Common Denominators and Default Unification[*]

Martin van den Berg and Hub Prüst
Department of Computational Linguistics, Faculty of Arts,
University of Amsterdam

1 Introduction

For a correct analysis of natural language discourse the notion *syntactic/semantic parallelism*, expressing the sharing of structure by successive clauses, is an important device. It plays an important role in establishing semantic coherence [Polanyi85], structuring the discourse [Scha&Polanyi88], and resolving VP-anaphora [Prüst&Scha90] and Gapping (in [Steedman90] some kind of process like ours is assumed).

In the framework we are developing, which is a unification grammar for discourse, the sharing of syntactic/semantic structure is a basic notion. Crucial for the incorporation of a new sentence (in general: of a discourse unit) in the preceding discourse is a method to establish what the structure they share is. In particular, suppose we want to add a sentence S to a discourse C. What we are looking for is a mechanism to determine the maximal part of the preceding structure C that is compatible with the possibly incomplete structure resulting from this new sentence S. This maximal compatible part can then be combined with S to form what we call the context dependent meaning of that sentence. In the case that S is incomplete, for example when it contains a VP-anaphor or exhibits gapping, this process can add new information. We call this compatible part the *common denominator*[1] of C and S.

The formal problem addressed in this paper is the definition of the notion of the common denominator $A¢B$ of two expressions A and B, such that $A¢B$

[*]We would like to thank Mary Dalrymple, for pointing out a mistake in an earlier version of this paper and Gosse Bouma for his constructive criticism. Also thanks to ghost-coauthor Remko Scha.

[1]In [Prüst&Scha90] this was called the *common ground*, but we changed the name to prevent confusion with other uses of that term.

is that part of A that is still compatible with B. The dual notion *default unification*, $A \gg B$, is defined as B unified with as much of A as possible. Default unification is a notion that is closely related to the notions default, or non-monotonic, inheritance and priority union.

Below we show that common denominators and default unification can be defined on *any* structure that allows for generalization and unification and that they can be defined in terms of these notions.

2 The Common Denominator

It will not come as a surprise that the two sentence (1) and (2) have something in common, and that what they have in common is sentence (3). We call (3) the *(Most specific) Common Denominator* of (1) and (2).

(1) *John eats lasagna.*

(2) *Bill eats lasagna.*

(3) *X eats lasagna.*

Because they have something in common they contain redundant information. Sentence (2) could be paraphrased as: *And the same holds for Bill.* That is of course exactly what we do in ordinary language:

(4) *John eats lasagna.*

(5) *Bill does too.*

The question is: how do we manage to interpret (5) as (2)? At first sight, it might be suggested that we can do this because (5) shares part of its structure with (4). But that is not true, because it is exactly the part which has been deleted which is the part that would have been shared had it been there — it is not there to help determine it! (if it were there we would not need it). We are not looking for *shared*, but for *deleted* information.

So how do we determine the structure that these two sentences share? The description above gives a good suggestion how to go about finding it. If the second sentence was constructed by deleting what it has in common with the first sentence, then if we look what parts of the first sentence *conflict* with the second and leave these parts out, then the rest is a good candidate for that common part. In words we can describe the process we are looking for as follows:

Find a sentence S, such that S is more specific than the incomplete (= too general) sentence (5) and such that what S and the preceding sentence (4) have in common is exactly that in which S differs from (5). This may seem dangerously close to a circular definition, but it is on the safe side of the divide, as was argued in [Prüst&Scha90] and is shown below.

This process, where an incomplete object (sentence 5) is unified with as much of the default information (sentence 4) as is possible, we call *default unification*, because, as we will see below, it is related to the notion of that same name in [Bouma90a, Bouma90b].

The above suggests that the process works on sentences. In fact, this is somewhat of a simplification. The process actually works on the propositions expressed by those sentences, in [Prüst&Scha90, Prüst,Berg&Scha] we identify these with their derivation trees, coded as unreduced lambda-terms[2]. For the moment we will simplify this even further and consider PROLOG like expressions.

3 Constructing the Common Denominator

The process of calculating the common denominator, the subject of this paper, is something that has the character of unification combined with some aspects of generalization to give it some spice.

