Topic/Focus Articulation and Intensional Logic

Tomas VLK

P.O. Box 55
43401 Most
Czechoslovakia

Abstract

A semantic analysis of topic and focus as two parts of tectogrammatical representation by means of transparent intensional logic (TIL) is presented. It is pointed out that two sentences (more precisely, their tectogrammatical representations) differing just in the topic/focus articulation (TFA) denote different propositions, i.e. that TFA has an effect upon the semantic content of the sentence. An informal short description of an algorithm handling the TFA in the translation of tectogrammatical representations into the constructions of TIL is added. The TFA algorithm divides a representation into two parts corresponding to the topic and focus; every part is analyzed (translated) in isolation and then the resulting construction is put together. The TIL construction discussed here reflect the scope of negation and some of the presuppositions observed.

1. Introduction: Transparent intensional logic

One of the current tasks of semantic studies consists in finding a procedure translating the disambiguated linguistic meanings of sentences (see Sgall et al., 1986) into the constructions of intensional logic. The core of such procedure was developed (Vlk, 1987), but a description of this procedure exceeds the scope of the present paper. The aim of this paper is rather to present some ideas used in the algorithm handling the topic/focus articulation within the translation.

Sufficient means for the semantic analysis of natural language are given by Tichy's Transparent intensional logic (TIL). Referring to exact definitions to Tichy (1980) and Materna (1985), we reproduce here only a brief characterization of TIL.

Let $o = \{T, F\}$ be a set of truth-values, let ι be a set of individuals (the universe of discourse) and let ω be a set of possible worlds (the logical space). Then

$B = \{o, \iota, \omega\}$ is an epistemic basis. Then

(i) any member of B is a type over B,
(ii) if $\xi, \zeta_1, \ldots, \zeta_n$ are types over B, then $(\xi\zeta_1 \ldots \zeta_n)$ is a type over B, where $(\xi\zeta_1 \ldots \zeta_n)$ is the set of (total and partial) functions from $\zeta_1 \times \cdots \times \zeta_n$ to ξ.
(iii) the types over B are just those introduced in (i),(ii).

Any member of type ξ is called an object of type ξ, or an ξ-object. An object is an ξ-object for any ξ. For every type a denumerably infinite set of ξ-variables is at our disposal.

The constructions are the ways in which objects can be given. They are defined inductively:

(i) any ξ-object, and also any ξ-variable, is an ξ-construction (called the atomic construction).
(ii) let F be a $(\xi\zeta_1 \cdots \zeta_n)$-construction, X_i a ζ_i-construction for $i=1,..,n$. Then the application $[F\ X_1\ X_2 \ldots X_n]$ of F to $X_1, X_2, \ldots, X_n$ is an ξ-construction.
(iii) let Y be an ξ-construction and $x_1, x_2, \ldots, x_n$ distinct variables of types $\zeta_1, \ldots, \zeta_n$, respectively. Then the abstraction $[\lambda x_1\ x_2 \ldots x_n\ Y]$ of Y on $x_1, x_2, \ldots, x_n$ is a $(\xi\zeta_1 \cdots \zeta_n)$-construction.
(iv) there are no constructions except those defined in (i)-(iii).

Let us characterize some important objects of TIL. For every type ξ we have objects Σ^ξ, Π^ξ of the type $(o(o\xi))$, such that (i) and (ii) hold:

(i) $[\Sigma^\xi\ X]$ = if X is empty class then F else T
(ii) $[\Pi^\xi\ X] = \sim\Sigma^\xi\ \lambda y. \sim[X\ y]$

For every type ξ we have the ξ-singularizer I^ξ of the type $(\xi(o\xi))$, which is defined on single-element ξ-classes only and returns the single element of the respective class. Propositions are objects of the type $(o\omega)$.

The following notation will be used throughout the paper. The outermost parentheses and brackets will be sometimes omitted. Furthermore, a dot will represent a left bracket whose corresponding right bracket is to be imagined as far to the right as is compatible with other pairs of brackets. The notation with an apostrophe will be used in the following meaning:

$$X' = \begin{cases} [X\ w] & \text{if X is of type } (\xi\omega) \text{ for any } \xi \\ X & \text{otherwise} \end{cases}$$

where X is a construction and w is a particular ω-variable.

