Point-Based Descriptions of Interval Relations

Kurt Eberle
Institut für Maschinelle Sprachverarbeitung
Universität Stuttgart
Azenbergstr. 12
7000 Stuttgart 1

Abstract

Temporal text understanding and question answering presupposes the availability of a component which computes the transitive closure of a set of statements about interval relations. Vilain and Kautz have addressed the question of computing the transitive closure by translating such statements into equivalent point-based descriptions. The problem is that there are basic statements about interval relations which have to be translated into disjunctions of point-based descriptions. Therefore, in order to handle the general case economically, it is necessary to have translations with a minimum number of disjuncts for each and every case.

In this paper we present an algorithm which computes a best translation, i.e. a translation which consists of a minimal number of disjuncts. This number amounts at most to 5.

1 Introduction

Understanding a (narrative) natural language text involves relating temporally the events introduced by the text to each other. Therefore, inferences using transitivity rules for temporal relations have to be drawn in order to account for events and other temporal units that are not directly connected in the text by linguistic means. Such inferences for sets of statements about the temporal relations between intervals are efficiently made by the path consistency algorithm suggested by Allen (cf. [All83]). However, this algorithm is incomplete. For this reason it has been suggested, mainly by Kautz and Vilain, that one should translate the given information about intervals into information about the endpoints of the intervals and tackle the problem in the point based framework. The point-based path consistency algorithm is complete provided it is extended by an additional test routine. This complete algorithm is, like Allen's algorithm, of polynomial complexity (cf. [VK86], [vB89], [VKvB89], [vB90], [Haj91]).

The translation of (underspecified) interval relations into relations of the corresponding endpoints is a problem in so far as there are cases where this translation necessarily ends up in a disjunction of basic statements. For instance, if we know that i occurs before or after j (call this information RS), we get two alternative translations into point descriptions, $T1_{RS}$ and $T2_{RS}$, the relevant parts of which we can render as follows (with $s(I)$ for the start and $e(I)$ for the end of I):

$$T1_{RS} = \{\ldots, s(i)\ \{<\}\ e(j), e(i)\ \{<\}\ s(j), \ldots\}$$
$$T2_{RS} = \{\ldots, s(i)\ \{>\}\ e(j), e(i)\ \{>\}\ s(j), \ldots\}$$

RS thus is expressed by the disjunction of $T1_{RS}$ and $T2_{RS}$. Note, that merging $T1_{RS}$ and $T2_{RS}$ by combining their decisive constraints to the conditions $s(i)\ \{<,>\}\ e(j), e(i)\ \{<,>\}\ s(j)$, standing for $s(i)$ before or after $e(j)$ and $e(i)$ before or after $s(j)$, and by listing the other constraints accordingly results in a set T_{RS} which indeed reflects the structural possibilities of RS but which, in addition, allows for other solutions like i overlaps j.

We learn from this that the number of alternative translations of sets Σ of interval statements can be exponential. (For a set Σ stating that i before or after j and j before or after k we would get four translations). So, in order to compute the transitive closure of Σ by translation into the point case we must take into account that there are cases where we must apply the revised path consistency algorithm for points to an exponential number of input sets. (However, the retranslation is unique). Kautz and Vilain therefore restrict themselves to Σs which have a unique translation. They show that the general problem is NP-complete. In spite of this result we think that, in particular temporal text understanding cannot be restricted to the case of *convex relations*, where convex relations mark the special case of the disjunctive use of Allen's relations with unique translation into point based descriptions (cf. for instance [Nö89]). Van Beek calls this set of relations $SP^{\neq}$ ([vB89]). It is true that most relations introduced by temporal conjunctions, by temporal adverbs or by the temporal incorporation of the event of a new sentence in the representation of the preceding text are convex relations.[1] But, nevertheless there are cases where this default is suspended. For instance, in

Last year a lot of important things happend$_{e_1}$. *Peter got married*$_{e_2}$. *John made a tour through the United States*$_{e_3}$ *and another through Poland*$_{e_4}$. *Mary won in the lotterie*$_{e_5}$.