For the moment, assume that VP-anaphors are anonymous variables (borrowing some PROLOG terminology) — holes in the formula that have to be filled. Then the example sentences above translate as

(6) *eat - Lasagna(john)*

(7) ___*(bill)*

The common denominator of these two is

(8) *eat-Lasagna(*___*)*

it seems that to get this result we sometimes we have to unify: *eat-lasagna* with ___ results in *eat-lasagna* and sometimes have to generalize: *john* with *bill* gives ___. How can we ever resolve this?

In fact, what happens is the following. We look at the two expressions term-wise. If two terms unify we take the unification. If they do not unify we take the generalization.

[2] In the end we might want to use some good theory of propositions. One candidate for this could be property theory [Chierchia87].

The structure formed by the logical representations of the sentences (including those containing holes) is that of a partial order[3]. In terms of this order we get the following definition.

A sentence ϕ is a specification of a sentence ψ, written as $\phi \preceq \psi$, if there is a substitution σ for some of the holes in ψ such that $\sigma(\psi) = \phi$. In that case we also say that ψ is a generalization of ϕ.

a simple example:

(9) *eat-Lasagna(john)* $\preceq$ *eat-Lasagna(__)*,

where *john* can be substituted for the hole.

A slightly less simple example is given by the sentence *Every man loves girls who sing.* Any of (10) to (13) could follow it (if we interpret the pronoun *them* as a hole).

(10) *Every boy loves girls who sing*

(11) *Every boy loves girls who do [too]*

(12) *Every boy loves them [too]*

(13) *Every boy does [too]*

Now this is a very particular example. In general, we have to take unification variables into account. This causes the order to be a pre-order[3]. Another problem is caused by the fact that a variable might occur more than once, and that the value substituted for that variable has to be the same for all occurrences of it. The way to go about this is to define the ordering using the variable substitutions explicitly. The general definition of the pre-order becomes:

A sentence ϕ is a specification under σ of a sentence ψ, written as $\phi \preceq_\sigma \psi$, iff σ substitutes values for the variables in ϕ such that $\sigma(\phi) = \psi$. A sentences ϕ is a specification of a sentence ψ, written as $\phi \preceq \psi$, iff there is such σ.

As before we also say that ψ is a *generalization* of ϕ.

[3]This order is the well-known subsumption relation between the terms of the unification structure. In general, a unification structure has terms with unification variables. In that case the ordering is a pre-order, because terms that only differ from each other by some variable renaming are different but equivalent under the ordering. Here, because of the absence of variables, it is a partial order.

In terms of the ordering we can define unification and generalization quite easily. *The most specific generalization (msg) of ϕ and ψ is that generalization ρ of both ϕ and ψ such that every generalization ρ of both ϕ and ψ is more general than χ. If we write $\phi \rightarrow \chi$ for $\phi \preceq \chi$, we can represent this in a diagram as (fig.1).*

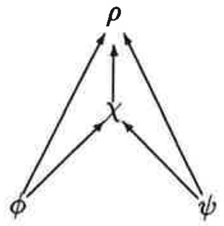

fig.1

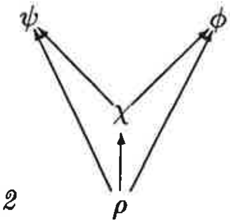

fig.2

The dual notion of this is the most general unifier (mgu) of ϕ and ψ is that specification χ of both ϕ and ψ such that every specification ρ of both ϕ and ψ is more specific than χ. Using the same arrows, we can also give a picture for this (fig.2).

We write $X = \phi \sqcup \psi$ for the *msg* of ϕ and ψ, and $\phi \sqcap \psi$ for their *mgu*, and for convenience sometimes speak of the generalization and the unification. One thing that complicates things a bit is that, given the definitions, there is always an *msg*, but not always an *mgu*. If ϕ and ψ do not have a unifier, we sometimes write $\phi \sqcap \psi = \perp$ but this is only a shorthand, $\perp$ is *not* an actual object. Other formalisms do add $\perp$ to the structure, but this is at the cost of always having to mention it as an exception, so we prefer to keep it outside the structure.

