points P_{LE} of LE and P_{RO} of RO such that $P_{LE} <_x P_{RO}$.

Def.: If $\lambda \in \{x,y,z\}$ is not location co-ordinate of R, then λ is called **deviation co-ordinate** of R. Each directional relation thus has two deviation co-ordinates. The degree of applicability results from the deviations of the located object LE in direction of the deviation co-ordinates.

Def.: If R is a basic directional relation with location co-ordinate κ, then $rel \in \{<, >\}$ is called **geometrical comparison operator** of R, if

$\forall LE\ \forall RO : R\,(LE, RO) > 0 \Rightarrow \exists P_1 \in LE,\ P_2 \in RO : rel\,(\kappa\text{-coord}(P_1),\ \kappa\text{-coord}(P_2))$

Def.: If *rel* is the geometrical comparison operator of R, then we call f_{rel} the **selection function** of *rel* , if

$$f_{rel} := \begin{cases} \min & \text{if } rel = < \\ \max & \text{if } rel = > \end{cases}$$

4.2 Basic Directional Relations

The analysis of each of the six basic relations R bases on three factors, that are the lateral deviations of LE in direction of the deviation co-ordinates of R , and the distance of the objects LE and RO. From that, we come to an algorithm for computing the value of R (LE, RO) in three steps :

(1) Rotation of the objects LE and RO to default view. The necessary rotation angle depends on
- the direction of view in deictic view
- the location of the intrinsic front of RO in intrinsic view, or
- specific influences (earth gravitation, direction of movement) in extrinsic view

For exact computation, we refer to [Mo91], p.90-91.

(2) Analysis of the lateral deviation in direction of the deviation co-ordinate

Given κ as the location co-ordinate of R, and $P_{\lambda 1},..., P_{\lambda n}$ as the points representing the referred front of LE in the κ–λ-plane (for the computation of $P_{\lambda i}$, cf. to [Mo91], p.81-83) in order of ascending λ-co-ordinates. Further may $RO_{\lambda 1}$ and $RO_{\lambda 2}$ be the extreme points of the referring front in the κ–λ-plane, i.e., the points with minimal or maximal λ-co-ordinate, respectively.

We compute the single point deviation $angle_\lambda$ ($P_{\lambda i}$) of the point $P_{\lambda i}$ = (x,y,z) by :

$$angle_\lambda(P_{\lambda i}) = \begin{cases} \arctan(\lambda\text{-coord}(RO_{\lambda 1}-P_{\lambda i})/\kappa\text{-coord}(RO_{\lambda 1}-P_{\lambda i})) & \text{if } P_{\lambda i} \leq_\lambda RO_{\lambda 1} \\ 0 & \text{if } RO_{\lambda 1} <_\lambda P_{\lambda i} <_\lambda RO_{\lambda 2}\text{, and there is a } RF_i \text{ with } f_{RF_i}(x,y,z) \geq 0 \\ \pi & \text{if } RO_{\lambda 1} <_\lambda P_{\lambda i} <_\lambda RO_{\lambda 2}\text{, and there is no } RF_i \text{ with } f_{RF_i}(x,y,z) \geq 0 \\ \arctan(\lambda\text{-coord}(P_{\lambda i}-RO_{\lambda 2})/\kappa\text{-coord}(P_{\lambda i}-RO_{\lambda 2})) & \text{if } P_{\lambda i} \geq_\lambda RO_{\lambda 2} \end{cases}$$

Here the function $<_\lambda$ ($>_\lambda$, $\leq_\lambda$, $\geq_\lambda$) compares the λ-co-ordinate of two points in space, i.e., $P_1 <_\lambda P_2 \Leftrightarrow \lambda\text{-coord}(P_1) < \lambda\text{-coord}(P_2)$.

This method is only guaranteed to work correctly on convex reference objects. An expansion to concave objects based on reduction to convex object parts is given in [Mo91]. This expansion shall not be discussed in here because it introduces no essential new ideas but only a refinement in computation.

We obtain for each of the deviation co-ordinates $\lambda \in \{\lambda_1, \lambda_2\}$ a set M_λ of deviation angles, that is $M_\lambda = \{\alpha_j \mid \alpha_j = angle_{\lambda i}(P_{\lambda i_j}), j = 1..n\}$. From that, we compute the mean lateral deviations by

$$md_{\lambda_i}(LE_\iota, RO) = [\Sigma\ [\lambda\text{-coord}(P_{\lambda_{\iota j+1}} - P_{\lambda_{\iota j}}) * (\alpha_j + \alpha_{j+1})/2]\ ]\ /\ \lambda\text{-coord}(P_{\lambda_{\iota n}} - P_{\lambda_{\iota 1}})$$

(3) Using the method introduced in [Mo91], p. 71-76 for computing the distance of spatial objects, and the valuation functions ξ and δ introduced in [Mo91], p.77-81, we compute the degree of applicability of the directional relation R by :

$$R(LE, RO) = \delta(md_{\lambda_1}(LE, RO)) * \delta(md_{\lambda_2}(LE, RO)) * \xi\ (LE, RO, d_0, \varepsilon)$$

with the standard values for d_0 and ε.