the rhetorical or discourse relation holding between the events of sentences 2-4, confirmed by sentence 1, is *enumeration.*[2] The order between e_2-e_5 obviously is of no interest to the author. This is part of the rhetorical function. But this does not exclude that one is aware of the fact that e_3 occurs before or after e_4, since tours through the United States and tours through Poland undertaken by the same agent cannot overlap. Though not in focus in the text bit presented, this knowledge can be used by the author and necessitated by the recipient of the text in order to strengthen the global temporal structure of the whole text. For instance, these travelling events can be used as temporal anchors for other events - *when John was in Poland,* ϕ_{e6} - thus having an impact on the temporal relations of other temporal entities of the text. High quality text understanding and in particular question answering must be able to deal with such cases and, clearly, the relation between $e3$ and $e4$ is not convex. There are other types of information that introduce non-convex relations, but we cannot go into detail with this here. Since the narratives that natural language systems deal with are normally relatively short, introducing only a restricted number of temporal entities, and since it is relatively seldom that non-convex relations are introduced (mainly by background knowledge accompanying the semantic analysis), in practice, the exponentiality of the closure algorithm does not lead to intractability. Nevertheless, it is exactly for this reason, that it is necessary to design this algorithm as efficient as possible. Therefore, it is very important to reduce the number of alternative translations of interval statements to a minimum. This is what we are concerned with here.

In this paper we show that the number of disjuncts necessary to express the information of an interval statement amounts at most to 5. [3] We present an algorithm which computes a best translation, i.e. a translation which consists of a minimal number of disjuncts. This will be done in section 4. In the sections 2 and 3 we list preliminary definitions and tackle the problem heuristically.

2 Some Definitions

Allen uses the following relation symbols:

Definition: REL, the set of Allen symbols:

$$REL = \left\{ \begin{array}{lllll} b\ (before) & bi\ (before\ inverse) & s\ (starts) & & si\ (starts\ inverse) \\ m\ (meets) & mi\ (meets\ inverse) & d\ (during) & id\ (identical) & di\ (during\ inverse) \\ o\ (overlaps) & oi\ (overlaps\ inverse) & f\ (finishes) & & fi\ (finishes\ inverse) \end{array} \right\}$$

Allen's path consistency algorithm is applied to sets consisting of statements like, for instance,

- $i\ \{b, o\}\ j$, which stands for: *i occurs before j or i overlaps j*
- $k\ \{bi, mi\} j$, which stands for *k occurs after j or is met by j*

(from what we conclude by means of Allen's algorithm that $i\ \{b\} k$).

[1] Compare, for instance, studies on temporal semantics like [KR83], [KR85], [Hin86], [Her90], [Ebe91].

[2] For rhetorical or discourse relations compare for instance [KR85], [TM87], [AL91], [Ebe91].

[3] This confirms the result that can be taken from the work about convex relations reported in [Nö89].

Definition: $PREL$, the set of point relation symbols, $BPREL$, the set of basic point relation symbols

$PREL = \{$l,g,e, le, ue, ge, 0$\}$

$BPREL = \{$l,g,e$\}$

l, g, e, le, ue, ge, 0 stand in turn for temporally *less* (precedence), *greater* (succession), *equal, less or equal, unequal, greater or equal* and *no information*.

The intuition is that the $PREL$-symbols stand for sets of pairs of points when interpreted modeltheoretically in the point substructure of a model which satisfies suited axioms for points and intervals. It is clear that this set of axioms has to be a superset of Allen's interval axioms and that the axioms regulating the interplay of points and intervals has to be compatible with Allen's axioms. We omit being more specific about this. (Compare for instance [Bit86]).

Against this background it is natural to stipulate $PREL$ to be partially ordered by means of $\leq_{ps}$ according to a join semi-lattice operation $\sqcup_{ps}$ which reflects the union in point structures.

Definition: The join semi-lattice $PREL$
$< PREL, \sqcup_{ps} >$ is a complete atomic $\sqcup_{ps}$-semi lattice with the set of atoms $BPREL$ and
l $\sqcup_{ps}$ g = ue , l $\sqcup_{ps}$ e = le , g $\sqcup_{ps}$ e = ge , ue $\sqcup_{ps}$ le = 0, ge $\sqcup_{ps}$ le = 0, ue $\sqcup_{ps}$ ge = 0.
$\leq_{ps}$ is the partial order resulting from $\sqcup_{ps}$.

The intuitive meaning of Allen's symbols now can be expressed by means of relations between the endpoints of intervals (the **s**tart of i, $s(i)$, the **e**nd of i, $e(i)$) using the symbols introduced (I for the set of intervals). For instance, we will have:

$$\forall i, j \in I : i\ o\ j \leftrightarrow s(i) < s(j) < e(i) < e(j)$$
$$\forall i, j \in I : i\ s\ j \leftrightarrow s(i) = s(j) < e(i) < e(j)$$

In the following we will use a compact notation in order to express an Allen statement by a point based description. Therefore we define 4-place-*vectors*.

