THE SIXTH JAPANESE-KOREAN
JOINT CONFERENCE
ON FORMAL LINGUISTICS, 1991

Complementation, Quantification and Potential Energy

HASIDA, Kôiti
Institute for New Generation Computer Technology (ICOT)
Mita Kokusai Bldg. 21F, 1-4-28 Mita,
Minato-ku, Tokyo 108 JAPAN
Tel: +81-3-456-3194,
E-mail: hasida@icot.jp, hasida%icot.jp@relay.cs.net

Abstract

Cognitive science in general and linguistics in particular necessitates some analog component of descriptive device, in order to account for the continuous gradation of the preferences among interpretations of utterances, for example. This paper proposes a descriptive formalism consisting of two layers, i.e., a system of symbolic aspect of constraints based on first order logic and a theory of cost or potential energy of the symbolic constraints. A grammar fragment is presented which accounts for some basic phenomena concerning complementation, adjunction and quantification of Japanese. This grammar reduces the combinatorial complexity, especially that of syntax, attributing relevant constraints to semantics and the analog component of the theory. A formal theory of the analog component is also sketched, and some implication of our approach with respect to cognitive science is discussed.

1 Introduction

In this paper we propose to incorporate an energy-based or cost-based component into the descriptive formalism for cognitive science in general and linguistics in particular. One thing that motivates this proposal is that linguistics should account for the graded nature of acceptability of utterances and preference of interpretations of utterances. An energy-based, analog component is necessary in order to capture such a continuous gradation. Given such a component, more preferable (or acceptable) interpretation would be assinged smaller potential energy (or smaller cost), and less preferable ones greater potential energy.

One step further, potential energy may also serve as the basis on which to control inferences. Namely, information processing would proceed towards energy minimization. Note further that the preference of interpretation and that of inference cannot be distinguished from each other. In fact, the most preferred interpretation is also the one which should be investigated most diligently. This means that when a linguistic theory formulates preference or acceptability of interpretation, it must automatically capture preference of inference as well.

Instead of introducing an analog component, one might attempt to work out a totally symbolic specification of the preference in question, as non-monotonic logics (McCarthy

(1980), McDermott and Doyle (1980), among others), for instance, have attempted to. A problem with this sort of approach, however, concerns the treatment of context sensitivity. The preference at issue must change from case to case, so that the preferred interpretation of a linguistic expression depends on the context. Totally symbolic specification of context sensitivity tends to be too complex to be a scientific account, due to the diversity of contexts; i.e., essentially for the same reason why it is impossible to symbolically define the meanings of words in terms of necessary and sufficient conditions. Another problem is that symbolic specification of preference increases combinatorial complexity of both description and computation. Fully symbolic approaches are hence less desirable than energy-based ones, in which combinatorial (alias symbolic) description is minimized and combinatorial inference is controlled without introducing extra combinatorial complexity.

The rest of the paper is organized as follows. Section 2, the main part, presents some grammar fragments to demonstrate how combinatorial complexity of a linguistic theory may be reduced by introducing an analog component based on a sort of cost. Section 3 gives a formalization of this cost in terms of potential energy, and suggests how to control inference based on an energy minimization principle. This line of reasoning suggests a formulation of situation theory as a multi-layered system consisting of a symbolic component and an energy-based component, instead of as a single uniform symbolic system.

2 Linguistic Constraints

The following discussion assumes the classical first order logic as the basis for modeling the combinatorial aspects of the reality. What the world is like is captured in terms of a constraint, which is a first order logic program of a clausal form. There is just one special clause, called *the top clause*, which, given the other clauses, represents the modelized part of the world. The top clause is of the form `?- B.`, as below, where B is a sequence of literals.[1]

(1) `?- p(X,Y), q(X).`

This top clause postulates that the world should satisfy `p(X,Y)` and `q(X)` for some `X` and `Y`.[2] Information processing is captured in terms of abduction (backward chaining), deduction (forward chaining) and factoring (unification). For expository convenience, such information processing is assumed to be done by rewriting the top clause.[3]

We will use 'literal' in a broad sense, so that literals include variable bindings. Thus, literal `X=f(Y)` consists of predicate `=f` and the first and the second arguments `X` and `Y`, respectively. As usual, one variable cannot be bound in two different ways in terms of two different predicates. Thus, `X=f(Y)` and `X=g(Z)` are incompatible. `p(f(X))`, for instance, stands for the conjunction of two literals `p(Y)` and `Y=f(X)`.

The literals are classified into two categories, *assumable* and *unassumable*. Assumable literals are those with *assumable* predicates. A predicate is assumable if it is NOT defined in terms of definition clauses (the clauses which have head literals with that predicate); Otherwise it is assumable. Predicates like `=f` are assumable, and thus variable bindings are typical assumable literals. For the sake of explanatory ease, we tentatively suppose that inference may stop only when the top clause contains only assumable literals.