For a visualization of the computation formula *angle* (P) compare to Fig.7. The handling of the points P_1,..., P_4 corresponds to the cases 1-4 in the computation formula of the function *angle*.

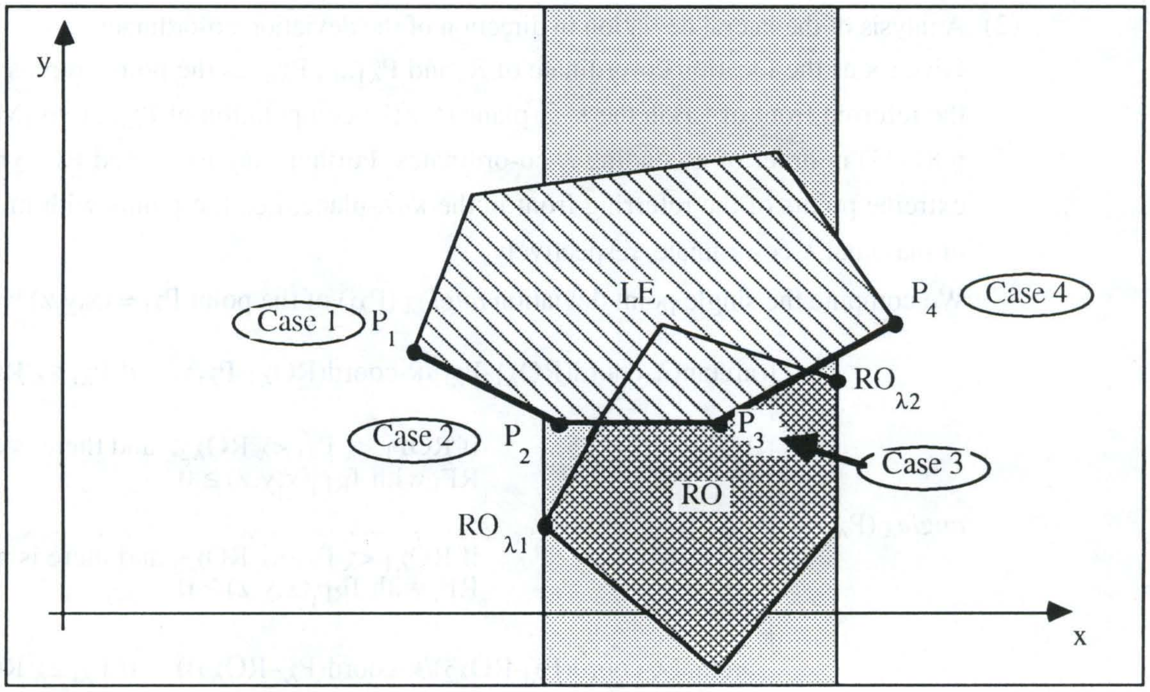


Fig.7 : Computation of the lateral deviation

4.3 Composed Directional Relations

If R_1 and R_2 are directional spatial relations, we call R a **composed spatial relation** if R results from R_1 and R_2 by a composition of R_1 and R_2 by a logical operator. We here only preoccupy ourselves with the operators AND and OR. In [Mo91] several methods for handling composed relations are introduced. Traditional methods of computing the conjunction of fuzzy values like multiplication or using the minimal value (for conjuncted relations) do not work as one can easily imagine by examples. Other methods base on the combination of the referring fronts corresponding to the relations R_1 and R_2. We will only show the method used in the implementation *COUGAR*, describing first the method for handling conjunctions and, based on that, disjunctions.

In the following, we presume that the relations to be combined do not exclude, i.e., the relations are not inverse. We then compute the degree of applicability of the conjuncted relation R_1 AND R_2(LE, RO) similarly to the method for analyzing basic directional relations. We compute a new referring front, using a new direction vector d that results from the combination of the direction vectors d_1 and d_2 of the relations R_1 and R_2. For an illustration, cf. to Fig.8.