Definition: VEC, the set of vectors

$$VEC := \{[A, B, C, D] \mid A, B, C, D \in PREL\}$$

Convention:
Be V a tuple (for instance $V \in VEC$).
Then V^i is the i-th projection of V, i.e. the i-th slot of the tuple (if existent, otherwise it is not defined).

For instance, if $V \in VEC$ with $V = [A, B, C, D]$, then $V^2 = B, V^4 = D, \ldots$.

Now, we require that for intervalls i, j, $PREL$-symbols A,B,C,D:
i {[A,B,C,D]} j stands for $s(i)$ A $s(j), s(i)$ B $e(j), e(i)$ C $s(j), e(i)$ A $e(j)$

Of course, as in the case of Allen statements
- $i\ \{V_1, \ldots, V_n\}\ j$ stands for $i\ V_1\ j \vee \ldots \vee i\ V_n\ j$ (for $V_1, \ldots, V_n \in VEC$).

It turns out that, on the basis of such axioms as mentioned above, such vector descriptions are sufficient to retain the information of Allen statements.

Now we use $\leq_{ps}$ to define a partial order for VEC.

Definition: "V contains V' "

$$\forall V, V' \in VEC : V \leq_{vs} V' \leftrightarrow \bigwedge_{i \in \{1,\ldots,4\}} (V^i \leq_{ps} V'^i)$$

Definition: $AVEC$, the set of atomic vectors

$$\forall V \in VEC : V \in AVEC \leftrightarrow \neg(\exists V' \in VEC : V' <_{vs} V)$$

(where $V' <_{vs} V \leftrightarrow V' \leq_{vs} V \wedge V' \neq V$).

It is clear that $AVEC = \{[A, B, C, D] \mid A, B, C, D \in BPREL\}$.
Having in mind the meaning and use of vectors as attributed to them here, it is clear what a *canonical translation* (ct) of the Allen symbols will look like.

Definition: The canonical translation of the Allen symbols: the function ct

$ct : REL \rightarrow AVEC$ with

$ct(b)$ = [l,l,l,l]	$ct(m)$ = [l,l,e,l]	$ct(o)$ = [l,l,g,l]	$ct(s)$ = [e,l,g,l]	
$ct(bi)$ = [g,g,g,g]	$ct(mi)$ = [g,e,g,g]	$ct(oi)$ = [g,l,g,g]	$ct(si)$ = [e,l,g,g]	$ct(id)$ = [e,l,g,e]
$ct(d)$ = [g,l,g,l]	$ct(f)$ = [g,l,g,e]	$ct(fi)$ = [l,l,g,e]	$ct(di)$ = [l,l,g,g]	

A vector statement $i\ V\ j$ reflects a set of relational possibilities with respect to Allen's symbols, those for which the canonical translation V' is contained within V. For this reason we call such V's a *solution* of V and more generally, abstracting from particular Vs:

Definition: SOL, the set of possible solutions
We call $V \in AVEC$ a possible solution iff $V \in SOL$, where:

$$\forall V \in VEC : V \in SOL \leftrightarrow (\exists R \in REL : \ ct(R) = V)$$

Of course, there is no need for Vs to be a solution or to have solutions. For instance, for V with V = [l,g,ue,ue] there is no V' with $V' \leq_{vs} V$ for which $V' \in SOL$ can be true. Along the lines of interpretation sketched above V denotes necessarily the empty set of interval pairs. We say that V does *not contain any solution*.

Definition: The set $NCSOL$
For $V \in VEC$: V does not contain any solution iff $V \in NCSOL$, where:

$$\forall V \in VEC : V \in NCSOL \leftrightarrow \neg(\exists V' \in SOL : V' \leq_{vs} V)$$

3 In Search of a Best Translation

When translating interval statements (using Allen symbols) into vector statements the relevant input is just the set of Allen symbols, not the intervals. Therefore, in the following we focus exactly on this input. Singleton sets are no problem. We use the canonical translation. For richer sets RR ($\subseteq REL$) the strategy will be to use the canonical translation of the elements R of RR and to combine them into the least possible number of vectors for which the retranslation returns exactly RR. Note, that the retranslation comes out with a definite value.

Definition: The retranslation rt

$$rt : Pow(VEC) \rightarrow Pow(REL), \text{ with:}$$
$$rt(VV) = \{R \mid \text{it exists} V \in VV : ct(R) \leq_{vs} V\}$$

Using this, we easily define the translation.

Definition: The translation T
Be $RR \subseteq REL, VV \subseteq VEC$

$$T(RR, VV) \leftrightarrow rt(VV) = RR$$

We stress that T is indeed a relation, not a function. We are interested only in best translations.