Another complication is that neither *msg* nor *mgu* (if it exists) need to be unique. The fact that we can always exchange one variable for another is easy to remedy; just divide out the variables by taking equivalence classes under the order. However, in other cases this might not always be possible[4]. In such cases we can only define *the set of msg's* and *the set of mgu's*. The whole argument of this paper can be reformulated in terms of some object being one of the msg's of some other objects, but this only complicates matters without adding anything essentially new, so we leave that as an exercise for those who need it and will only sometimes add the provision "(up to equivalence)" to remind the reader of possible problems.

In terms of *msg* and *mgu* we can now easily define the notion of most specific common denominator (*mscd*). Here is a definition that works for

[4] cf. footnote 6.

our purposes, and almost works in the general case.

The mscd of ϕ relative to ψ is that generalization χ of ϕ that unifies with ψ such that every generalization γ of ϕ that also unifies with ψ is a generalization of χ.

We will write $\phi¢\psi$ for this *mscd* (cf. *fig.3*).

Now this might have been all there is to this definition, if not for a slight complication. Even if the *msg* and the *mgu* (if it exist) are always unique (upto equivalence), the *mscd* defined in this way does not have to be. We already saw that in our simple example of discourse parallelism the *mscd* is always unique, because variables do not play any role in that framework. If variables do occur in a formula they can cause problems if they occur more than three times. The following is an example of this:

Let X, Y, Z, P, Q denote variables, and a, b constants. Both XXY and YXX satisfy the conditions given above for being the *mscd* of XXX relative to aPb, although they are not equivalent under renaming of variables. They are both generalizations of XXX, unify with aPb and there does not exist a more specific object satisfying these conditions.

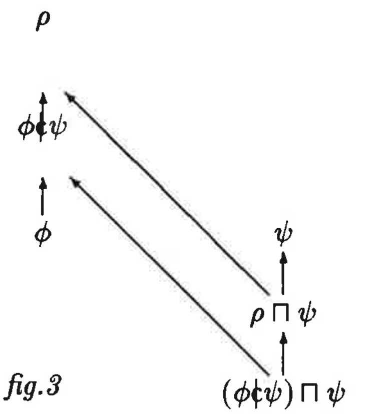

fig.3

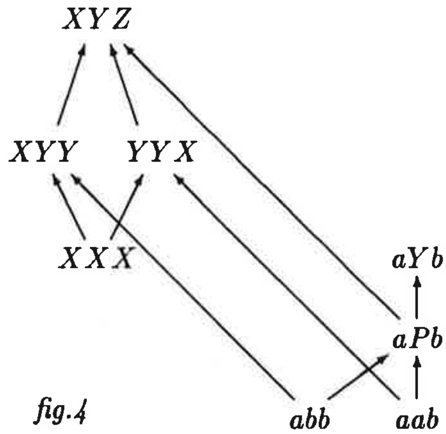

fig.4

In the light of this you have two choices, either you live with it, and consider it an actual ambiguity or you try to do away with it. There is something to say for both positions, but, at least for the moment, we choose the latter. It seems to reflect our intuitions best. But given that these intuitions are formed by an example that does not need variables anyway, we might not be the best judges in this case[5].

[5]following a suggestion by *Dalrymple et.al.*, we might try to see the ambiguity in VP-

Our stand-point might be formulated like this. If a connection between positions in a term fails, a connection that is expressed by the fact that the same variable occurs on the connected positions, the complete connection should be thrown out, and not only part of it. The simplest solution is to just take the *msg* of all the candidates of the previous method. In the case of the example this would result in the not unreasonable XYZ (*fig. 4*).

We can now give the complete "algorithm" of *mscd* calculation:

a) *find the set* $\hat{C}(\phi, \psi)$ *of objects* χ, *such that* χ *is more general than* ϕ, *and* χ *unifies with* ψ,

b) *define* $C(\phi, \psi)$ *to be the set of those elements in* $\hat{C}(\phi, \psi)$ *such that no element in* $\hat{C}(\phi, \psi)$ *is more specific than it,*

c) *take the msg of* $C(\phi, \psi)$.