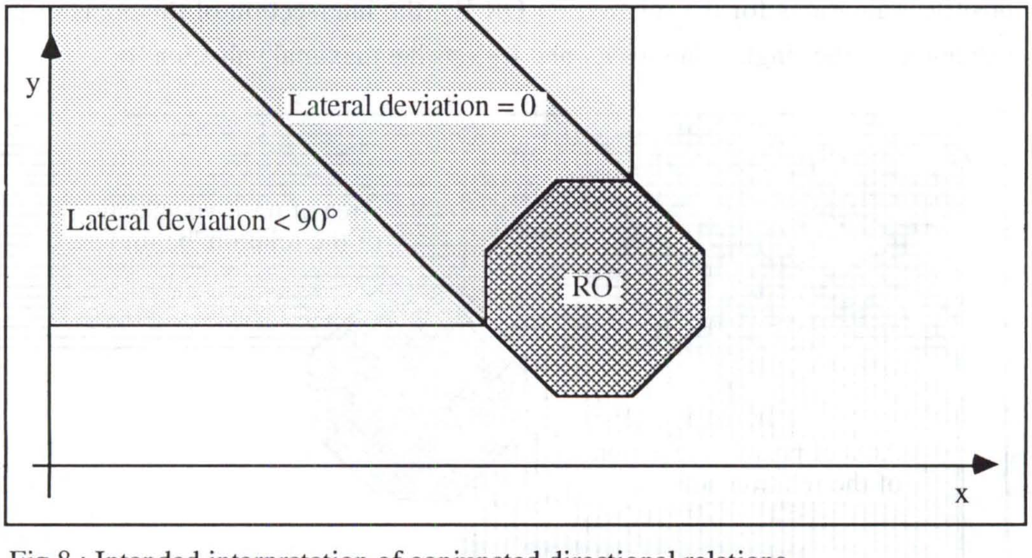


Fig.8 : Intended interpretation of conjuncted directional relations

We only value the intersection area of the single relations at positive values. If κ_i and $\lambda_{i.1}$, $\lambda_{i.2}$ are the location resp. deviation co-ordinates of the relation R_i, it ensues that $\{\lambda_{1.1}, \lambda_{1.2}\} \cap \{\lambda_{2.1}, \lambda_{2.2}\} \neq \emptyset$. Without restriction of the generality shall $\lambda_{1.1} = \lambda_{2.1}$. We see that the degree of applicability of both relations uses the deviation in direction $\lambda_{1.1} = \lambda_{2.1}$. We thus compute the deviation angles $angle_\lambda(P)$ for each point P of the referred front of LE by :

- $angle_{\lambda 1.1}$ (P) as usual as deviation in direction $\lambda_{1.1}$, and
- $angle_{\lambda 1.2 \otimes \lambda 2.2}(P) = \min(2 * angle_{\lambda'}(P), \pi)$

We multiply the value of $angle_{\lambda'}(P)$ by 2 to adapt the smaller interval $[-\pi/4, \pi/4]$ of positive valuations to the usual interval $[-\pi/2, \pi/2]$. The value $angle_{\lambda'}(P)$ computes as visualized in Fig.8.

The computation of disjunctions of directional relations is still easier, it can even be completely realized on the propositional level, i.e., only using the values of the constituent relations. We use the formula

$$R_1 \underline{OR} R_2 (P) = \begin{cases} 1 & \text{if } R_1(LE, RO) > 0 \text{ and } R_2(LE, RO) > 0 \\ R_1 (LE, RO) & \text{if } R_1 (LE, RO) > R_2 (LE, RO) \\ R_2 (LE, RO) & \text{if } R_1 (LE, RO) \leq R_2 (LE, RO) \end{cases}$$

The valuation obtained by this is shown in Fig.9. The whole hatched area obtains positive valuations for the relation R_1 OR R_2, the intersection of the areas of positive valuation for the single relations R_1 and R_2 gets the maximal valuation '1'.