We write $\exists x.Y$ in place of $\Sigma^\xi\ \lambda x\ Y$ and $\forall x.Y$ in place of $\Pi^\xi\ \lambda x\ Y$, $\imath x.Y$ in place of $[I^\xi\ \lambda x\ Y]$. Logical connectives and identity will be written in the standard way, e.g. $a \,\&\, b$, $a = b$ in place of $[\&\ a\ b]$, $[=_\xi\ a\ b]$, respectively.

2. The topic/focus articulation

The procedure is divided into two parts: into the Basic algorithm handling such phenomena as the scope of quantifiers, several kinds of reference, and so on, and the TFA algorithm handling the topic/focus articulation (TFA). The Basic algorithm is recursively applied to all subtrees of the dependency tree and returns the construction(s) corresponding to the subtree. The TFA algorithm divides the dependency tree into two parts corresponding to the topic and to the focus, respectively; either part is translated by the Basic algorithm, and then the resulting construction is put together.

The topic/focus articulation (TFA) plays a crucial role in analysis of the presupposition, of the scope of negation and also of the so called exhaustive listing (see Sgall, Hajicova, Panevova, 1986, Hajicova 1974, 1984). First, its importance will be shown on an extremely simple 'toy' example; we will then discuss some problems in detail in connection with other examples.

Informally, the topic of a sentence is what the sentence talks about, and the focus is what the sentence says about the topic. A formal definition of topic and focus as two parts of the tectogrammatical representation (TR) see, in details in Sgall et al. (1986, Ch. 3). For the purpose of this paper we only need to know that the boundary between topic and focus is always placed in such a way that there is an item A in the TR such that every item of the TR which is less commun. dynamic than A belongs to the topic (focus).

In all the examples, that part of sentence which belongs to the focus (on the reading discussed) is underlined.

Let us remark that A is a presupposition of B if B entails A and not-B also entails A (Hajičová 1974, 1984). Propositions are understood as (oω)-objects in TIL. Let A, B be (oω)-objects. We say that A is a presupposition of B if for all possible worlds w it holds: that if (B w) is defined then (A w) is true (and if (A w) is not true then (B w) is undefined). The presupposition A determines all possible worlds where B is defined (has a truth-value).

3. About Charles and Mary

Let us consider the examples:

(1) (a) Charles met Mary.

(b) Charles didn't meet Mary.

(2) (a) Charles met Mary.

(b) Charles didn't meet Mary.

(c) Charles didn't meet Mary.

Case (1) brings no problem to the semantic analysis. It corresponds to the "normal reading" of the sentences. We can construct the constructions corresponding to the topic and focus.

Topic = Charles

Focus = λw.λx [Met' x Mary]

where Charles, Mary, x / ι, Met'/(oιι)ω.
Now we can formulate the constructions (they are reduced step by step through the lambda-reductions):

(1')(a) λw [Focus' Topic']
λw [λx(Met' x Mary) Charles]
λw [Met' Charles Mary]

(b) λw. ~ [Focus' Topic']
λw. ~ [λx(Met' x Mary) Charles]
λw. ~ [Met' Charles Mary]

Case (2) is more complicated. Here, the verb belongs to the topic. The sentences (2a-c) can be paraphrased in the following way:

(2) (a') It was Mary who Charles met.
The individual that Charles met is Mary.
(b') It was not Mary who Charles met.
The individual that Charles met is not Mary.
(c') It was Mary who Charles didn't meet.
The individual that Charles didn't meet is Mary.

The sentences (2a,b) have a presupposition that Charles met somebody, while (2c) has a presupposition that there exists somebody who Charles didn't meet. In (2b) the scope of negation is constituted by the focus, the verb itself is not negated. In (2c) only the verb is negated (see Hajičová, 1974, 1984).

In addition, the sentences (2a,c) say that Mary was (in the given context) the only one individual that Charles met, or did not meet. The focus is an exhaustive listing of such individuals, whenever the verb belongs to the topic (Materna, Sgall, 1980, Sgall et al. 1986).