Definition: The best translation τ
Be $RR \subseteq REL, VV \subseteq VEC$

$$\tau(RR, VV) \leftrightarrow T(RR, VV) \wedge \neg(\exists VV' : T(RR, VV') \wedge \mid VV' \mid < \mid VV \mid)$$
$$\wedge(\forall VV' : T(RR, VV') \wedge \mid VV' \mid = \mid VV \mid) \rightarrow unequ(VV') \geq unequ(VV))$$

This should be selfexplaining, except for the necessity and meaning of the last conjunct of the definition using *unequ*.

Definition: The inequalites of a set of vectors *unequ*
Be $VV \subseteq VEC$:

$$unequ(VV) = \Sigma_{V \in VV} \Sigma_{i=1}^{4} 1_{ue}(V^i)$$

(Here 1_{ue} is the characteristic function which returns 1 if the argument is *ue* and 0 otherwise.) *unequ* counts the *ue* slots of the vectors of VV. Provided the same cardinality of translations VV and VV', we prefer VV to VV' iff VV counts at most as many inequalities as VV'. We do this since van Beek has shown that the inequalities must trigger an additional subroutine to guarantee the completeness of the Vilain/Kautz-algorithm (cf. [vB90]).

We observe that if we want to combine solution vectors V with $V =$ [A,B,C,D] and V' with $V' =$ [A',B',C',D'] to a vector V'' whose retranslational impact is the same as that of the union of V and V', V'' must contain V and V', i.e. it must hold that $V, V' \leq_{vs} V''$. This means that it must hold: $V^i \sqcup_{ps} V'^i \leq_{ps} V''^i$ for all $i \in \{1, \ldots, 4\}$.

We say that V'' must contain the *space* built up by V and V'.

Definition: The space of a set of vectors
Be $VV \subseteq VEC$:

$$space(VV) = W, \text{ where for all } i \in \{1, \ldots, 4\} : W^i = sup_{ps}\{V^i \mid V \in VV\}$$

(Here sup_{ps} stands for the function which, applied to a subset of $PREL$, returns the least upper bound in the sense of $\leq_{ps}$ of this subset.)

In order to reduce the number of vectors a procedure for computing translations cannot use the *space*-function in an unrestricted way. Take, for instance, the canonical translations [l,l,g,l], [e,l,g,e] of o and id. The space of these vectors, [le,l,g,le], contains, in addition to $ct(o)$ and $ct(id)$, [l,l,g,e] (= $ct(fi)$) and [e,l,g,l] (= $ct(s)$). This is due to the fact that $ct(o)$ and $ct(id)$ differ in more than one place. For this reason, other atomic vectors are contained in the space which are constructed from the alternate use of the differing $PREL$-projections of $ct(o)$ and $ct(id)$. This does no harm if the additional atomic vectors are not solutions, but it does if they are (as in the example).

Of course, this problem disappears if, from the beginning of the translation procedure, we combine pairs of vectors which differ in exactly one place, as suggested by Bittel in [Bit86]. This strategy defines the following reduction procedure:

PROC$_N$ Input: $VV_I \subseteq VEC$
Output: $VV_O \subseteq VEC$, a shorter version of VV_I $(rt(VV_O) = rt(VV_I))$

- $VV \leftarrow VV_I$
- WHILE there are $V, V' \in VV$ with $N(V, V')$

  DO

  BEGIN
  - SELECT $V, V' \in VV$ with $N(V, V')$
  - $VV \leftarrow (VV \bigcup \{space(\{V, V'\})\}) \setminus \{V, V'\}$

  END
- $VV_O \leftarrow VV$

Here, N tests for neighborhood, i.e. for the difference in one place.

Definition: The neighborhood
Be $V, V' \in VEC$:

$$N(V,V') \leftrightarrow \Sigma_{i=1}^{4} 1_{\neq}(V^i, V'^i) = 1$$

($1_{\neq}$ is the characteristic function which tests for the inequality of pairs of symbols).

There are some problems connected to this strategy. We observe that there are cases where a more powerful combination scheme is necessary:

Example1: $VV = \{V_1, V_2, V_3\}$
with $V_1 = ct(fi) =$ [l,l,g,e], $V_2 = ct(o) =$ [l,l,g,l] $V_3 = ct(b) =$ [l,l,l,l].