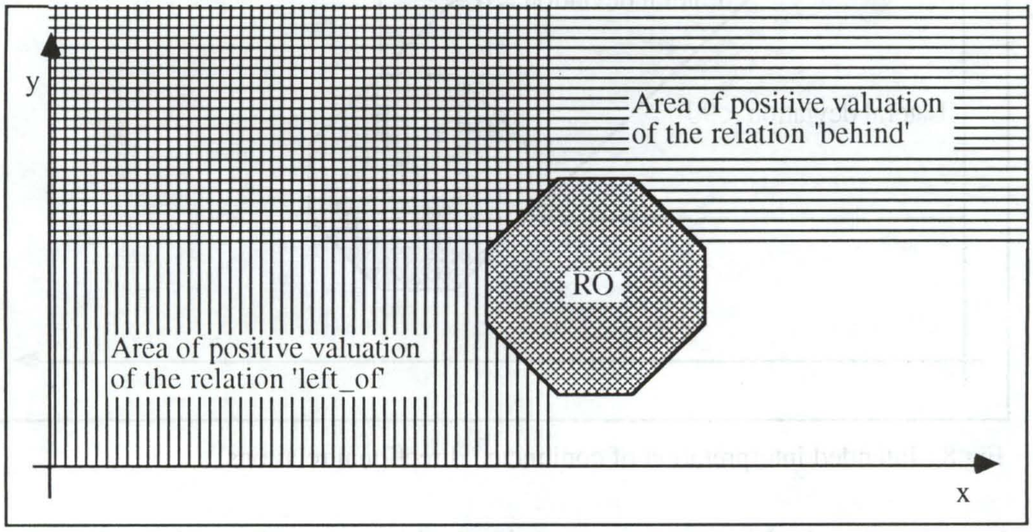


Fig.9 : Disjunctions of directional relations

4.4 Other Spatial Relations

We won't enter closer into the handling of the other spatial relations, especially the topological relations like *in, at* or *between*. Under the presumption that there are no partial intersections between spatial objects in the scene, the relation *in* can be handled quite easily. The relation *at* only bases on the object distance, and can thus be seen as a special case of the directional relations, namely a directional relation without the influence of lateral deviations. The relation *between* can be reduced to directional relations. By rotating the objects LE and RO_1, RO_2 it is always possible to achieve that the relation *between* (LE, RO_1, RO_2 corresponds to the directional relations *in_front_of* (LE, RO_1) and *behind* (LE, RO_2). So we can compute the lateral deviations of the relation *between* to : $md_{between\_rotated}(LE, RO_1, RO_2) = \max\{md_{vor}(LE, RO_1), md_{hinter}(LE, RO_2)\}$ (for more information, cf. [Mo91], p. 84-89).

5. COUGAR - A Computer Supported System For Spatial Relation Analysis

COUGAR is a system implemented in Common Lisp on Symbolics lisp machines. It primary aim is to visualize the methods of different algorithms for analyzing spatial relations, and compare these algorithms as for time complexity and correctness. The system allows the user to create own scenes and apply his algorithms to them. For the implemented algorithms (method of referred fronts, and simplified centroid method which reduces all objects to center points and then tests the spatial relations of these) there

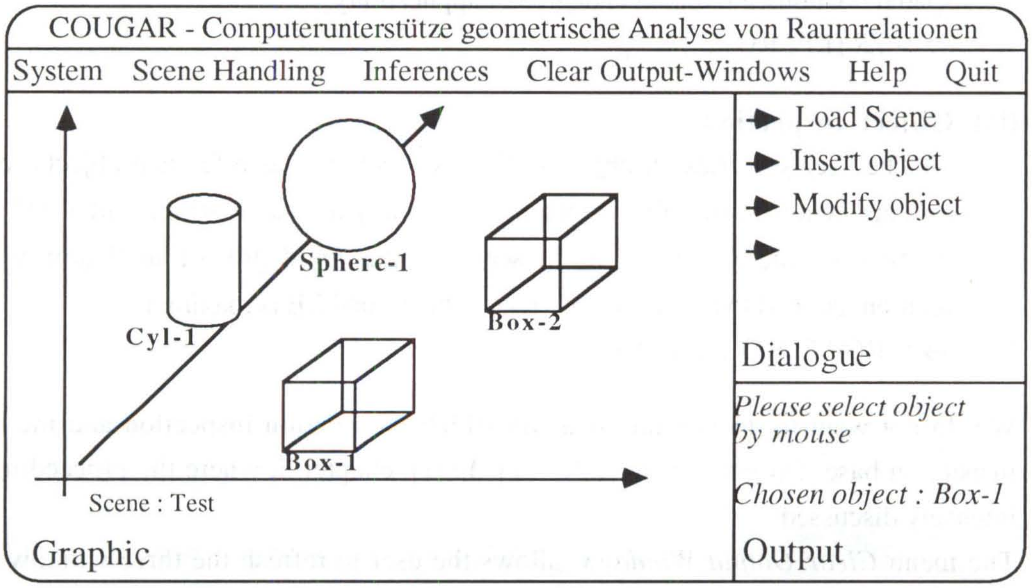


Fig.10 : User interface of the system COUGAR

is an optional **trace mode** for a better visualization of their proceeding. Further there exists a component for measuring the runtime of an algorithm, applied to objects in the scene.

Fig.10 shows the main menu of the system. *COUGAR* is completely mouse controlled, but for a faster handling all the commands can also be typed in by keyboard on any menu level. We shall now briefly explain the meaning of the single sub-menus.

The menu *System* offers features for resetting the system variables, setting system parameters and a system information. *Scene Handling* offers menu points for creating scenes and inserting resp. modifying objects in a given scene by mouse. *Inferences* offers the following features for handling spatial relations :

(i) **Relation verification**

$\rightarrow$ The user enters a relation, an (optional) relation specification and several objects that serve as reference and located objects. The system computes the degree of applicability between the objects

$\rightarrow$:- R(LE, RO_1,..., RO_n)

(ii) **Relation inspection**

$\rightarrow$ The user specifies reference and located object, and *COUGAR* computes the spatial relation with maximal degree of applicability

$\rightarrow$:- *?R* (LE, RO_1,..., RO_n)

(iii) **Object inspection**

$\rightarrow$ The user specifies an object in the scene serving as reference objects (or two objects when using the relation *between*) and a relation, and *COUGAR* investigates the objects LE in the scene, so that the degree of applicability of the relation, applied to the chosen reference objects and LE is maximal

$\rightarrow$:- R(*?LE*, RO_1,..., RO_n)

We do not want to deepen the ideas on which the relation inspection and the object inspection base. On interest, we refer to [Mo91], chapter 5, where the proceedings are intensely discussed.