Here we tentatively introduce what we will call the *cost* of constraints, just in order to informally demonstrate the efficacy of energy-based account. Section 3 will give a formalization of the cost in terms of potential energy that captures the essence of the discussion in this section. Like *assumability cost* (Hobbs et al. (1988)), our cost is the cost at which to maintain the top clause. Thus the top clause had better be rewritten through inferences so as to have smaller cost. Smaller cost corresponds to better interpretation of a sentence, for example. For simplicity, we assume that cost is primarily assigned to literals in the top clause, and that the cost of the top clause is the sum of the costs of the literals it contains. In this section we do not care small differences of the cost, but just distinguish very costly literals by enclosing them within boxes, like $\boxed{\texttt{p(X)}}$.

Throughout the following discussion, we employ a very tentative and simplified version of cost dynamics. That is, when two literals with costs α and β unify (factor), the resulting literal has cost $\alpha\beta$.[4] This means that literals with large cost should unify with others, so that the entire cost of the top clause be reduced. In fact, more intricate patterns of desired factoring may be encoded in terms of the cost assignment. For instance, (2) may be programmed so as to be best interpreted as one of the 6 clauses in (3), provided that `p` is assumable.

(2) `?-` $\boxed{\texttt{p(X)}}$, $\boxed{\texttt{p(Y)}}$, $\boxed{\texttt{p(Z)}}$, `p(a), p(b), p(c).`

```
(3) ?- X=a, Y=b, Z=c, p(a), p(b), p(c).
    ?- X=a, Y=c, Z=b, p(a), p(b), p(c).
    ?- X=b, Y=c, Z=a, p(a), p(b), p(c).
    ?- X=b, Y=a, Z=c, p(a), p(b), p(c).
    ?- X=c, Y=a, Z=b, p(a), p(b), p(c).
    ?- X=c, Y=b, Z=a, p(a), p(b), p(c).
```

Tentatively, an appropriate cost assignment here is that the cost is 10 for `p(X)`, `p(Y)` and `p(Z)`, 0.5 for `p(a)`, `p(b)` and `p(c)`, and 0 for each binding. This will prevent `X=Y=a`, for instance, because that would impose cost 50 upon `p(a)`, whereas the cost of each interpretation in (3) is 15. Thus, a single factoring as in (3) is allowed but a double factoring is not.

Another way of encoding roughly the same constraint as (2) would be:

```
(4) ?- p(X), p(Y), p(Z), X≠Y, Y≠Z, Z≠X.
    p(a).
    p(b).
    p(c).
```

This is more complex than (2), in the sense that 3 extra literals and 3 extra clauses are introduced here, whereas the cost has nothing to do with combinatorial complexity. The cost thus allows us to have simpler representation of constraints. In what follows, we will work out more implicit and simpler equivalents for **subcat** feature (Gunji (1987), Pollard and Sag (1987)) and quantifier storage (Cooper (1983)).

2.1 Complementation and Adjunction

First let us take a look at a role the cost may play in putting smaller expressions together to form bigger ones.

In the following discussion, grammatical categories are represented in terms of records, (alias feature bundles or attribute-value pairs), such as `{pos/v, sem/X}`. A record is regarded as a set of literals. For instance, `Cat = {pos/v, sem/X}` is a synonym of the following conjunction of literals.

(5) `pos(Cat,v)` $\wedge$ `sem(Cat,X)`

Feature names, such as `pos` and `sem`, are all assumable predicates. They are partial functions, too, in the sense that the first argument uniquely determines the second argument. For example, if `pos(X, Y)` and `pos(X, Z)`, then `Y = Z`. In other words, unification (factoring) between `pos(X, Y)` and `pos(X, Z)` is compulsory. We assume that a record cannot be bound. For instance, `sem(C, X)` and `C = a` do not obtain simultaneously.

We consider the following phrase structure rule.

```
(6) constituent(Head,STR0) :-
        constituent(Modifier,STR1), constituent(Head,STR2),
        modify(Modifier,Head), concatenate(STR1,STR2,STR0).
```

Predicate `concatenate` means that the concatenation of the first and second arguments is the third argument. For instance, `concatenate("foo","bar","foobar")` is true. `modify` is an assumable predicate. We tentatively assume that it is a partial function, again in the sense that the first argument uniquely determines the second one. As it turns out shortly, this is to the effect that every syntactic constituent except the rightmost one modifies exactly one other constituent.

To interpret sentence (7), lexical entries (8), (9) and (10) are postulated.[5]

(7) Gakusei -ga sake -wo nonda.
student NOM sake ACC drank
'A student[6] drank sake.'

(8) `constituent(Cat,"gakusei-ga ") :-`
$\quad\texttt{S}\models\texttt{X}\bullet\langle\langle\texttt{student,X}\rangle\rangle\texttt{,}$
$\quad\texttt{Cat=\{pos/p,form/ga,sem/X\},}\ \boxed{\texttt{modify(Cat,\{pos/v\})}}\,\texttt{.}$

(9) `constituent(Cat,"sake-wo ") :-`
$\quad\texttt{S}\models\texttt{X}\bullet\langle\langle\texttt{sake,X}\rangle\rangle\texttt{,}$
$\quad\texttt{Cat=\{pos/p,form/wo,sem/X\},}\ \boxed{\texttt{modify(Cat,\{pos/v\})}}\,\texttt{.}$

(10) `constituent(Cat,"nonda ") :-`
$\quad\texttt{Cat=\{pos/v,sem/}\langle\langle\texttt{drank,X,Y}\rangle\rangle\texttt{\},}$
$\quad\boxed{\texttt{modify(\{pos/p,form/ga,sem/X\},Cat)}}\,\texttt{,}$
$\quad\boxed{\texttt{modify(\{pos/p,form/wo,sem/Y\},Cat)}}\,\texttt{.}$

The intuitive meaning of the cost assignment here is: *gakusei-ga* and *sake-wo* should modify some verbal category, and *nonda* should be modified by a *ga*-phrase and a *wo*-phrase.