The test N allows for combining V_1 and V_2, V_2 and V_3, not for combining V_1 and V_3. Choosing the first alternative yields the space [l,l,g,le]. This vector cannot be combined to V_3 under N. The same situation results if we choose the second alternative. We get [l,l,ue,l] which cannot be combined to V_1. However, building the space of VV results in the vector [l,l,ue,le] which, next to V_1, V_2, V_3 contains only the atomic vector [l,l,l,e] which is no solution. So we could correctly reduce VV to one vector.

If we concentrate for a moment on the first alternative yielding [l,l,g,le] as the result of combining V_1 and V_2, we see that we could combine this vector under N with V_3 if, first, we would "pump up" the latter one by the non-solution [l,l,l,e].

Therefore, we define the revised version AN of the neighborhood condition which accepts vectors V, V' as neighbors iff the sum of differences can be restricted to 1 by "pumping up" V or V' respectively by specific atomic vectors which are not solutions. (In our translation procedure, presented in the next section, this "pumping up" will be only necessary for vectors which stem from $\{b, m\}$ or from $\{bi, mi\}$ respectively. Therefore in the following definitions we will restrict ourselves to this specific case).

Definition : Generalized neighborhood
For $V, V' \in VEC$:

$$AN(V,V') \leftrightarrow \exists V*, V'* \in VEC : (V* \in \textit{add-ncsol}(V) \wedge V'* \in \textit{add-ncsol}(V') \wedge N(V*, V'*))$$

Definition : "pumped up" version of vectors (add specific $NCSOL$-vectors)
For $V, V* \in VEC$:

$$\begin{aligned} V* \in \textit{add-ncsol}(V) \quad \leftrightarrow \quad & [V^2 \in \{ge, g, e\} \wedge V*^2 = V^2 \wedge \bigwedge_{i \in \{1,3,4\}} (V*^i \geq_{ps} V^i)] \\ & \vee [V^3 \in \{le, l, e\} \wedge V*^3 = V^3 \wedge \bigwedge_{i \in \{1,2,4\}} (V*^i \geq_{ps} V^i)] \\ & \vee V* = V \end{aligned}$$

We stress, that the *ncsol*- definition guarantees that pumping up does not consist of a blind adding of non-solutions (which could result in the undesired $\leq_{vs}$-inclusion of new solutions which may develop from the crossproduct of suitable projections). We can only generalize (in the sense of $\leq_{ps}$) these projections of the initial vector which are predicted by the value of a particular place by means of the underlying knowledge about endpoints of intervals and point structures. The first disjunct reflects the case where the start of the first interval does not precede the end of the second. From this follows that, with respect to the other projections, solutions require strict succession. The second disjunct reflects the symmetric case for vectors from {b,m}.

PROC$_{AN}$: the same as PROC$_N$, but with the condition $N(V, V')$ replaced by $AN(V, V')$

With PROC$_{AN}$ we get the satisfying result with respect to example 1 if, in the first step, we choose the first alternative (the combination of V_1 and V_2). But choosing the second alternative (yielding [l,l,ue,l]) puts us before another (deeper) problem. We can further weaken the filter for the application of the amalgamating *space*-function. Considering the relevant cases shows that even the weakest version is not sufficient. This version would allow for the application of the *space*-function iff all solutions reached this way already are contained in one of the vectors of the actual VV. This filter version is not sufficient since it cannot rule out misleading reduction steps.

Example2: $\{ct(di), ct(o), ct(s), ct(oi)\}$

Here the choice of combining $ct(di)$ and $ct(o)$ prevents the procedure from doing any further combination (which would be a false one) whereas the choice of combining $ct(di)$ and $ct(oi)$ allows for the second combination of $ct(o)$ and $ct(s)$.

Instead of using a weak filter and correcting misleading combinations by expensive backtracking we have decided to use the filters N and AN and to direct the combining steps by a suited sorting of the input set VV_I.

This sorting is based upon the very relevant neighborhood property N. To begin with, N singles out a specific cover of REL, the cover TT , consisting of sets of pairwise neighboring Allen symbols. Since each of these sets consist of 3 symbols we call them the *triangle sets*.

Definition : The set TT of the triangles of REL

$$TT := \{T_1, T_2, T_3, T_4, T_2', T_3', T_4', T_5\}, \text{ where:}$$

$$\begin{array}{lll} T_1 := \{b, m, o\} & T_2 := \{di, fi, o\} & T_2' := \{di, si, oi\} \\ T_5 := \{oi, mi, bi\} & T_3 := \{si, id, s\} & T_3' := \{fi, id, f\} \\ & T_4 := \{oi, f, d\} & T_4' := \{o, s, d\} \end{array}$$

T_2, T_3, T_4 are called the *horizontal* triangles, T_2', T_3', T_4' are the *vertical* triangles.