Let us try to find the construction corresponding to case (2a). After the division of the TR into topic and focus we get

Topic1 = λw. λy [Met' Charles y] (oι)ω-construction
Focus1 = Mary ι-construction

Intuitively, this is not what we need. The focus is to assert something about the topic, but here the focus is only a counterpart of an individual. Intuitively, the topic of (2a) is that individual that Charles met and the focus declares about this individual that it is Mary. The constructions must be further modified.

Topic2 = λw. ιy.Met' Charles y
(that individual that Charles met)
Focus2 = λw. λx.x=Mary
(the property of being Mary)

The construction corresponding to (2a) is obtained by application of Focus2 to Topic2.

(2')(a) λw [Focus2' Topic2']
λw. [λx.x=Mary] [ιy.Met' Charles y]
λw. ιy [Met' Charles y] = Mary

Analogically for (2b) we get

(2')(b) λw. ~ [Focus2' Topic2']
λw. ιy [Met' Charles y] ≠ Mary

and for (2c)

Topic1 = λw. λy. ~ [Met' Charles y]
Focus1 = Mary

Topic2 = λw. ιy. ~ [Met' Charles y]
Focus2 = λw. λx.x=Mary

(2')(c) λw [Focus2' Topic2']
λw. [λx.x=Mary] [ιy. ~ [Met' Charles y]]
λw. ιy [~ .Met' Charles y]= Mary

Do these constructions reflect presupposition, negation and exhaustive listing as observed in (2a-c)? The iota-operator (singularizer) is not defined on the empty class, i.e. the propositions (2'a,b) are undefined in those possible worlds where Charles met nobody, and (2'c) is undefined in those possible worlds where Charles met everybody. Also the two scopes of negation corresponding to the contextually bound and non-bound operator of negation are distinguished by (2'b) and (2'c). In (2'a) and (2'c) the equality says that Mary was the only one individual with the given property, i.e. the constructions reflect the exhaustive listing.

Nevertheless, at least two objections to these constructions can be raised:

1. In (2a) Mary is not the single individual in the world that Charles met, but the single one in the given context, the single one from all currently present in the speaker's mind. The construction

λx.Met' Charles x

should be substituted by

λx.[Met' Charles x] & [c x]

where c is a free $(o\iota)$-variable that is to be connected in every situation (context) with a class of individuals activated in the stock of knowledge shared by speaker and hearer. As many problems concerning the context and the stock of shared knowledge are not yet resolved, we will neglect it in this paper.
2. The presupposition of (2'a-c) established by the iota-operator is too strong. It requires that the class under the iota-operator has only one element. In case Charles met more than one individual (2'a) would be undefined, but (2a) would be false in this situation. If Charles met for example Jane and Eve, sentence (2a) would be false, because it was Mary who Charles met. If Charles met Mary and Jane, the sentence (2a) would be (at least partly) false, because the focus part of (2a) is not an exhaustive listing, it does not include Jane. Consider the sentence

(3) Charles met Mary and Jane.

It is not possible to express (3) as a conjunction of two sentences (Charles met Mary and Charles met Jane), because the focus of (3) is an exhaustive listing of individuals that Charles met. The topic of (3) is the same as in (2a). Therefore, the presupposition of both sentences is the same. The topic of both sentences determines a class of individuals that Charles met. The presupposition of the two sentences is that Charles met somebody, i.e. that the class is non-empty. Sentence (2a) says that this non-empty class is equal to the single-element class containing Mary and sentence (3) says that it is equal to the class containing just Mary and Jane. To ensure the presence of the desired presupposition we define the function

$$Ne =_d \lambda c.\, \iota y\, [y=c\ \&\ \exists x\, [y\, x]]$$

$$Ne\ /\ (o\iota)(o\iota) \qquad y,c\ /\ (o\iota) \qquad x\ /\ \iota$$

The function Ne is an identical function defined only on non-empty classes.