Feature name `pos` stands for *part of speech*, `form` stands for *form of postposition*, and `sem` stands for *semantics*. Infix predicate $\models$ reads 'contains.' In the terminology of Situation Theory (Barwise and Perry (1983), Barwise and Etchemendy (1987)), literal $\texttt{S}\models\texttt{A}$ means that either *situation* `S` supports *soa* `A`, or a non-soa object `A` belongs to `S`.[7] $\models$ is a partial function from the second argument to the first argument. Namely, we assume that an soa or an object is located in at most one situation. Incidentally, we may write, for instance, (11) instead of (12) for visual simplicity.

(11) `S|⊨A•B•C`

(12) `S|⊨A ∧ S|⊨B ∧ S|⊨C`

Soas are represented in terms of records as grammatical categories are. For instance, soa ⟨⟨**drank, X, Y**⟩⟩ is a shorthand of the following record.

(13) `{rel/drank,agent/X,patient/Y}`

Sentence (7) is represented in terms of the following top clause.

(14) `?- constituent(Cat,"gakusei-ga sake-wo nonda ").`

We consider that to interpret (7) is to derive from (14) a new top clause (approximately) all of whose implications are consistent with the other clauses. Let us call such a new clause the *interpretation* of (7). To get an interpretation, (14) is transformed into the following through resolution with (8), (9) and (10).

(15) `?- S0|⊨X0•⟨⟨student,X0⟩⟩,`
`S1|⊨X1•⟨⟨sake,X1⟩⟩,`
`Cat0={pos/p,form/ga,sem/X0}, modify(Cat0,Cat),`
`Cat1={pos/p,form/wo,sem/X1}, modify(Cat1,Cat),`
`Cat={pos/v,sem/⟨⟨drank,X,Y⟩⟩},`
$\boxed{\texttt{modify(\{pos/p,form/ga,sem/X\},Cat)}}$,
$\boxed{\texttt{modify(\{pos/p,form/wo,sem/Y\},Cat)}}$.

The compulsory unifications among literals of the form `modify(···)` have been finished here, leaving two literals `modify(Cat0, Cat)` and `modify(Cat1, Cat)`. A better interpretation is one with smaller cost. The minimum cost is obtained by further factoring the literals of the form `modify(···)` in the expected combination, and accordingly unifying other literals:[8]

(16) `?- S0|⊨X•⟨⟨student,X⟩⟩,`
`S1|⊨Y•⟨⟨sake,Y⟩⟩,`
`Cat={pos/v,sem/⟨⟨drank,X,Y⟩⟩},`
`modify({pos/p,form/ga,sem/X},Cat),`
`modify({pos/p,form/wo,sem/Y},Cat).`

Thus (16) has been inferred from (14) by default.

As is illustrated by this example, the above grammar fragment captures the following principle.

(17) A specified argument place of a verb requires a complement to be associated with it.[9]

We can encode principle (18) as well, which predicts the low acceptability of (19).

(18) Only one complement should be associated with one argument place of a verb.

(19) Gakusei -ga gakusei -ga kita.
student NOM student NOM came
'A student a student came.'

(18) is implemented along the line discussed before. That is, we might assume that the costs of costly literals and ordinary literals of the form `modify(···)` are 10 and 0.05, respectively.[10] This will allow double factoring, resulting in cost 5, but disallow triple factoring, resulting in cost 50. (19) is hence predicted to be more costly, hence less acceptable, than *gakusei-ga kita* 'a student came.'

Note also that the above grammar fragment captures the freedom of the order among the modifiers (complements and adjuncts) of a verb. For instance, it allows *sake-wo gakusei-ga nonda* as well as *gakusei-ga sake-wo nonda*.

Owing to the cost-based component, our account of complementation is much simpler than that in those theories, in terms of combinatorial complexity. In contrast, the HPSG and JPSG implement (17) and (18) in terms of **subcat** feature plus the Subcat Feature Principle (SFP), which are of combinatorial nature. The value of **subcat** feature is a list, a combinatorial structure, and the application of SFP accompanies combinatorial operations where a new list is made by subtracting (or adding, if you look at it on a top-down basis) an element in a given list, in order to embody (18). JPSG assumes that the **subcat** value is a set instead of a list. Although a set might appear less combinatorial than a list, however, the application of SFP is no less complicated with a set than with a list: you must process all the elements of a set when you tailor another set from it by adding or subtracting some elements.