The geometrical terminology used here is due to a suited diagrammatical representation of the neighborhood which, for lack of space, we have to omit here.

Nevertheless we continue discriminating specific geometrical subsets of REL the use of which, however, will be made explicit only later.

Definition : The set SS of the small squares of REL

$$SS := \{S_1, S_2, S_3, S_4\}, \text{ where:}$$

$$\begin{array}{ll} S_1 := \{di, fi, si, id\} & S_2 := \{fi, o, id, s\} \\ S_3 := \{si, id, oi, f\} & S_4 := \{id, s, f, d\} \end{array}$$

Definition : The set $\bar{SS}$ of the big squares of REL

$$\bar{SS} := \{\bar{S}_1, \bar{S}_2, \bar{S}_3, \bar{S}_4, \bar{S}_5\}, \text{ where:}$$

$$\begin{array}{ll} \bar{S}_1 := \{di, fi, oi, f\} & \bar{S}_2 := \{fi, o, f, d\} \\ \bar{S}_3 := \{di, o, si, s\} & \bar{S}_4 := \{si, s, oi, d\} \\ \bar{S}_5 := \{di, o, oi, d\} & \end{array}$$

In addition, we partition REL by:

Definition : The full maximal square REL^s and the rest REL^r

$$\mathrm{REL}^s := \{di, fi, o, si, id, s, oi, f, d\}$$

$$\mathrm{REL}^r := \{b, m, mi, bi\}$$

It is easily verified that PROC_N applied to subsets of triangles and squares - except subsets of the full maximal square - always results in a best translation, independent on the choices about combinatorical alternatives. For subsets of triangles the result always consists of just one vector. This is equally true for sets which are complete squares including the full maximal square.

The problem is to sort the input into a suited cover consisting of subsets of triangles and squares that is not misleading with respect to combining steps after this first step of combining the vectors of a particular triangle or square.

There are mainly two cases to be considered. First, the case where a best translation does not require that the same solution is contained in more than one of the resulting vectors and the second case where it does. Example 1 and example 2 illustrate the first case. To the second we will turn later. For the rest of this section we will concentrate on input sets which are subsets of REL^s. We will say something about the general case only in the next section.

Example 2 is based on a subset of REL^s. We call it RR. With respect to RR we have to make sure that for the first combination $ct(di)$ and $ct(oi)$ are taken or $ct(o)$ and $ct(s)$, but not $ct(di)$ and $ct(o)$. This is guaranteed if we choose the cover consisting of the horizontal

triangles T_2, T_3, T_4, not the corresponding vertical cover.

Definition: The partition of REL^s-symbols
Be $RR \subseteq REL^s$:

$$P_h(RR) = \{RR \bigcap T_2, RR \bigcap T_3, RR \bigcap T_4\}$$

$P_h(RR)$ is called the horizontal partition of RR.

$$P_v(RR) = \{RR \bigcap T_2', RR \bigcap T_3', RR \bigcap T_4'\}$$

$P_v(RR)$ is called the vertical partition of RR.

The value of a partition:
We omit here the exact definition. val is used (by the corresponding order $\leq_w$) to prefer one partition alternative to the other, namely the one which needs fewer triangles of the corresponding dimension for a cover than the other. If both partitions need the same number of triangles the number of the inequality symbols contained in the space vectors of the partition elements determines the choice in a rather tricky way. We do not discuss this here, but only observe that the val-criterion decides example 2 in the right way.

Finally, we turn to the second case mentioned above.

Example3: $VV = \{ct(s), ct(si), ct(f), ct(fi), ct(id)\}$

VV, a set of five vectors, can be reduced, independently of the successive choices of vector pairs, by PROC$_N$ to two vectors. The problem here is that the results arrived at this way are not best translations in that one of the returned vectors will contain an inequality ue as can be easily checked. We wanted to avoid inequalities if possible and, here, it is possible ($VV_O = \{$[0,l,g,e], [e,l,g,0]$\}$). The sorting in this case has to avoid the decision between the horizontal cover and the vertical cover. Instead of this it must be sensitive to some exceptional cases among which we should find the constellation $RR = T_2 \bigcup T_2'$. In the translation procedure of the next section we take into account such specific cases by the conditions C1-C7.

4 The algorithm

In addition to the procedures defined in the last section, we need another one which serves as subprocedure of the main procedure defined below.

PROC$^{as}_N$	Input:	$RR \subseteq REL$
	Output:	$VV_O \subseteq VEC$, a translation of RR ($rt(VV_O) = RR$)

- $VV_I \leftarrow \{ct(R) \mid R \in RR\}$
- $VV_O \leftarrow$PROC$_N(VV_I)$

This is the same as PROC$_N$ but with the canonical translation put at the beginning.