The menu *Clear Output Windows* allows the user to refresh the three windows. The menu point *Help* can be found on any menu level and causes a help message on the menu points of the actual menu to be displayed on the in the output window.

6. Conclusion And Further Work

Not all aspects of spatial relations are completely resolved by the system *COUGAR* and the method of referred fronts used in it. This chapter will show up the main weak points of this method.

1.) Object inspection

COUGAR uses a quite primitive proceeding for computing the located object LE in the scene for which the degree of applicability is maximal. Except for some minor improvements, the system tests the relation for <u>all</u> the objects in the scene. It can be easily seen that this proceeding is not senseful, especially if there are a lot of objects in the scene. The system lacks in an additional data structure that allows to conclude an object (i.e., the object's name) from its position, instead the position from the object, as the actual data structure does. The basic idea for an efficient implementation of the object inspection is to compute the 'areas of maximal degree of applicability' relative to the reference object and the relation, and coming from this area, detect the objects that are lying in it. [Mo91],p.121-128, introduces informally a method for representing spatial knowledge that allows the conclusion of an object from its position.

2.) External influences

COUGAR only regards influences to the analysis of spatial relations that come from the located object and the reference objects themselves. But over that, we can still have influences by other objects, e.g., a wall or any other obstacle impenetrable for the regarding agent between reference and located object can eventually diminish the degree of applicability of the spatial relation between them. The problem here is that it is quite difficult to detect objects in the surrounding of the regarded objects using only the data structure introduced. A second data structure, as shown under 1.), could solve this problem (cf. also [Mo91], p.121-125).

3.) Extensions of the data structure

The introduced data structure is not capable of representing all kinds of geometric objects, although they can all be approximated as nearly as desired. For an exact representation, we need some modifications to the data structure (cf. [Mo91], p. 126-128). The question indeed is if it is necessary to do so, namely if approximations won't do i the majority of cases.

4.) Further query types

COUGAR only offers three basic types of queries, namely relation verification, relation inspection and object inspection. There are lots of more query types imaginable of which the most important shall be introduced here :

Location inspection $\rightarrow$:- *?R* (LE, ?RO*)

$\rightarrow$ We here don't ask for the relation of LE to a special object but for the location of LE itself, i.e., we ask 'Where is LE ?'. This requires first the choice of one or more suitable reference objects, and then the determination of the relation between them. As we see, we have a similar problem as under 1.), i.e., we have to locate RO by its position close by LE

5.) Integration of time

Time and space are domains in knowledge representation that are closely related, not only in their way of representation. The representation of movements requires both, representation of space as well as of time. If we can also regard moving objects, it will be possible to handle dynamic relations as *past* or *along*. In general, spatial relations must be extended by a time slot, showing up the time of validity of the relation (*R* (LE, RO_1,..., RO_n, t)). We so obtain two demands :

(i) Adequate representation of moving objects, and

(ii) Integration of 'time inferences'

The first demand shall not be treated nearer in here, and refer to [ABHR86a]. The second point refers to new inferences of which the most important shall be specified here :

(i) Temporal relation verification $\rightarrow$:- *R* (LE, RO*, t)
Temporal relation inspection $\rightarrow$:- *?R* (LE, RO*, t)
Temporal object inspection $\rightarrow$:- *R* (?LE, RO*, t)

(ii) Time inspection $\rightarrow$:- *R* (LE, RO*, ?t)

t can here be either a time point or a time interval, a distinction problem that is well known in time representation. If t is an interval, it produces additional problems, e.g., if the degree of applicability of a relation *R* (LE, RO*, I) shall be the minimal, maximal or average degree of applicability in the interval I (comp.[Mo91], p.128-134). Eventually, we so obtain different type of queries, e.g. the question for the maximal (minimal, average) degree of applicability of a spatial relation in a time interval I.

As we see, there are still lots of possible extensions to the *COUGAR* system. Future will show how the method of referred fronts will prove to be good. There are several other approaches to the handling of spatial relations (cf. [HP88], [Pr90] for the approach of the working group LILOG, and [ABHR 86a] for the one of the VITRA project). More examples are shown up in [Mo91], p.137-142. An intensive comparison will show which approach is the best for which problem, and perhaps, only a mixture between different approaches will really prove to be able to compete with human abilities of space representation and analysis.