In the following construction, "Focus2" corresponds to the focus of (2a), "Focus3" corresponds to the focus of (3) and "Topic" corresponds to the topic of both the sentences.

$$Topic = \lambda w.Ne\ [\lambda x.Met'\ Charles\ x]$$
$$Focus2 = \lambda z.z = \lambda x\ [x=Mary]$$
$$Focus3 = \lambda z.z = \lambda x\ [x=Mary \vee x=Jane]$$

(2'')(a) $\lambda w\ [Focus2'\ Topic']$
$\lambda w.\ [Ne\ \lambda x.Met'\ Charles\ x] = \lambda x.x=Mary$

(3') $\lambda w\ [Focus3'\ Topic']$
$\lambda w.[Ne\ \lambda x.Met'\ Charles\ x] = \lambda x[x=Mary \vee x=Jane]$

These constructions have the same presupposition (that the class in topic is non-empty), they differ only in their foci. The two propositions are not defined in those possible worlds where Charles met nobody.

4. Dividing A-edges

In this section we discuss examples in which the dividing edges between topic and focus are edges having actants as their functors (A-edges). We restrict ourselves in this paper to sentences having just one dividing edge.

(4) Tom sells Jim a car.

(5) Tom sells a car to Jim.

(6) Tom sells Jim a car.

Again, (4) is a neutral (primary) reading. The construction corresponding to (4) is

(4') $\lambda w.\ [Sm\ Car']\ [\lambda x.Sell'\ Tom\ Jim\ x]$
$\lambda w.\ \exists x.\ [Car'\ x]\ \&\ [Sell'\ Tom\ Jim\ x]$

$Sell\ /\ (o\iota\iota\iota)\omega$ $\qquad Sm\ /\ (o(o\iota))(o\iota)$
$Tom,\ Jim,\ x\ /\ \iota$ $\qquad w\ /\ \omega$
$Car\ /\ (o\iota)\omega$

The sentence (5) is analysed similarly as (2a). The verb belongs to the topic, so that (5) triggers a presupposition and contains exhaustive listing like (2a).

$$Topic = \lambda w.Ne\ \lambda x.\ [Sm\ Car']\ [\lambda y.Sell'\ Tom\ x\ y]$$
$$Focus = \lambda z.z=\lambda x.x=Jim$$

(5') $\lambda w\ [Focus'\ Topic']$
$\lambda w.[Ne\ \lambda x.[Sm\ Car'][\lambda y.Sell'\ Tom\ x\ y]] = \lambda x.x=Jim$

The construction (5') ensures the presupposition that Tom sells somebody a car.

Sentence (6) is a little more complicated. Here the topic determines a class of individuals that Tom sells to Jim. Again, the presupposition of (6) is that the class is non-empty. The focus of (6) is an exhaustive listing of such individuals (because the verb belongs to the topic). We feel that (6) is not (fully) true if Tom sells Jim a car and a house or if Tom sells Jim two cars. Focus of (6) says that the class determined by the topic has only one element and this element is a car. We need a function "One" dividing a class X into a set of single-element subclasses of X. It should hold that

$[[One\ X]\ Y]$ iff Y is single-element and $Y \cap X \neq \emptyset$

For every type γ we define

$$One^{\gamma} =_d \lambda p.\lambda y.\ [Ne\ p]\ [\lambda x.y= [\lambda z.z=x]]$$

$$z,\ x\ /\ \gamma \qquad p,\ y\ /\ (o\gamma) \qquad Ne\ /\ (o(o\gamma))(o\gamma)$$

The constructions corresponding to the topic and the focus of (6) are

$Topic = \lambda w\ [Ne\ \lambda x.Sell'\ Tom\ Jim\ x]$
$Focus = \lambda w\ [One\ Car']$

and to sentence (6)

(6') $\lambda w.\ [[One\ Car']\ [Ne\ \lambda x.Sell'\ Tom\ Jim\ x]]$

The proposition constructed by (6') is defined only in those possible worlds where Tom sells something to Jim (because only in such possible worlds function Ne is defined), and it is true in those possible worlds where Tom sells Jim just one thing and this thing is a car. An alternative analysis of (6) can be found in Materna et. al. (1987) and the two analyses are compared in Vlk (1987).