The following main procedure is meant to compute best translations for incoming sets of Allen symbols. In a first step of ordering the input along the lines of the "geometrical" partitionings motivated in the last section, we consider nine cases with preconditions C1-C9 which exclude each other but which, taken together, reflect all combinatorical possibilities. In a second step the other procedures are used as sub-routines.

PROC$_\tau$ Input: $RR \subseteq REL$
Output: $VV \subseteq VEC$, a best translation of RR ($rt(VV) = RR$)

- $RR^s \leftarrow RR \bigcap REL^s$
- $RR^r \leftarrow RR \bigcap REL^r$
- CASE1 (C1: $RR^s = \emptyset$)
  - $VV_C \leftarrow \emptyset$
- CASE2 (C2: Exists $i \in \{1, \dots, 4\}, j \in \{1, \dots, 5\}$ with $RR^s = S_i \bigcup \bar{S}_j$)
  - $VV_C \leftarrow \text{PROC}_N(\text{PROC}_N^{as}(S_i) \bigcup \text{PROC}_N^{as}(\bar{S}_j))$
- CASE3 (C3: Exists $i, j \in \{1, \dots, 4\}, i \neq j$ with $RR^s = S_i \bigcup S_j$)
  - $VV_C \leftarrow \text{PROC}_N(\text{PROC}_N^{as}(S_i) \bigcup \text{PROC}_N^{as}(S_j))$
- CASE4 (C4: Exists $i, j \in \{1, \dots, 4\}, i \neq j$ with $RR^s = \bar{S}_i \bigcup \bar{S}_j$)
  - $VV_C \leftarrow \text{PROC}_N(\text{PROC}_N^{as}(\bar{S}_i) \bigcup \text{PROC}_N^{as}(\bar{S}_j))$
- CASE5 (C5: Exists $i \in \{1, \dots, 4\}, T \in \{T_2, T_3, T_4, T_2', T_3', T_4'\}$ with $RR^s = S_i \bigcup T$)
  - $VV_C \leftarrow \text{PROC}_N(\text{PROC}_N^{as}(S_i) \bigcup \text{PROC}_N^{as}(T))$
- CASE6 (C6: Exists $i \in \{1, 2, 3\}$ with $RR^s = T_i \bigcup T_2'$)
  - $VV_C \leftarrow \text{PROC}_N(\text{PROC}_N^{as}(T_i) \bigcup \text{PROC}_N^{as}(T_2'))$
- CASE7 (C7: Exists $i \in \{1, 2, 3\}$ with $RR^s = T_i' \bigcup T_2$)
  - $VV_C \leftarrow \text{PROC}_N(\text{PROC}_N^{as}(T_i') \bigcup \text{PROC}_N^{as}(T_2))$
- CASE8 (C8: $\neg(\bigvee_{i \in \{1, \dots, 7\}} \text{C}_i)$ and $val(P_h(RR^s)) \leq_w val(P_v(RR^s))$)
  - $VV_C \leftarrow \text{PROC}_N(\bigcup_{i=1}^3 \text{PROC}_N^{as}(P_h(RR^s)^i))$
- CASE9 (C9: $\neg(\bigvee_{i \in \{1, \dots, 8\}} \text{C}_i)$)
  - $VV_C \leftarrow \text{PROC}_N(\bigcup_{i=1}^3 \text{PROC}_N^{as}(P_v(RR^s)^i))$
- $VV \leftarrow \text{PROC}_{AN}(VV_C \bigcup \text{PROC}_N^{as}(RR^r))$

Theorem:

a) For all $RR \in REL$: PROC$_\tau$ applied to RR returns a best translation of RR.

b) For all $RR \in REL$: A best translation of RR contains at most 5 vectors.

c) There are $RR \in REL$ a best translation of which contains exactly 5 vectors.

With respect to a), for lack of space we can only sketch the prove. It mainly consists of going through the cases 1)-9) of PROC$_\tau$, checking the described constellations. (One easily sees that geometrically symmetrical constellations can be merged into one checking case). With this we get the proof for the case $RR \in REL^s$. In order to prove the general case, we use this result and, in addition, the following lemma, which is easily checked:

Lemma:

a) $RR \subseteq \{b, m\}, RR \neq \emptyset, V \in VEC, rt(V) \not\subseteq \{b, m\}$.
Then $ct(o) \leq_{vs} space(\{V\} \bigcup \{ct(R) \mid R \in RR\})$.

b) $RR \subseteq \{bi, mi\}, RR \neq \emptyset, V \in VEC, rt(V) \not\subseteq \{bi, mi\}$.
Then $ct(oi) \leq_{vs} space(\{V\} \bigcup \{ct(R) \mid R \in RR\})$.