Or in one big formula:

(14) $\phi \cent \psi = msg\{\chi . \phi \preceq \chi \& \psi \sqcap \chi \neq \bot \& \neg \exists \rho . (\phi \preceq \rho \& \psi \sqcap \rho \neq \bot \& \rho \prec \chi)\}$

Remarks

- The definition of the *mscd* is an all or nothing definition. We really throw away everything of a connection that conflicts on some of its positions, and not only the connection between the conflicting positions. for example, suppose we have $XXXX$ and $aPQb$, this gives $C(XXXX, aPQb) = \{YXXX, XXXY, XXYY\}$, and this results in: $XXXX \cent aPQb = XYZP$. No connection between the positions is preserved.

- The result of unifying the *mscd* of $\phi \cent \psi$ with ψ is the *default unification* of ψ with ϕ, $\phi \gg \psi = (\phi \cent \psi) \sqcap \psi$. The notions common denominator and default unification are each others dual.

- Another way of defining unification is to say that ϕ and ψ unify if there is one substitution σ such that $\sigma(\phi) = \sigma(\psi)$. The formulation we use here is more convenient because it makes sure that the variables in ϕ

anaphora like *John kissed his wife and so did Bill*, which is ambiguous between a reading where bill kissed his own wife and one where he kissed john's wife, as a case where the *mscd* is ambiguous. However, we will have to look at this in more detail before we can say anything detailed about this.

and ψ are always different, even if they are written as the same letter. In other words, variables are made local to the formula, which is as we think it should be.

- The above uses only term-unification. No accidental identifications between terms exist. There is nothing sacred about this. The above reasoning holds for any pre-order. If the *msg* is unique, so is the *mscd*. If it the *msg* is not unique, then neither is the *mscd*. If the *mgu* is not unique, you have to be careful and check whether what you get is what you want[6].

4 Two applications of the Common Denominator

4.1 anaphora

We already discussed VP-anaphora above. NP-anaphora is just as simple and the current framework sheds some light on the distinction between topic and focus. Look at the following examples (capitals indicate stress):

(15) *John is having dinner. He has veal with potatoes.*

(16) *John is talking to Bill over dinner. He pours him another wine.*

(17) *John is talking to BILL over dinner. HE is the one with the answers.*

Without wanting to claim that the following is a complete theory of pronoun resolution, we might hypothesize the following. The main influence on deciding what refers to what is the syntactic/semantic relations between the sentences. What we do is compare the derivation trees of sentences and try to map them onto each other.

We like to represent these trees in terms of lambda-expressions, but it should be stressed that these are syntactic lambda terms: $[\lambda x(f(x))](g) \neq f(g)$. It is just a convenient one-dimensional way of writing the derivational history. Trees are now compared, taken the common denominator of, and default unified, by doing term-unification on the lambda-terms in

[6]For an example of a unification structure that has a non-unique unification take the following. Suppose the language contains a relation, say something like a conjunction $\wedge$, that is associative for the unification operator. In our way of formulating this corresponds to the property of the ordering: $((\phi\wedge\psi)\wedge\chi) \preceq (\psi\wedge(\phi\wedge\chi))$. Then $((\phi\wedge\psi)\wedge\chi)\sqcap(X\wedge Y)$ results in $X = (\phi\wedge\psi)$ and $Y = \chi$, whereas $(\phi\wedge(\psi\wedge\chi))\sqcap(X\wedge Y)$ results in $X = \phi$ and $Y = (\psi\wedge\chi)$, which is not the same, although $(\phi\wedge\psi)\wedge\chi)$ and $(\phi\wedge\psi)\wedge\chi)$ are equivalent by associativity. The unifier of two terms is not always unique.

the standard way, interpreting pronouns as "holes" in the formula. Example (15) translates as $[\lambda x.(\textit{have-dinner}(x))](\textit{john})$ for the first sentence and $[\lambda z.\textit{has-veal}(z)](\_\_)$ for the second. Calculating the common denominator of these results in $\_\_(\textit{john})$ and, finally, default unifying this with the tree of the second sentence gives the end result $[\lambda z.\textit{has-veal}(z)](john)$. The advantage of coding trees as non-reduced lambda-expressions is obvious: their reduction gives the meaning of that sentence.