7. References

[ABHR 86a] Elisabeth Andre, Guido Bosch, Gerd Herzog, Thomas Rist
Characterizing trajectories of moving objects using natural language descriptions
SFB 314 (VITRA), Memo Nr. 5, March 1986
in : ECAI 1986 - The 7th European Conference on Artificial Intelligence
Proceedings, Vol.2
Brighton 1986, p. 1-8

[ABHR 86b] Elisabeth Andre, Guido Bosch, Gerd Herzog, Thomas Rist
Coping with the intrinsic and deictic uses of spatial prepositions
SFB 314 (VITRA), Report Nr. 9, July 1986
in : Ph. Jorrand, S. Sgurev (ed.)
Artificial Intelligence II : Proceedings of AIMSA-86
Amsterdam: North Holland, 1987, p. 357-382

[BM 91a] Peter Breuer, Jürgen Müller
Two level representation for spatial relations, Part One
DFKI Research Report RR-91-14, May 1991

[BM 91b] Peter Breuer, Jürgen Müller
Defining almost exact spatial relations
Proceedings of the First World Conference on Fundamentals of AI
Paris, July 1991

[DS 82] Roger M. Downs, David Stea
Maps in Mind
Harper & Row, Publishers, New York 1982

[HP 88] Christopher Habel, Simone Pribbenow
Gebietskonstituierende Prozesse
LILOG-Report 18
IBM Deutschland GmbH, Stuttgart

[Ko 80] Stephen Michael Kosslyn
Image and Mind
Harvard University Press : Cambridge, England

[Mo 91] Jörg-Peter Mohren
Geometrische Verfahren zur RAUM-Repräsentation
Dokumentation zur Diplomarbeit am FB Informatik der Universität Kaiserslautern, July 1991

[Pr 90] Simone Pribbenow
Interaktion von propositionalen und bildhaften Repräsentationen
in : Christian Freksa, Christopher Habel
Repräsentation und Verarbeitung räumlichen Wissens
Informatik-Fachberichte 245, p. 156-174
Springer-Verlag, Berlin

[Re 88] Gudula Retz-Schmidt
Various views on spatial prepositions
AI Magazine 9 (2) 1988, p. 95-105

Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

DFKI
-Bibliothek-
PF 2080
D-6750 Kaiserslautern
FRG

DFKI Publikationen

Die folgenden DFKI Veröffentlichungen sowie die aktuelle Liste von allen bisher erschienenen Publikationen können von der oben angegebenen Adresse bezogen werden.
Die Berichte werden, wenn nicht anders gekennzeichnet, kostenlos abgegeben.

DFKI Publications

The following DFKI publications or the list of all publisched papers so far can be ordered from the above address.
The reports are distributed free of charge except if otherwise indicated.

DFKI Research Reports

RR-91-10
Franz Baader, Philipp Hanschke: A Scheme for Integrating Concrete Domains into Concept Languages
31 pages

RR-91-11
Bernhard Nebel: Belief Revision and Default Reasoning: Syntax-Based Approaches
37 pages

RR-91-12
J.Mark Gawron, John Nerbonne, Stanley Peters: The Absorption Principle and E-Type Anaphora
33 pages

RR-91-13
Gert Smolka: Residuation and Guarded Rules for Constraint Logic Programming
17 pages

RR-91-14
Peter Breuer, Jürgen Müller: A Two Level Representation for Spatial Relations, Part I
27 pages

RR-91-15
Bernhard Nebel, Gert Smolka:
Attributive Description Formalisms ... and the Rest of the World
20 pages

RR-91-16
Stephan Busemann: Using Pattern-Action Rules for the Generation of GPSG Structures from Separate Semantic Representations
18 pages

RR-91-17
Andreas Dengel, Nelson M. Mattos:
The Use of Abstraction Concepts for Representing and Structuring Documents
17 pages

RR-91-18
John Nerbonne, Klaus Netter, Abdel Kader Diagne, Ludwig Dickmann, Judith Klein:
A Diagnostic Tool for German Syntax
20 pages

RR-91-19
Munindar P. Singh: On the Commitments and Precommitments of Limited Agents
15 pages

RR-91-20
Christoph Klauck, Ansgar Bernardi, Ralf Legleitner
FEAT-Rep: Representing Features in CAD/CAM
48 pages

RR-91-21
Klaus Netter: Clause Union and Verb Raising Phenomena in German
38 pages

RR-91-22
Andreas Dengel: Self-Adapting Structuring and Representation of Space
27 pages

RR-91-23
Michael Richter, Ansgar Bernardi, Christoph Klauck, Ralf Legleitner: Akquisition und Repräsentation von technischem Wissen für Planungsaufgaben im Bereich der Fertigungstechnik
24 Seiten

RR-91-24
Jochen Heinsohn: A Hybrid Approach for Modeling Uncertainty in Terminological Logics
22 pages

RR-91-25
Karin Harbusch, Wolfgang Finkler, Anne Schauder:
Incremental Syntax Generation with Tree Adjoining Grammars
16 pages