As a consequence the lemma tells us that vectors V which stem from $\{b, m\}$ on the one hand or from $\{bi, mi\}$ on the other can only be integrated in a set of vectors VV ($rt(VV)$ as in the lemma) - without increasing the cardinality of the set VV and modulo the retranslational impact of $\{V\} \bigcup VV$ - if an element of VV contains $ct(o)$ as a solution or if an element contains $ct(oi)$ respectively. In this case the integration of V can be done by the procedure PROC_{AN} using the generalized neighborhood condition for combinations. Using this result we get the final step of the proof of a).

b) is easily deduced from a). Note that for subsets of triangles and subsets of big or small squares PROC_N always returns exactly one vector.

c) finally shows that the limit stated in b) cannot be improved. Here it suffices to give the following example: $RR := \{b, di, s, f, mi\}$. We easily check that there are no $R, R' \in RR (R \neq R')$ with $rt(space(\{ct(R), ct(R')\})) \subseteq RR$. But in order to arrive at a translation with less than five vectors such a pair of relation symbols is needed.

5 Conclusion

In this paper we have outlined an algorithm which translates descriptions of (underspecified) interval relations expressed in terms of Allen's relation symbols into minimal sets of vectors reflecting the corresponding point-based descriptions. A complete algorithm for computing the transitive closure of a set of interval relations which comprises this translation algorithm and a point-based path consistency algorithm has been implemented and is part of the temporal resolution component of a NL-text understanding system that was developed at the University of Stuttgart.

References

[AL91] Nicholas Asher and Alex Lascarides. Discourse relations and defeasible knowledge. In *Proceedings of ACL*, pages 55–62, 1991.

[All83] James Allen. Maintaining knowledge about temporal intervals. *Comm.ACM*, 26:832–843, 1983.

[Bit86] Oliver Bittel. Prädikatenlogische Beschreibung von Zeitstrukturen mit Anwendung. Master's thesis, Universität Stuttgart, 1986.

[Ebe91] Kurt Eberle. *Ereignisse: Ihre Logik und Ontologie aus textsemantischer Sicht.* PhD thesis, Universität Stuttgart, 1991.

[Haj91] Elzbieta Hajnicz. Another approach to formalizing the point and interval calculi. *International Journal of Man-Machine Studies*, 34:703–716, 1991.

[Her90] Michael Herweg. *Zeitaspekte. Die Bedeutung von Tempus, Aspekt und temporalen Konjunktionen.* PhD thesis, Universität Hamburg, Hamburg, 1990.

[Hin86] Erhard Hinrichs. Temporal anaphora in discourses of english. *Linguistics and Philosophy*, 9(1):63–82, 1986.

[KR83] Hans Kamp and Christian Rohrer. Tense in texts. In Rainer Bäuerle, R. Schwarze, and Arnim von Stechow, editors, *Meaning, Use and Interpretation of Language.* de Gruyter, Berlin, 1983.

[KR85] Hans Kamp and Christian Rohrer. Temporal reference in french. (ms.), IMS, Universität Stuttgart, 1985.

[Nö89] Klaus Nökel. Conves relations between time intervals. In *Proceedings of ÖGAI*, Igls, 1989.

[TM87] S. Thompson and W.C. Mann. Rhetorical structure theory: A framework for the analysis of texts. In *International Pragmatics Association Papers in Pragmatics, Vol.1*, pages 579–105, 1987.

[vB89] Peter van Beek. Approximation algorithms for temporal reasoning. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, pages 1291–1296, 1989.

[vB90] Peter van Beek. Reasoning about qualitative temporal information. In *Proceedings of the Nineth National Conference on Artificial Intelligence (AAAI-90)*, pages 728–734, 1990.

[VK86] Mark Vilain and Henry Kautz. Constraint propagation algorithms for temporal reasoning. In *Proceedings of the Fifth National Conference on Artificial Intelligence (AAAI-86)*, pages 377–382, 1986.

[VKvB89] Mark Vilain, Henry Kautz, and P.G. van Beek. Constraint propagation algorithms for temporal reasoning: a revised report. In D.S.Weld and J.de Kleer, editors, *Readings in Qualitative Reasoning about Physical Systems*, pages 373–381. Morgan Kaufmann, San Mateo, CA, 1989.