Further, our approach naturally accounts for adjunction also. So we are using the name of the predicate '`modify`' in a sense neutral between complementation and adjunction. In the case of adjunction, the lexical entry of the verb does not contribute a literal of the form $\boxed{\texttt{modify}(\cdots)}$, so that the thematic role of the adjunct is determined by the adjunct itself and the semantic property of the verb. Thus complementation and adjunction are given parallel accounts. Namely, both phenomena are accounted for in terms of cost-driven factoing among literals of the form `modify(···)`, and the only difference between the two is that the verb contributes some syntactic information to determine the thematic role of a complement but not that of an adjunct. So our approach contrasts again with the accounts based on **subcat** feature, which sharply distinguish between complements and adjuncts, in the sense that complementation is accounted for in terms of **subcat** feature and adjunction in terms of some other feature.

Note that our approach is more similar to dependency grammars (Sgall and Panevová (1989)) than the theories based on constituency, including GPSG, HPSG, JPSG, LFG, and transformational grammars. The above grammar fragment builds no constituent structure, because the mother category and the head daughter are the same thing in complementation and adjunction. Instead, the literals of the form `modify(···)` represent dependencies among the categories.

So-called θ-criterion is formulated in terms of cost assignment, as a purely semantic rather than syntactic principle. For example, the semantic structure of *wo*-phrase of Japanese might include the following.

(20) `patient(X,SOA)` $\wedge$ $\boxed{\texttt{rel(SOA,R)}}$

This means that *wo*-phrase should require a soa to which it supplies the patient-role parameter.[11] In this connection, our approach will also contribute to the investigation of children's language use (Cooper (1989)) and more drastic attempts to minimize syntax (Suzuki and Tutiya (1989)).

2.2 Quantification and Abstraction

Here we consider how quantifier scoping may be treated without explicit encoding of quantifier storage (Cooper (1989)). Unlike Gawron and Peters (1989), we consider the scopes of abstraction (or quantification in particular) as situations. This introduces a quasi-order relation $\preceq$ among situations. That is, situations greater with respect to $\preceq$ correspond to wider scopes.[12] There are several maximum (not merely maximal) scopes, which correspond to the reality or, in Situation Semantics term, the frames of minds of cognitive agents.

Sentence (21) has two plausible readings, (22) and (23), of which the former is the default interpretation.

(21) Gakusei -ga san -nin sake -wo ro -ppon nonda.
student NOM three person sake ACC six bottle drank
'3 students drank 6 bottles of sake.'

(22) 3 students drank 6 bottles of sake as a whole. (branching quantification)

(23) Each of 3 students drank 6 bottles of sake. (wide scope reading of *san-nin*)

To account for this, we assume the following lexical entries.

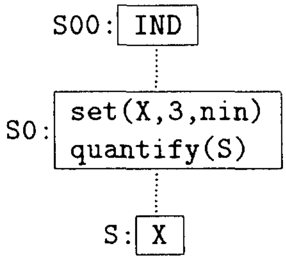
(24)
```
constituent(Cat,"san-nin ") :-
    S00|⊨IND, S0|⊨set(X,3,nin)•quantify(S), S0⪯S00,
    S|⊨X, Cat={pos/p,form/FORM,sem/X}, ga_wo(FORM),
    modify(Cat,{pos/v,sem/IND}) .
```

(25)
```
constituent(Cat,"ro-ppon ") :-
    S00|⊨IND, S0|⊨set(X,6,hon)•quantify(S), S0⪯S00,
    S|⊨X, Cat={pos/p,form/FORM,sem/X}, ga_wo(FORM),
    modify(Cat,{pos/v,sem/IND}) .
```

(26)
```
constituent(Cat,"nonda ") :-
    Drank=⟪drank,X,Y⟫, project(Drank,DRANK),
    Cat={pos/v,sem/DRANK},
    modify({pos/p,form/ga,sem/X},Cat) ,
    modify({pos/p,form/wo,sem/Y},Cat) .
```

Lexical entries for *gakusei-ga* and *sake-wo* are the same as before. `set(···)` denotes a set. For example, `set(X, 3, nin)` denotes a set of 3 people each of whom is like `X`. The semantic structure encoded in the body of (24) may be depicted like this:

(27)



Here each situation is illustrated as a box containing the objects and soas that this situation contains. Relation $\preceq$ among situations is represented as dotted lines between the boxes. The upper end situation of each line is greater than the lower end with respect to $\preceq$.

We further postulate some clauses as below.

(28) project(IND,IND).

(29) project(IND0,IND1) :- S$\VDash$IND0, project($\boxed{\texttt{quantify(S)}}$,IND1).

(30) S$\preceq$S.

(31) S0$\preceq$S1 :- S$\VDash$$\boxed{\texttt{quantify(S0)}}$, S$\preceq$S1.

(32) S0$\preceq$S1 $\leftarrow$ S0$\VDash$SOA, agent(SOA,X), S1$\VDash$X.

(33) S0$\preceq$S1 $\leftarrow$ S0$\VDash$SOA, patient(SOA,X), S1$\VDash$X.

(34) ga_wo(ga).

(35) ga_wo(wo).