In the same way we can describe what goes on (16). Sentence one translates as: $[\lambda x \lambda y.(talkOverDinner(y,x))](\textit{bill})(\textit{john})$ and the second as $[\lambda p \lambda q.pourWine(q,p)](\_\_)(\_\_)$. Calculating the common denominator gives $\_\_(\textit{bill})(\textit{john})$, resulting in a context dependent meaning for the second sentence of the form: $pourWine(\textit{bill})(\textit{john})$

Example (17) can be used to show how we might try to deal with topic/focus in our framework. The hypothesis we would like to put forward is, that the topic/focus structure is articulated in the structure of the derivation of the sentences. Stressing of words is seen as a essential part of syntax contributing to the derivation. Instead of the derivation of (16) given above, the interpretation of the stressed (17) is the formula

(18) $[\lambda p \lambda q[\lambda x \lambda y(talkoverDinner(y,x))](p,q)](john)(\mathbf{\textit{bill}})$

You might say that stress pushes the stressed term to the *outside*. This is not so far fetched if we compare this with a theory of topic/focus that is quite well-known, namely that of Mats Rooth [Rooth85]. In that framework terms that are in focus in some sentence are first marked with a special index during the syntactic analysis that constructs the derivation tree of that sentence. Then the topic is constructed by replacing the focused terms by variables. If there is only one stressed term in the sentence, this results in the topic being the a set of alternative values that could have occurred in place of the actual value (which was stressed on the location of the variable). In general, when there are more variables, the result is a lambda term. Our last example is formalized as something like:

the topic: $[\lambda p \lambda q(talkOverDinner(q,p))]$

the meaning: $[talkOverDinner(john, \mathbf{\textit{bill}})]$

4.2 default unification of feature structures

Studying implementations of unification grammars with feature structures, in particular HPSG, Gosse Bouma [Bouma90a, Bouma90b] developed a notion of default unification not unlike ours. This is used to implement the notion of an almost-always-applying feature structure. We can then add default structures to the unification grammar, which function as default rules, to be applied if nothing blocks them. Roughly speaking, this functions as follows. As long as a default "rule" being considered can be unified with the structure under construction by the grammar it is unified with it, but if it is incompatible with the structure as constructed thus far the default structure is ignored. This notion is closely related to the notion "adding as much as possible" described in the previous section and it is no coincidence that this was also called default unification.

In order to examine in more detail how we might implement default unification for feature structures in the common denominator formalism we first have to find a way of formalizing feature structures so that the terminology of the previous chapters applies. To do this, we define a flat representation of feature structures, define the ordering of the structures in terms of these representations, and then calculate the default unification that follows from this ordering. We will simplify things somewhat to keep the axioms manageable, but we don't think that anything essential is left out.

4.3 Feature structures

The feature structures used in most grammatical theories like HPSG are labeled non-cyclical re-entrant finite atomic graphs. This mouthful means the following: there is a set of atoms $\mathcal{A}$, a set of labels, called features, $\mathcal{F}$, and a set of feature structures defined in terms of these as follows:

(19) a. *If a is an atom, then a is a feature structure.*

b. *If f is a feature and ϕ a feature structure, then $\xrightarrow{f} \phi$ is a feature structure.*

c. *If f, g are features and ϕ, ψ feature structures, then $\begin{array}{l} {}^{f}\nearrow \phi \\ {}_{g}\searrow \psi \end{array}$ is a feature structure.*

It might be thought to we forgot the re-entrancy that these structures also have. However, we will not distinguish between two (or more) sequences of features, $\vec{f_1}$ =)$f_1^1 : f_1^2 : \ldots : f_1^{m_1}$) and $\vec{f_n} = (f_2^1 : f_2^2 : \ldots : f_2^{m_2})$, being re-entrant or being non-re- entrant but having the same value. As far as we are concerned the following equality holds:

$$\begin{array}{c} f_1 \nearrow \\ f_n \searrow \end{array} \vdots \begin{array}{c} \searrow \\ \nearrow \end{array} \phi = \begin{array}{l} f_1 \nearrow \phi \\ f_n \searrow \phi \end{array}$$

The reason we think we can identify these is that the grammatical properties expressed by a feature structure are expressed by the paths from the root to the leaves through the structure. Then any structure is equivalent to the fan-form structure that you get by taking all of these paths and combining them together in a graph that is only connected at its root.