RR-91-26
M. Bauer, S. Biundo, D. Dengler, M. Hecking, J. Koehler, G. Merziger:
Integrated Plan Generation and Recognition - A Logic-Based Approach -
17 pages

RR-91-27
A. Bernardi, H. Boley, Ph. Hanschke, K. Hinkelmann, Ch. Klauck, O. Kühn, R. Legleitner, M. Meyer, M. M. Richter, F. Schmalhofer, G. Schmidt, W. Sommer:
ARC-TEC: Acquisition, Representation and Compilation of Technical Knowledge
18 pages

RR-91-28
Rolf Backofen, Harald Trost, Hans Uszkoreit:
Linking Typed Feature Formalisms and Terminological Knowledge Representation Languages in Natural Language Front-Ends
11 pages

RR-91-29
Hans Uszkoreit: Strategies for Adding Control Information to Declarative Grammars
17 pages

RR-91-30
Dan Flickinger, John Nerbonne:
Inheritance and Complementation: A Case Study of *Easy* Adjectives and Related Nouns
39 pages

RR-91-31
H.-U. Krieger, J. Nerbonne:
Feature-Based Inheritance Networks for Computational Lexicons
11 pages

RR-91-32
Rolf Backofen, Lutz Euler, Günther Görz:
Towards the Integration of Functions, Relations and Types in an AI Programming Language
14 pages

RR-91-33
Franz Baader, Klaus Schulz:
Unification in the Union of Disjoint Equational Theories: Combining Decision Procedures
33 pages

RR-91-34
Bernhard Nebel, Christer Bäckström:
On the Computational Complexity of Temporal Projection and some related Problems
35 pages

RR-91-35
Winfried Graf, Wolfgang Maaß: Constraint-basierte Verarbeitung graphischen Wissens
14 Seiten

RR-92-01
Werner Nutt: Unification in Monoidal Theories is Solving Linear Equations over Semirings
57 pages

RR-92-02
Andreas Dengel, Rainer Bleisinger, Rainer Hoch, Frank Hönes, Frank Fein, Michael Malburg:
Π_{ODA}: The Paper Interface to ODA
53 pages

RR-92-03
Harold Boley:
Extended Logic-plus-Functional Programming
28 pages

RR-92-04
John Nerbonne: Feature-Based Lexicons: An Example and a Comparison to DATR
15 pages

RR-92-05
Ansgar Bernardi, Christoph Klauck, Ralf Legleitner, Michael Schulte, Rainer Stark:
Feature based Integration of CAD and CAPP
19 pages

RR-92-06
Achim Schupetea: Main Topics od DAI: A Review
38 pages

RR-92-07
Michael Beetz:
Decision-theoretic Transformational Planning
22 pages

RR-92-08
Gabriele Merziger: Approaches to Abductive Reasoning - An Overview -
46 pages

RR-92-09
Winfried Graf, Markus A. Thies:
Perspektiven zur Kombination von automatischem Animationsdesign und planbasierter Hilfe
15 Seiten

RR-92-11
Susane Biundo, Dietmar Dengler, Jana Koehler:
Deductive Planning and Plan Reuse in a Command Language Environment
13 pages

RR-92-13
Markus A. Thies, Frank Berger:
Planbasierte graphische Hilfe in objektorientierten Benutzungsoberflächen
13 Seiten

RR-92-14
Intelligent User Support in Graphical User Interfaces:
1. InCome: A System to Navigate through Interactions and Plans
Thomas Fehrle, Markus A. Thies
2. Plan-Based Graphical Help in Object-Oriented User Interfaces
Markus A. Thies, Frank Berger

22 pages

RR-92-15
Winfried Graf: Constraint-Based Graphical Layout of Multimodal Presentations
23 pages

RR-92-16
Jochen Heinsohn, Daniel Kudenko, Berhard Nebel, Hans-Jürgen Profitlich: An Empirical Analysis of Terminological Representation Systems
38 pages

RR-92-17
Hassan Aït-Kaci, Andreas Podelski, Gert Smolka: A Feature-based Constraint System for Logic Programming with Entailment
23 pages

RR-92-18
John Nerbonne: Constraint-Based Semantics
21 pages

RR-92-19
Ralf Legleitner, Ansgar Bernardi, Christoph Klauck
PIM: Planning In Manufacturing using Skeletal Plans and Features
17 pages

RR-92-20
John Nerbonne: Representing Grammar, Meaning and Knowledge
18 pages

RR-92-21
Jörg-Peter Mohren, Jürgen Müller
Representing Spatial Relations (Part II) -The Geometrical Approach
25 pages

RR-92-22
Jörg Würtz: Unifying Cycles
24 pages

RR-92-24
Gabriele Schmidt: Knowledge Acquisition from Text in a Complex Domain
20 pages

DFKI Technical Memos

TM-91-09
Munindar P. Singh: On the Semantics of Protocols Among Distributed Intelligent Agents
18 pages