Due to (28) and (29), project(A, B) means that B is obtained by applying quantification to A more than or equal to zero time. Thus, a literal of the form project($\cdots$) in the lexical entry of a verb accounts for why a sentence such as *tori-wa tobu* 'a bird flies' may be interpreted as if it involved a universal quantifier.[13] So our theory on such an implicit quantification will be purely semantic. Although (29) and (31) allow infinitely many different interpretations, it is not just the case that anything goes.

Literal p($\boxed{\texttt{f(X)}}$) is a synonym of the following.

(36) p(Y) $\wedge$ $\boxed{\texttt{Y=f(X)}}$

So $\boxed{\text{quantify}(\cdots)}$ indicates that there should better be a quantification introduced elsewhere in order to use (29) or (31). As discussed below, this explains why (21) has the interpretation preference mentioned above.

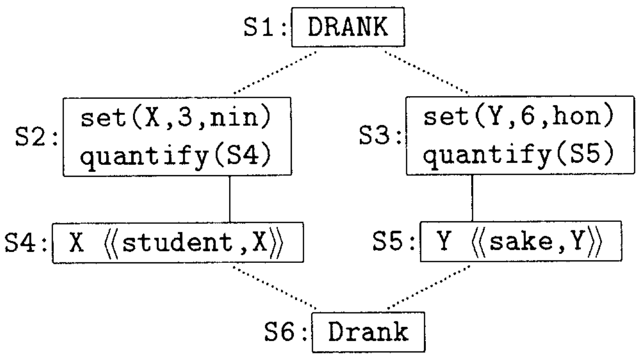
From (37) we can infer (38) as an abductive explanation, after reducing the cost of the literals of the form modify($\cdots$) by factoring.

(37) ?- constituent(Cat,"gakusei-ga san-nin sake-wo ro-ppon nonda ").

(38) ?- S1$\VDash$DRANK, project(Drank,DRANK),
S2$\VDash$set(X,3,nin)$\bullet$quantify(S4), S2$\preceq$S1,
S3$\VDash$set(Y,6,hon)$\bullet$quantify(S5), S3$\preceq$S1,
S4$\VDash$X$\bullet\langle\!\langle$student,X$\rangle\!\rangle$, S6$\preceq$S4,
S5$\VDash$Y$\bullet\langle\!\langle$sake,Y$\rangle\!\rangle$, S6$\preceq$S5,
S6$\VDash$Drank, Drank=$\langle\!\langle$drank,X,Y$\rangle\!\rangle$,
modify({pos/p,form/ga,sem/X},Cat),
modify({pos/p,form/wo,sem/Y},Cat),
Cat={pos/v,sem/DRANK}.

This inference involves deduction by (32) and (33), which gave rise to two literals S6$\preceq$S4 and S6$\preceq$S5. As is suggested by the notation different from the other clauses, these clauses do not constitute the definition of $\preceq$, and thus are used only deductively (i.e., forward inference for consistency checking) rather than abductively (i.e., resolution with the top clause). The semantic aspect of (38) is pictorialized as follows.

(39)

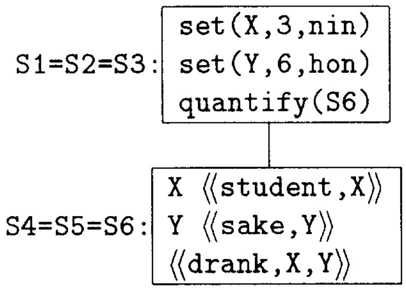


The lines between S2 and S4 and between S3 and S5 indicate that there are no intervening situations in between regarding ⪯.

Three minimal interpretations obtained by further abduction unfolding the literals with predicates ⪯ and `project`[14] and some factoring after that are the following, together with the illustrations of the semantic aspects.

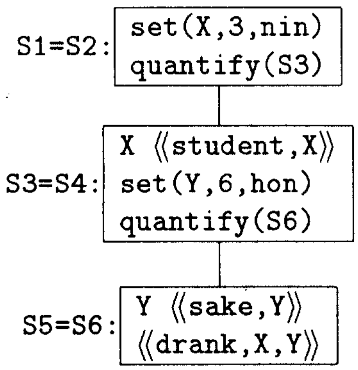
(40) 3 students drank 6 bottles of sake as a whole. (branching quantification)

```
?- S1=S2=S3, S4=S5=S6, DRANK=quantify(S6),
   S1|=set(X,3,nin)•set(Y,6,hon)•DRANK,
   S6|=⟨⟨drank,X,Y⟩⟩•X•⟨⟨student,X⟩⟩•Y•⟨⟨sake,Y⟩⟩, ⋯.
```



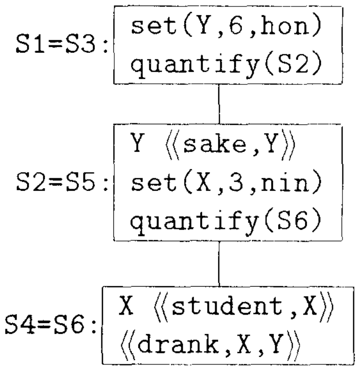
(41) Each of 3 students drank 6 bottles of sake. (wide scope reading of *san-nin*)

```
?- S1=S2, S3=S4, S5=S6, DRANK=quantify(S3)
   S1|=set(X,3,nin)•DRANK,
   S3|=X•⟨⟨student,X⟩⟩•set(Y,6,hon)•quantify(S6),
   S6|=⟨⟨drank,X,Y⟩⟩•Y•⟨⟨sake,Y⟩⟩, ⋯.
```