We will also make a notational simplification. We will code atoms as special features ending in one fixed atomic value $\top$. This gives an equivalent class of structures (with every atom a replaced by a sub-feature structure $\xrightarrow{a} \top$) and does away with the need for separate axioms for atoms.

We also want to be able to express that certain sequences of features having the same value, without having to specify explicitly the value that they share. This is where we do use re-entrancy in a non-trivial way. Say that we have sequences $\vec{f_1} = (f_1^1 : f_1^2 : \ldots : f_1^{m_1}), \ldots, \vec{f_n} = (f_n^1 : f_n^2 : \ldots : f_n^{m_n})$ we have

(20) *d. If $\vec{f_1}, \ldots, \vec{f_n}$ are sequences of features, then* $\begin{array}{l} \vec{f_1} \nearrow \searrow \\ \quad \vdots \quad \cdot \\ \vec{f_n} \searrow \nearrow \end{array}$ *is also a feature structure.*

Note that this is essentially different from the earlier structures in that it is an incomplete structure. It lacks the actual value that the features share. If we fill this value in, we get a complete structure back.

The set of structures as definied above is in fact a bit to large. Not all values can occur after a given feature. It does not make sense to say that the value for the feature **number** is **transitive**, or that the **gender** is **singular**. For any feature f in a given feature algebra we are given a set $adm(f)$ of admissible features. Furthermore, not all features can be present simultaneously. It makes sense that a specific pronoun has the features **singular** and **male**, but it does not make sense to say that it has the features **singular** and **plural** — the latter pair is just not compatible[7]. So for any two features f, g, $comp(f,g)$ is true when f and g are compatible, and false otherwise. A subset V of some set W is said to be maximally compatible if any element in W that is not in V is incompatible with at least one element in V. We will use these sets below to define both the actual structures and the relations between these.

[7] Note that the example makes use of atomic feature, i.e. features that are atoms in Bouma's system. He seems to assume that only atoms are ever incompatible, and that if two (normal) features are admissible for a given feature f, they are also compatible. Certainly some of his results seem to depend on this assumption (cf (22) and the last example of this section).

We would prefer to define the ordering and the *msg*, *mgu* and *mscd* directly in terms of the feature structures themselves, but writing rules for such two dimensional objects is a bit clumsy. We therefore define a linear language that is equivalent to it. Given a set of features $\mathcal{F}$, the set of feature structures $\mathcal{FSTR}$ is the smallest set such that:

(21) a. $\top \in \mathcal{FSTR}$

b. *If* $f \in \mathcal{F}$ *and* $g : \phi \in \mathcal{FSTR}$ *then* $f : g : \phi \in \mathcal{FSTR}$, *provided* $g \in adm(f)$

c. $f : \phi \in, g : \psi \in \mathcal{FSTR}$, *then* $f : \phi \wedge g : \psi \in \mathcal{FSTR}$, *provided* *comp(f,g)*

d. $\vec{f_1}, \ldots, \vec{f_n} \in \mathcal{F}^*$, *then* $[\vec{f_1}, \ldots, \vec{f_n}] \in \mathcal{FSTR}$

As a consequence, feature structure containing incompatible sub-branches or non admissible sub-structures are excluded. For example, if we follow what seems to be the general assumption, that the following properties hold for *adm* and *comp*:

(22) a. *All atomic features are incompatible.*

b. *all non-atomic features f,g are compatible.*

c. *the set of features admissible after a feature contains either only atomic features or only non-atomic ones.*

the feature structures will not include an object of the form $a \wedge b$, for a, b atomic features. Consequently, no unification defined in terms of the ordering can give it as a result (it doesn't exist, so how could it). We will see thatthis will make $a \sqcap b$undefined, or, in terms given earlier, $a \sqcap b = \perp$. This is what we expect for atoms, given our assumptions.