TM-91-10
Béla Buschauer, Peter Poller, Anne Schauder, Karin Harbusch: Tree Adjoining Grammars mit Unifikation
149 pages

TM-91-11
Peter Wazinski: Generating Spatial Descriptions for Cross-modal References
21 pages

TM-91-12
Klaus Becker, Christoph Klauck, Johannes Schwagereit: FEAT-PATR: Eine Erweiterung des D-PATR zur Feature-Erkennung in CAD/CAM
33 Seiten

TM-91-13
Knut Hinkelmann:
Forward Logic Evaluation: Developing a Compiler from a Partially Evaluated Meta Interpreter
16 pages

TM-91-14
Rainer Bleisinger, Rainer Hoch, Andreas Dengel:
ODA-based modeling for document analysis
14 pages

TM-91-15
Stefan Bussmann: Prototypical Concept Formation
An Alternative Approach to Knowledge Representation
28 pages

TM-92-01
Lijuan Zhang:
Entwurf und Implementierung eines Compilers zur Transformation von Werkstückrepräsentationen
34 Seiten

TM-92-02
Achim Schupeta: Organizing Communication and Introspection in a Multi-Agent Blocksworld
32 pages

TM-92-03
Mona Singh
A Cognitiv Analysis of Event Structure
21 pages

DFKI Documents

D-91-10
Donald R. Steiner, Jürgen Müller (Eds.):
MAAMAW'91: Pre-Proceedings of the 3rd European Workshop on „Modeling Autonomous Agents and Multi-Agent Worlds"
246 pages
Note: This document is available only for a nominal charge of 25 DM (or 15 US-$).

D-91-11
Thilo C. Horstmann: Distributed Truth Maintenance
61 pages

D-91-12
Bernd Bachmann:
$\mathrm{H}^{\mathrm{iera}}\mathrm{C}_{\mathrm{on}}$ - a Knowledge Representation System with Typed Hierarchies and Constraints
75 pages

D-91-13
International Workshop on Terminological Logics
Organizers: Bernhard Nebel, Christof Peltason, Kai von Luck
131 pages

D-91-14
Erich Achilles, Bernhard Hollunder, Armin Laux, Jörg-Peter Mohren: KRIS: Knowledge Representation and Inference System
- Benutzerhandbuch -
28 Seiten

D-91-15
Harold Boley, Philipp Hanschke, Martin Harm, Knut Hinkelmann, Thomas Labisch, Manfred Meyer, Jörg Müller, Thomas Oltzen, Michael Sintek, Werner Stein, Frank Steinle:
µCAD2NC: A Declarative Lathe-Worplanning Model Transforming CAD-like Geometries into Abstract NC Programs
100 pages

D-91-16
Jörg Thoben, Franz Schmalhofer, Thomas Reinartz:
Wiederholungs-, Varianten- und Neuplanung bei der Fertigung rotationssymmetrischer Drehteile
134 Seiten

D-91-17
Andreas Becker:
Analyse der Planungsverfahren der KI im Hinblick auf ihre Eignung für die Abeitsplanung
86 Seiten

D-91-18
Thomas Reinartz: Definition von Problemklassen im Maschinenbau als eine Begriffsbildungsaufgabe
107 Seiten

D-91-19
Peter Wazinski: Objektlokalisation in graphischen Darstellungen
110 Seiten

D-92-01
Stefan Bussmann: Simulation Environment for Multi-Agent Worlds - Benutzeranleitung
50 Seiten

D-92-02
Wolfgang Maaß: Constraint-basierte Plazierung in multimodalen Dokumenten am Beispiel des Layout-Managers in WIP
111 Seiten

D-92-03
Wolfgan Maaß, Thomas Schiffmann, Dudung Soetopo, Winfried Graf: LAYLAB: Ein System zur automatischen Plazierung von Text-Bild-Kombinationen in multimodalen Dokumenten
41 Seiten

D-92-06
Hans Werner Höper: Systematik zur Beschreibung von Werkstücken in der Terminologie der Featuresprache
392 Seiten

D-92-08
Jochen Heinsohn, Bernhard Hollunder (Eds.):
DFKI Workshop on Taxonomic Reasoning Proceedings
56 pages

D-92-09
Gernod P. Laufkötter: Implementierungsmöglichkeiten der integrativen Wissensakquisitionsmethode des ARC-TEC-Projektes
86 Seiten

D-92-10
Jakob Mauss: Ein heuristisch gesteuerter Chrat-Parser für attributierte Graph-Grammatiken
87 Seiten

D-92-15
DFKI Wissenschaftlich-Technischer Jahresbericht 1991
130 Seiten

D-92-21
Anne Schauder: Incremental Syntactic Generation of Natural Language with Tree Adjoining Grammars
57 pages

Representing Spatial Relations (Part II) -The Geometrical Approach

Jörg-Peter Mohren, Jürgen Müller

RR-92-21

Research Report