(42) Each of 6 bottles of sake was drunk by 3 students. (wide scope reading of *ro-ppon*)

```
?- S1=S3, S2=S5, S4=S6, DRANK=quantify(S2),
    S1|⊨set(Y,6,hon)•DRANK,
    S2|⊨Y•⟨⟨sake,Y⟩⟩•set(X,3,nin)•quantify(S6),
    S6|⊨⟨⟨drank,X,Y⟩⟩•X•⟨⟨student,X⟩⟩, ⋯.
```



Our of these, only (42) is barred. Basically the reason is that S2 cannot contain set(X, 3, nin) and quantify(S6). The relevant constraints may be stated as below.

(43) A situation should be *individuated* in order to accommodate a quantification.

(44) A situation introduced by a floating quantifier is individuated only if that quantifier is associated with the subject.

Informally, a situation is individuated when the quantities contained in it are definite. The notion of individuation is yet to be elaborated on a linguistic basis, but it satisfies the present purpose just to state formal relationships that it stands in with the rest of the story. (43) is encoded by adding indiv(S0) to (the bodies of) (24) and (25). So (24) should be replaced by the following.

(45)
```
constituent(Cat,"san-nin ") :-
    S00|⊨IND, S0|⊨set(X,3,nin)•quantify(S), indiv(S0),
    S|⊨X, Cat={pos/p,form/FORM,sem/X},
    ga_wo(FORM), modify(Cat,{pos/v}).
```

(44) may be captured by augmenting (26) by adding indiv(S) for situation S such that S|⊨X, but not for situation S such that S|⊨Y.[15]

(46)
```
constituent(Cat,"nonda ") :-
    Drank=⟨⟨drank,X,Y⟩⟩, project(Drank,DRANK),
    S|⊨X, indiv(S),
    Cat={pos/v,sem/DRANK},
    modify({pos/p,form/ga,sem/X},Cat),
    modify({pos/p,form/wo,sem/Y},Cat).
```

One could encode the same piece of constraint by modifying the lexical entry of postposition *ga* if the relevant constraint is syntactic, or by adding some clause regarding the feature agent if the constraint is semantic.

As for the rest of the interpretations, it is natural to consider that (40) (=(22)) defaults to have smaller cost than (42) (=(23)), because the former is a smaller explanation in terms of the size of the (not necessarily mental) representation, and the total cost of an interpretation is the sum of the costs of its parts. This accounts for why it is the default interpretation. The cost-based account thus straightforwardly captures some aspects, such as the size of the representation, which have been abstracted away from the underlying symbolic logic.

Note that the above account of quantification does not stipulate quantifier storage. However, the information structure corresponding to quantifier storage has automatically emerged from independently motivated pieces of constraints. The most relevant pieces of constraint are clauses (28) through (31), which restrict the scoping among quantifiers. Literals of the form `S0`$\preceq$`S00` in (24) and (25) limit the quantifier scope within the clause boundary.

In this connection, note that our approach exploits minimal syntactic structures. As we have pointed out in the former discussion on the complementation and adjunction, we postulate a dependency hierarchy but no constituency hierarchy in terms of syntax. That is all right because we have enough structure in semantics. For instance, there is a hierarchy of situations as seen in the example discussed above. Although semantic constraints are less computationally tractable than syntactic ones, the analog component of the theory restricts the diversity of semantic interpretation by introducing preferences among them.

3 Potential Energy

Now let us formalize the cost that has so far been informally conceived. A formalization in terms of potential energy is considered, for the convenience of mathematical treatment. We assume that the potential energy E of the entire information structure of (the relevant part of) the reality is a function of the *activation* of the *units* in that information structure. Along the line of the discussion so far, we assume that the combinatorial aspect of this information structure is modelized in terms of a first-order logic program in the clausal form, which is looked upon as a network. Here the atomic formulae are regarded as units, and the activation is a real number which is a monotonically increasing function of the probability of the truth of the atomic proposition. In order to formalize the energy minimization principle on a local basis, we further suppose that E is the sum of the local energy functions; i.e., the sum of the energy of clauses, for instance.

Information processing is controlled on the basis of energy minization principle. The analog part of information processing is spreading activation, and the symbolic part would be some sort of combinatorial inference involving resolution, factoring, etc. In general, every local piece of state transition, analog or symbolic, of the information structure is promoted in accordance with its expected contribution to energy minimization. Thus, the activation u_i of the i-th is increased by the force $F_i = \frac{\partial E}{\partial u_i}$, on the analogy of dynamical system. The spreading activation saturates over the entire network if E has a global minimum and only finite number of minimal points. On the other hand, the control of combinatorial operations is not determined by the energy function E alone, because they change its definition. We must specify which operation changes the energy function in what way, in order to estimate which operation would contribute best to the reduction of the potential energy.

Before defining the inventory of combinatorial operations, we must spell out the energy function E so as to encode the meaning of clauses, among others. Let us consider the following clause for example.