4.4 Ordering the Feature Structures

For the expressions defined by (21) to mean the same as the re- entrant formulas we want it to represent, we have to define rules for how to use it. In particular, expressions that represent the same feature-structure should turn out to be equivalent under the ordering we define below, otherwise this representation is not of much use.

The following ordering gives the required structure The ordering of the feature structures is inductively defined by the following axioms:

first five axioms that are standard for all subsumption orderings of the kind we are looking for:

(a) $\top \preceq \top$

(b) $f \preceq g$ iff $f = g$

(c) $(f : \phi) \preceq (g : \psi)$ iff $f = g$ and $\phi \preceq \psi$

(d) $(\phi \wedge \psi) \preceq (\chi \wedge \rho)$ if $\phi \preceq \chi$ and $\psi \preceq \rho$

Next there are four structural axioms to make sure that the linear formulas are order insensitive and really represent the graphs that are "actual" feature structures:

(e) $(\phi \wedge \psi) \preceq (\psi \wedge \phi)$ always

(f) $(\phi \wedge (\psi \wedge \chi)) \preceq ((\psi \wedge \phi) \wedge \chi)$ always

(g) $f : (\phi \wedge \psi) \preceq (f : \phi \wedge f : \psi)$ always

(h) $(f : \phi \wedge f : \psi) \preceq f : (\phi \wedge \psi)$ always

Then there are two axioms that says that you can glue on new branches at will to make a structure more specific, but that if the branch is the same it doesn't add anything:

(i) $\phi \wedge \psi \preceq \phi$ always

(j) $\phi \preceq \phi \wedge \phi$ always

And finally we have to give axioms to make sure that the expression for re-entrancy does what we expect it to do:

(k) $[f_1, f_2, \ldots, f_n] \preceq [f_2, \ldots, f_n]$

(l) $[f_1, g_1, \ldots, g_m] \wedge [f_1, f_2, \ldots, f_n] \preceq [f_1, f_2, \ldots, f_n, g_1, \ldots, g_m]$

(m) $[f_1, f_2, \ldots, f_n] \wedge \bigwedge_{g \in V} f_1 : g \preceq \bigwedge_{g \in V} [f_1 : g, \ldots, f_n : g]$

(n) $\bigwedge_{g \in V} [f_1 : g, \ldots, f_n : g] \preceq [f_1, f_2, \ldots, f_n] \wedge \bigwedge_{g \in V} f_1 : g$

Where (m) and (n) hold for all maximally compatible subsets V of $adm(f_1) \cap \ldots \cap adm(f_n)$.

These axioms characterize feature structures, including the underspecified re-entrant ones, completely. Lets look at some consequences of this definition. We will write $\phi \cong \psi$ if both $\phi \preceq \psi$ and $\phi \succeq \psi$. Let V be any maximally compatible subset of $adm(f_1) \cap \ldots \cap adm(f_1)$, then

(23) $$\bigwedge_{g \in V} [f_1 : g, \ldots, f_n : g] \cong [f_1, \ldots, f_n] \wedge \bigwedge_{g \in V} f_1 : g$$

(24) $$[f_1, \ldots, f_n] \wedge \bigwedge_{g \in V} f_1 : g \cong [f_1, \ldots, f_n] \wedge \bigwedge_{g \in V} f_2 : g$$

A special case that is of some interest is the case where $f_1, \ldots, f_n$ take only atomic features. combining axioms (m) and (n), with the fact that atomic features are mutually incompatible, The following properties can be seen to hold:

(25) $$[f_1, \ldots, f_n] \wedge f_1 : a \cong f_1 : a \wedge \ldots \wedge f_n : a$$

(26) $$[f_1, f_2, \ldots, f_n] \wedge f_1 : a \cong [f_1, f_2, \ldots, f_n] \wedge f_2 : a$$

(27) $$[f_1, f_2, \ldots, f_n] \gg f_1 : a \cong [f_1, f_2, \ldots, f_n] \wedge f_1 : a$$

(28) $$[f_1, f_2, f_3, \ldots, f_n] \gg f_1 : a \wedge f_2 : b \cong [f_3 \ldots, f_n] \wedge f_1 : a \wedge f_2 : b$$

One of the interesting properties of the default unification as described in [Bouma90a, Bouma90b] is that as much information as possible is preserved.