The negative counterparts of (5),(6) again have two readings each, in accordance with the contextual boundness of the operator of negation (see Hajičová, 1974, 1984). Let us take the negative counterparts of (5) for illustration.

(5a) Tom does not sell a car to Jim.

(5b) Tom does not sell a car to Jim.

In (6a) the focus of (6) is negated, the topic is true (case).

Topic4 = λw.Ho λx. [Ho Car'] [λy.Sell' You x y]
Focus4 = λx.∀y[x=Jim]

(6a') λw [Focus4' Topic4']
λw.[Ho λx.[Ho Car'][λy.Sell' You x y]] → λx.x=Jim

The sentence (6a') conveys the same presupposition as (6'), it is defined only in those possible worlds where You sells a car to somebody, and it asserts that the class containing just Jim is not an exhaustive listing of individuals whom You sells a car to. (You may sell a car to John or to Jim, John and Mary, etc.)

In the following example, where the negation is contextually bound, it is presupposed that there exists a non-empty class of persons to whom You does not sell a car, and it is asserted that this class has Jim as its single element.

Topic2b = λw .Ho λx. [Ho Car'] [λy.¬[Sell' You x y]]
Focus2 = λx.x=Jim

(6b') λw [Focus2' Topic2b']
λw.[Ho λx.[Ho Car'][λy.¬[Sell' You x y]] → λx.x=Jim

5. Dividing R-edges

In this section we discuss examples in which the dividing edges between topic and focus are edges having 'Cause', 'Aim' or 'Condition' as their functors (R-edges). They are analyzed rather as relation-in-intension between propositions (Materna, Sgall, 1980).

Let us take the following examples

(7) Because Mary was ill, Charles came.

(8) Because Mary was ill, Charles didn't come.

(9) Charles came because Mary was ill.

(10) Charles didn't come because Mary was ill.

(11) Charles didn't come because Mary was ill.

Let us note that (11) includes a contextually non-bound operator of negation and can be continued e.g. by "..., but (came) rather since he wanted to meet John there."

Each of the sentences (7-11) has one dividing edge with the functor "Cause". All sentences have "come" as their main verb (the root of the dependency tree). In these examples "Cause" is the relation-in-intension between two propositions. The first proposition is in the topic and the second one is in the focus of the sentence. The proposition in the topic is always a presupposition of the sentence. Thus (7) and (8) have the presupposition that Mary was ill, (9) and (11) presuppose that Charles came and (10) presupposes that Charles didn't come. To observe such presuppositions we need a function Tr that would be a "filter" on a "class" for true propositions. It should hold for every possible world W and proposition P that

$$[\mathrm{Tr}(P)](W) = \begin{cases} P & \text{if } P(W) = T \quad (P \text{ is true in } W) \\ \text{undefined} & \text{otherwise} \end{cases}$$

Tr / ((oω)(oω))ω   P / (oω)   W / ω

This requirement is fulfilled by the definition:

Tr = $_{d}$ λw λp (ιq.q=p & [q w])

Tr / ((oω)(oω))ω   p,q / (oω)   w / ω

Let us look at the scope of negation. In (8) only the second proposition (in focus) is negated, not the causal relationship between topic and focus. In (10) only the first proposition is negated and again "Cause" is not in the scope of negation. On the other hand, in (11), where the main verb is contextually bound and the operator of negation is non-bound, it is just the causal relationship that is negated.

If Caus / (o(oω)(oω))ω realizes the relationship "Cause", so that λw [Caus' P1 P2] means that P2 causes P1, the constructions corresponding to (7-11) are

(7') λw.Caus'[λw.Come' Charles][Tr' λw.Ill' Mary]

(8') λw.Caus'[λw.¬[Come' Charles]][Tr' λw.Ill' Mary]

(9') λw.Caus'[Tr' λw.Come' Charles][λw.Ill' Mary]

(10') λw.Caus'[Tr' λw.¬[Come' Charles]][λw.Ill' Mary]

(11') λw.¬[Caus'[Tr' λw.Come' Charles][λw.Ill' Mary]]

From (11') being true it follows (i) that Charles came, (ii) that this was not caused by Mary's illness, and (iii) neither that Mary was, nor that she was not ill. In the positive counterpart (9') the fact that Mary was ill is entailed. In other words, (iii) is an allegation of (9), in the terminology of Hajičová (1974, 1984).