(47) `¬p ∨ ¬q ∨ r.`

We would like to use this clause both for deduction and for abduction. For instance, from `p` we want to deductively infer `¬q ∨ r`, and from `¬p` we want to abductively infer `q ∧ ¬r`.

The following energy function of this clause will meet this end.

$$(48)\ E_{pqr} = A(\exp(c_p u_p + c_q u_q) + \exp(c_q u_q + c_r u_r) + \exp(c_r u_r + c_p u_p)) \\ +D/(\exp(c_p u_p) + \exp(c_q u_q) + \exp(c_r u_r)) \\ +\exp(s_p - c_p u_p) + \exp(s_q - c_q u_q) + \exp(s_r - c_r u_r)$$

A and D are positive constants which we might call *the abduction coefficient* and *the deduction coefficient*, respectively. u_p, u_q and u_r are the activation of the units corresponding to literals `p`, `q` and `r`, and c_p, c_q, and c_r are constants and called the *commitment coefficients* of `p`, `q` and `r`, respectively. In the current example, we have c_p, $c_q < 0 < c_r$ in accordance with (47). s_p, s_q and s_r are constants.

According to the energy minimization principle, (48) as a whole means, very roughly, that just one out of `¬p`, `¬q` and `r` should hold. The first term of its right-hand side intuitively says, with the strength specified by A, that at least one of $c_i u_i$ may be large. Given `¬p`, for instance, we would abductively infer `q ∧ ¬r`, with confidence positively related to $-c_p$ and A. Similarly, the second term says, with the strength specified by D, that some of $c_i u_i$ must be large. Given `p`, for example, we would hence deductively infer `¬q ∨ r`, with confidence positively related to $-c_p$ and D. The last three terms are local energy functions of `p`, `q` and `r`. These terms are included in order to saturate the spreading activation.

Thus, we interpret clauses like (48) as saying something more than the ordinary interpretation which approximates only the second term of the right-hand side of (48). Our energy function encodes some aspects of the probability distribution over the truth-value combinations of the atomic formulae; i.e., aspects concerning the confidence of abductive and deductive inferences.

A full formalization of inference control is definitely a very hard task and calls for a further scrutiny. We will instead tentatively work out how the cost dynamics exploited above in Section 2 may be incorporated into our current framework. According to (48), in an abductive explanation,[16] larger cost of a literal, corresponding to the i-th unit, can be encoded in terms of greater s_i, which tends to render this literal to be true. Let us assume that the unit resulting from unification of the i-th unit and the j-th unit will have its local energy defined by $\exp(s_i + s_j - (c_i + c_j)u_i)$ (now we have $u_i = u_j$). This will enable us the same sort of programming as we have discussed regarding (2) and (18). For instance, $\boxed{\texttt{p(X)}}$ in (2) should have large s_i.

4 Final Remarks

Some grammar fragments have been worked out to demonstrate that some combinatorial aspects of linguistic theory could be attributed to cost-based, analog aspects of the descriptive formalism. We have also discussed a tentative formalization of the cost in terms of potential energy.

If totally symbolic account of language should be abandoned, as we discussed in the beginning, then probably it should also be given up to work out a totally symbolic formal theory of cognitive science in general. Along this line of reasoning, we naturally face the need for formalizing a general descriptive framework, such as Situation Theory, as a multi-layered system consisting of two components, one being a symbolic logic and the other an energy-based theory on preference.

In empirical science, every formal theory is an approximation. As scientists, we believe that everything can be approximated by some formal theory to a satisfactory degree of precision. Whether or not that theory is a good explanation, however, is quite another story. For instance, human information processing could in principle be described precisely enough in terms of neurophysiology or even quantum mechanics. This description, however, is of course not an adequate one, because of its complexity. An adequate theory of human behavior must capture some higher level of abstraction. That is, the theory must employ a folk psychology vocabulary including 'purpose,' 'request,' 'expectation,' 'joy,' 'anger,' etc. In general, however, the precision of approximation becomes worse through abstraction, as details are neglected. Many phenomena which tend to be *continuous* so that small errors tend to have only small influences can be described precisely enough in terms of tractable formal theories. The difficulty in the case of cognitive science, however, is that at least one adequate level of abstraction tend to have a symbolic, or combinatorial structure. Small errors may easily have big effects in combinatorial domains. So formal theories to deal with symbol structures, such as logics, cannot achieve a satisfactory precision.

The underlying view of the reality here would be that the reality is a loosely coupled, local ways of interaction with the environments. Those ways, constituting parts of the reality, may still be contradictory with each other, so that there is no uniform formal theory to capture the reality to the cognitive agent in its entirety. Relativity theory and quantum mechanics are contradictory, for example, but they are parts of the reality to humans in the sense that each of them is a way of abstraction about what the physical world is like. Potential energy is necessary to loosely couple various aspects of the reality which are inconsistent with each other.

Notes

Preliminary versions of this paper have also been presented at PSG Working Group, a regular meeting of AIUEO, and NLU+PSG Joint Workshop. The author has greatly benefited from comments raised there.