(29) $$\begin{aligned} &[f, g, h, k] \gg (f : (l : a) \wedge g : (l : a)) \cong \\ &(f : (l : a) \wedge g : (l : a)) \wedge [h, k] \wedge \bigwedge_{m \neq l} [f : m, g : m, h : m, k : m] \end{aligned}$$

where the big conjunction $\bigwedge$ is over all suitable value for m, in our case these are expressed by the sets of admissible features. Using the axioms, and in particular, using (n) we can see that our system gets the same result, provided only atoms are ever incompatible (cf. footnote 7).

5 Conclusion

Mechanisms like the calculation of the Common Denominator have been proposed before to account for a number of linguistic phenomena. Almost always this is formulated as being the determination of some "underlying question" that a sentence is an answer to. For example Steedman mentions it as such in [Steedman90]. We would like to suggest that this might be turned around. At least part of the discourse meaning of a question is to establish an undisputable explicit common denominator, the only coherent continuation left being to give an answer. In fact something like this has already been suggested in [Groe&Stokh84].

This paper is intended to give some idea of the use of the notion of Common Denominator. The definition of the notion itself is straightforward[8], and we discussed a number of applications in a sketchy fashion to show where a notion like this might be of some use. We think that the interested reader can fill in the gaps herself. We hope to have shown that the Common Denominator and his sister Default Unification are useful abstract notions that can be applied to a great number of problems. Happy hacking!

References

[Bouma90a]
Gosse Bouma, 1990, 'Non-Monotonic Inheritance and Unification', in: W. Daelemans & G. Gazdar (eds), *Inheritance in NL processing. Workshop Proceedings*, Institute for Language Technology and AI, Tilburg University, The Netherlands.

[Bouma90b]
Gosse Bouma, 1990, 'Defaults in Unification Grammar', in: *Proceedings of the 28th Annual Meeting of the ACL*, Pittsburgh.

[Chierchia87]
Gennaro Chierchia, 1987, 'Structured Meanings, Thematic Roles and Control', in: G.Chierchia, B. Partee & R. Turner (eds), *Properties,Types and Meaning, Vol II*, D.Reidel, Dordrecht .

[Groe&Stokh84]
Jeroen Groenendijk & Martin Stokhof, 1984, *Studies on the Semantics of Questions and the Pragmatics of Answers*, diss., University of Amsterdam.

[Polanyi85]
Livia Polanyi, 1985, 'A Theory of Discourse Structure and Discourse Coherence', in: *Papers from the general session of the Chicago Linguistic Society, CLS 21*, pg. 306 - 322, 1985.

[Prüst&Scha90]
Hub Prüst & Remko Scha, 1990, 'A Discourse Perspective on VP

[8]although its applications might not be. That should not come as a surprise. Addition and Multiplication are extremely simple notions, but it is still far from trivial to solve a cubic equation.

Anaphora', in: Stokhof & Torenvliet (ed), Proceedings of the 7th Amsterdam Colloquium, University of Amsterdam: Institute for Language, Logic and Information, 1990.

[Prüst,Berg&Scha]
Hub Prüst & Martin van den Berg & Remko Scha, manuscript, *A Formal Discourse Grammar tackling Verb Phrase Anaphora.*

[Rooth85]
Mats Rooth, 1985, *Association with Focus*, diss. University of Massachusetts, Amherst.

[Scha&Polanyi88]
Remko Scha & Livia Polanyi, 1988, 'An Augmented Context free Grammar for Discourse', in: *Proceedings of the 12th International Conference on Computational Linguistics (COLING)*, pg. 2227, 1988.

[Steedman90]
Mark Steedman, 1990, 'Gapping as Constituent Coordination', in: *Linguistics and Philosophy* **13**, 207–263, Kluwer Academic Publishers, Dordrecht, Boston, London.