The constructions (7'-11') construct pairwise different propositions. To demonstrate this, let us compare (8') and (10'). Consider two possible worlds W1 and W2:

W1:	Charles came.	Mary was ill.
W2:	Charles didn't come.	Mary was not ill.

In W1, (8') is false and (10') is undefined. In W2, (8') is undefined and (10') is false.

It can be seen that a dividing R-edge is in the scope of negation only when the main verb is contextually bound and the related operator of negation is non-bound.

6. TFA Algorithm

In this section we describe an algorithm handling the TFA. For the sake of simplicity, only examples with one dividing edge between the topic and focus were presented. A general case (with more than one dividing edge) requires the introduction of additional functions (see Vlk, 1987). Here we present a simplified version of the algorithm, which reflect only the phenomena discussed in this paper.

First we summarize the functions defined in the previous sections.

Tr = $_{d}$ λw.λp (ιq.q=p & [q w])
Ho = $_{d}$ λp.ιq [q=p & ∀ x$_1$...x$_n$ [q x$_1$ ... x$_n$]]
One = $_{d}$ λp.λy. [Ho p] [λz.y=λx.x=z]

Tr / ((oω)(oω))ω
Ho / (oβ$_1$...β$_n$)(oβ$_1$...β$_n$)
One / (o(oγ))(oγ)

The following functions are used in the description:

```
CB        : DepTree -> Bool
NB        : DepTree -> Bool
NBNeg     : DepTree -> Bool
Tree      : Edge    -> DepTree
Fun       : Edge    -> Functor
H         : Functor -> Construction
R-Edge    : Edge    -> Bool
A-Edge    : Edge    -> Bool
DivEdge   : DepTree -> Edge
DelEdge   : DepTree Edge -> DepTree
PutVar    : DepTree Edge -> DepTree
Translate: DepTree -> Construction
GetTyp    : Contruction -> Type
```

The meanings of the functions are as follows:

CB(dt) returns true iff the root of dt is contextually bound. NB(dt) returns true iff CB(dt) returns false (NB(dt) = ~CB(dt)). NBNeg(dt) returns true iff the contextually non-bound operator of negation is connected with the root of dt (contextually bound operator of negation is handled by the Basic algorithm).

Tree(e) returns the dependency tree suspended on edge e. Fun(e) returns the functor of edge e. H(f) returns the object of TIL realizing relationship f ('Cause','Aim'). R-Edge(e) returns true iff e is an R-Edge. A-Edge(e) returns true iff e is an A-Edge. DivEdge(dt) returns the dividing edge between the topic and the focus of dt.

Functions DelEdge and PutVar realize dividing of the dependency tree. DelEdge(dt,e) returns dependency tree dt' without edge e (edge e is removed from dt). PutVar(dt,e) replaces the tree suspended on edge e in tr by a variable and returns the resulting dependency tree.

Translate(dt) returns the construction of TIL corresponding to dt to which dt is translated by the Basic algorithm. GetTyp(c) returns the type of construction c.

Now we can describe the following procedures:

```
TFA     - the main procedure (function)
FA      - verb in the focus, dividing A-edge
TA      - verb in the topic, dividing A-edge
FR      - verb in the focus, dividing R-edge
TR      - verb in the topic, dividing R-edge
```

```
TFA : DepTree -> Construction

TFA (dt) =
    let  e = DivEdge (dt)  in
     ( A-Edge(e) & NB(dt) ->  FA(dt),
       A-Edge(e) & CB(dt) ->  TA(dt),
       R-Edge(e) & NB(dt) ->  FR(dt),
       R-Edge(e) & CB(dt) ->  TA(dt)
     );
```

If the dividing edge is an A-edge and the verb belongs to the focus the tree is handled by function FA. The tree suspended on the dividing edge is replaced by a variable, the topic and focus are translated separately and the resulting construction is put together. F is the construction corresponding to the focus and T is the construction corresponding to the topic.

```
FA : DepTree -> Construction

FA (dt) =
    let  e = DivEdge (dt),
         F = Translate (PutVar (dt,e)),
         T = Translate (Tree(e))
    in
      if NBNeg(dt) then [ λv.~ [F' T'] ]
                   else [ λv [F' T']]
```