[1] DEC-10 Prolog notation being employed, names beginning with capital letters stand for variables (or parameters, if you prefer), and other names constants, including predicates and function names.

[2] Such a model may be an open system, just as ordinary software systems are. For instance, the body of the top clause may have a literal `read_loop(X)`, where predicate `read_loop` is defined as follows.

```
read_loop([X|Y]) :- read(X), read_loop(Y).
```

`read` is a 'built-in' predicate such that `read(X)` provides `X` with some information obtained from the environment.

[3] A more serious account of information processing could be based on transformation of the entire constraint. See Hasida and Ishizaki (1987), Tuda, Hasida and Sirai (1989), and Hasida (1989), among others.

[4] We neglect the dimensions here.

[5] For simplicity, we do not analyze these 'words' into morphemes.

[6] The English translation of *gakusei* may also be 'the student,' 'students' or 'the students'; the Japanese language does not usually mark definiteness and number.

[7] In DRT (Kamp (1981)) terms, $S \| \!\models A$ means that A is a reference marker or a condition in a discourse representation S.

[8] We have done resolution entirely before cost-driven factoring here, but this processing order is just for explanatory ease. Of course it is possible to interpret the sentence on a more incremental basis, for instance.

[9] The optionality of complements in Japanese may be formulated in terms of a lexical rule which deletes argument slots from a verb, for instance.

[10] A symbolic way to implement approximately the same constraint is to augment the grammatical category with a new feature whose value is the part of the surface string associated to the category. This would prevent two different occurrences of *gakusei-ga* to be associated with the same *kita*, for example. We will not take this approach, however, because it raises the combinatorial complexity of the theory.

[11] Of course the story would not be that simple if we take into consideration *wo*-phrases which do not play a patient role, as in *sora-wo tobu* 'fly in the sky.'

[12] Circularity as in the Liar sentence (Barwise and Etchemendy (1987)) is allowed because $S0 \preceq S1$ and $S1 \preceq S0$ may sometimes obtain at the same time.

[13] Although this paper deals only with existential quantifiers, it is straightforward to extend our account to include universal or partitive quantifiers, such as *mina* 'all' and *hotondo* 'most.'

[14] One might notice that, due to (28) and (30), situations may be unified with each other where they should not. This is avoided by binding `S6` to an atom, say **`ground`**, and considering that `X=quantify(S)` and `Y=quantify(S)` implies `X=Y`.

[15] This is too simplistic to capture a wider range of phenomena including passive constructions. Moreover, a more serious approach must take into consideration how common nouns and classifiers contribute to the individuation of scopes.

[16] Note that the polarity of a literal is reversed in an abductive explanation.

References

Barwise, J. and Perry, J. (1983) *Situations and Attitudes*, MIT Press.

Barwise, J. and Etchemendy, J. (1987) *The Liar: An Essay on Truth and Circular Propositions*, MIT Press.

Cooper, R. (1983) *Quantification and Syntactic Theory*, Vol. 21 of *Synthese Language Library*, D. Reidel.

Cooper, R. (1989) "Information in the Early Stage of Language Acquisition," presented at the 1st Conference on Situation Theory and its Applications, Asilomar.

Gawron, J. M. and Peters, S. (1989) *Anaphora and Quantification in Situation Semantics*, manuscript.

Gunji, T. (1987) *Japanese Phrase Structure Grammar*, Reidel.

Hasida, K. and Ishizaki, S. (1987) "Dependency Propagation: A Unified Theory of Sentence Comprehension and Generation," *Proceedings of the 10th IJCAI*, 664-670.

Hasida, K. (1989) *Sentence Processing as Constraint Transformation*, manuscript.

Hobbs, J., Stickel, M., Martin, P., and Edwards, D. (1988) "Interpretation as Abduction," *Proceedings of the 26th Annual Meeting of ACL*, 95-103.

Kamp, H. (1981) "A Theory of Truth and Semantic Representation," in Groenendijk, J., Janssen, T., and Stokhof, M. (eds.) *Formal Methods in the Study of Language*, Mathematical Centre, Amsterdam.

McCarthy, J. (1980) 'Circumscription: A Form of Non-Monotonic Reasoning,' *Artificial Intelligence*, *13*, 27-39.

McDermott, D. and Doyle, J. (1980) "Non-Monotonic Logic I," *Artificial Intelligence*, *13*, 41-72.

Pollard, C. and Sag, I. A. (1987) *Information-Based Syntax and Semantics, Volume 1*, CSLI Lecture Notes No. 13.

Rumelhart, D. E., McClelland, J. L., and The PDP Research Group (1986) *Parallel Distributed Processing — Explorations in the Microstructure of Cognition, Volume 1: Foundations*, MIT Press.

Sgall, P. and Panevovă, J. (1989) "Dependency Syntax — A Challenge," *Theoretical Linguistics*, *15*, 73-86.

Suzuki, H. and Tutiya, S. (1989) "A Strictly Incremental Approach of Japanese Grammar," presented at the 1st Conference on Situation Theory and its Applications, Asilomar.

Tuda, H., Hasida, K., and Sirai, H. (1989) "JPSG Parser on Constraint Logic Programming," *Proceedings of the European Chapter of ACL'89.*