If the dividing edge is an A-edge and the verb belongs to the topic the tree is handled by function TA. The tree is divided in the same manner as in FA. The resulting construction is more complicated than in TA because it has to reflect presuppositions and exhaustive listing.

```
TA : DepTree -> Construction

TA (dt) =
    let  e = DivEdge (dt),
         T = Translate (PutVar (dt,e)),
         F = Translate (Tree(e))
    in
    let
         Y = (if GetTyp(F')=GetTyp(T')
                          then [ [One F'][No T'] ]
                          else [ [λy.y=F']=[No T'] ] )
    in
       if NBNEg(dt) then    [ λv.~ Y ]
                    else    [ λv. Y  ]  ;
```

If the dividing edge is an R-edge and the verb belongs to the focus the tree is translated by function FR. Here the dividing edge is removed from the tree and the functor of the dividing edge determines a relationship between the topic and focus. The proposition in the focus is presupposed, the presupposition is ensured by function Tr. The relationship between the topic and the focus is not within the scope of negation.

```
FR : DepTree -> Construction

FR (dt) =
    let  e  = DivEdge (dt) ,
         F  = Translate (DelEdge (dt,e)),
         T  = Translate (Tree(e)),
         P  = H(Fun(e))
    in
      if NBNeg(dt) then [ λv.[P' [λv.~ F'] [Tr' T]]]
                   else [ λv [P' F [Tr' T]]]  ;
```

If the dividing edge is an R-edge and the verb belongs to the topic the tree is translated by function TR. The tree is divided in the same manner as in FR. A relationship between the topic and focus is within the scope of negation here.

```
TR : DepTree -> Construction

TR (dt) =
    let  e  = DivEdge (dt) ,
         T  = Translate (DelEdge (dt,e)),
         F  = Translate (Tree(e)),
         P  = H(Fun(e))
    in
      if NBNeg(dt) then [ λv.~ [P' [Tr' T] F]]
                   else [ λv [P' [Tr' T] F]] ;
```

Although many problems are open, it is seen that the topic/focus articulation has an effect on the semantic content of the sentence and, therefore, it can be analyzed by means of formal semantics.

Acknowledgement.
The author wishes to thank Prof. Petr Sgall for his support of this work and valuable discussions.

References

Hajicova, E. (1974):
Meaning, Presupposition and Allegation. Philologica Pragensia 17, 18-25.

Hajicova, E. (1984):
Presupposition and allegation revisited. Journal of Pragmatics 8, 155-167.

Materna, P. (1985):
"Linguistic Constructions" in the Transparent Intensional Logic. The Prague Bulletin of Mathematical Linguistics 43, 5-24.

Materna, P. - Hajicova, E. - Sgall, P. (1987):
Redundant Answers and Topic/Focus Articulation. Linguistics and Philosophy 10, 101-113.

Materna, P. - Sgall, P. (1980):
Functional Sentence Perspective, the Question Test and Intensional Semantics, SMIL 1-2, 141-160.

Materna, P. - Sgall, P. (1984):
Optional Participants in Semantic Interpretations (Arity of Predicates and Case Frames of Verbs). In: [Sgall 1984], 51-62.

Platek, M. - Sgall, J. - Sgall, P. (1984):
A Dependency Base for Linguistic Descriptions. In: [Sgall 1984], 63-97.

Sgall, P., ed. (1984):
Contributions to Functional Syntax, Semantics and Language Comprehension. Prague, Academia 1984.

Sgall, P. - Hajicova , E. - Panevova , J. (1986):
The Meaning of the Sentence in Its Semantic and Pragmatic Aspects. Academia, Prague 1986.

Tichy, P. (1980):
The logic of temporal discourse. Linguistics and Philosophy 3, 343-369.

Vlk, T. (1987):
Towards a transduction of underlying structures into intensional logic. The Prague Bulletin of Mathematical Linguistics. (Forthcoming)