

## エスペラントを仲介言語とする機械翻訳

牧野 悟, 平田 満, 勝守 寛

(中部大学・工学部・工業物理学教室)

### 1. はじめに

われわれは数年前から、エスペラントの特長を生かすことにより、これを仲介言語とする多種言語間の機械翻訳の可能性について検討してきた。<sup>1)</sup> エスペラントから日本語への翻訳処理システムとして以前のものよりかなり改良したものを作成したので報告したい。<sup>2)</sup>

周知のように、一般に  $n$  種類の自然言語  $L_1, L_2, \dots, L_n$  間では  $n(n-1)$  通りの相互翻訳プログラム  $L_i \rightleftharpoons L_j$  を必要とするが、普遍的意味表現、ピボット (pivot) が共通の中間言語として実現すれば  $2n$  通りのプログラムですむことになる。ピボットに到達するのは容易ではないが、一方その代りにエスペラントのような人工的国際補助語を共通の仲介言語  $L_0$  として橋渡しに使う機械翻訳システム (Fig. 1) を考えると、やはり  $n$  種類の自然言語間で  $2n$  通りのプログラムを作ればよいことになる。しかもこの場合エスペラントの合理的構成や規則正しい文法の特長を生かして、エスペラント文の解析、エスペラント文の生成の部分が非常に簡単であり、まいさが少ないため、深いレベルまで下がってピボットに近づくことなく、構文トランスファに意味トランスファを多少加味したレベルで比較的質の良い訳文を手軽に得ることが期待される。

10年余り前に西ドイツで開発された機械翻訳システム SUSY ではドイツ語を仲介として英、独、仏、露、エスペラントなどの言語間の自動翻訳を試みたが、後になってエスペラントも仲介言語に使われた。<sup>3)</sup> 1983年にオランダの BSO 社で発表されたシステム DLT では内部言語としてエスペラントをベースに開発したものが使われている。<sup>4)</sup>

最近大メーカーなどが開発し、すでに販売している実用的な日英間翻訳システムでは、たとえば富士通の ATLAS や日本電気の PIVOT など、意味トランスファからピボットへ一步近づこうとするものと考えられるが、本格的なピボットの理想に到達するにはまだ時間を要するであろう。しかも将来ピボットが実現したとしても異なるメーカーのシステムで別々のピボットが開発されるだろうから、それが本来もっているはずの共通性が保証されるかどうか疑問であり各メーカーともブラックボックスとするであろう。

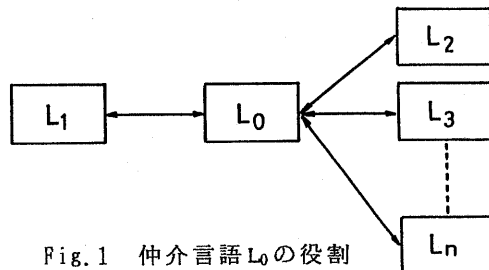


Fig. 1 仲介言語  $L_0$  の役割

他方エスペラントを仲介言語 $L_0$ とするシステムでは、たとえば Fig.1で $L_1$ を日本語とすると、日本語 $L_1$ ⇌エスペラント $L_0$ 間の翻訳プログラムを作っておけば、他の任意の自然言語 $L_i$  ( $i=2, 3, \dots, n$ )とエスペラント $L_0$ 間の翻訳プログラムがどんな方式のものであっても、 $L_0$ を仲介として $L_1$ と $L_i$ 間の ( $L_1=L_0=L_i$ ) 訳文を求めることが容易である。もちろん日本語と英語間の翻訳だけに限れば、すでに実用化されている大がかりなシステムが市販されているし、また近い将来翻訳の需要が多くなると見込まれる言語に対しては大メーカーによって日本語との機械翻訳システムが開発されるであろう。しかし少数民族の言語や国際性の少ない民族語については、同様の翻訳システムが市販されるのを期待することは当分無理であろう。このような少数派言語に対してもエスペラントとの間の翻訳システムさえ作っておけば、エスペラントを仲介として日本語との相互翻訳だけでなく、ヨーロッパで開発されているシステムと組合わせて他の任意の言語との相互翻訳が容易になるものと思われる。

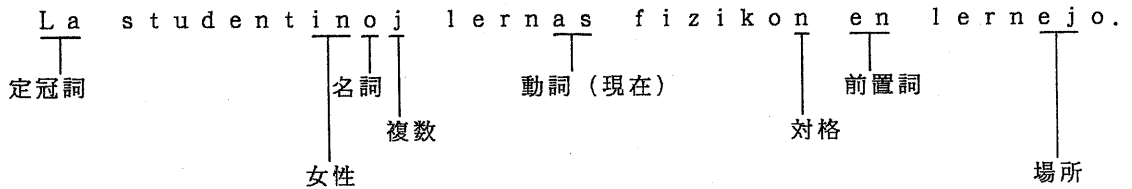
さて一般にある言語を仲介とする2重翻訳、たとえば日本語⇌英語⇌ロシア語のようなことを行うと、人間が行う場合でも多義性などあいまいさが重なって誤訳のおそれが増大することは当然なので、エスペラントを仲介にする場合に対しても同様の懸念が生ずるかもしれない。しかしエスペラントの場合は規則的な体系をもち、例外のない文法、単語に多義性が少ないことなど機械処理にすぐれた特長をもっているため、誤訳のおそれはむしろ減少すると思われる。もちろん完全に近い訳文を得るには前処理、後処理などの人間の介入は各段階である程度必要であるが、エスペラントについてはいわゆる民族語に比べてはるかに少ない労力ですむものと思われる。

## 2. エスペラントの特長と有用性

エスペラントの語彙論的特長は、単語中に文法情報 (grammatical information : 文中における単語の形態・機能の記述) のラベルを含むことである。名詞、形容詞、動詞は語尾が決定しており、動詞は時制、態、相が決定している。また複数、対格も語尾によって示される。このように品詞は確定的であり、また単語には多義性がないことが特長である。(Fig.2)

エスペラントは、膠着言語 (agglutinative language) なので、解析において単語は、はっきりした一連の形態素に分析可能であり、語に含まれる種々の情報を容易に抽出できる。また生成においては単純な語幹から規則的な接辞添加によって語を形成できる。このように既存の概念から新しい概念が容易に作られ、意味が正確になり、基本的な語は少なくてもよい。また不規則変化がないので辞書に独立した項目を与える必要がない。

エスペラントの文法的特長は意識的に合理化された規則正しい体系であり、文法のきまりは、どんな場合にも例外なく通用する。文法以外の習慣的用法はごく限られたものだけである。語彙論的にも品詞が確定的であり、多義性がないことから、エスペラントは文の構造にあいまいさがない。



その女学生たちは学校で物理学を学ぶ。

Fig. 2 単語中の文法情報を示す例文

このようにエスペラントは民族的な自然言語に近い形で実用性がありながら、人工的に作られた規則的な体系を持っていて、機械処理にはきわめて適した言語であることがわかる。もちろんヨーロッパの言語にもとづいて作られているので、特にヨーロッパ諸国における多国語間翻訳の仲介言語として有利なのは当然である。

### 3. エスペラントから日本語への翻訳システム

本システムは、中部大学情報処理センター FACOM M-360AP 上に FACOM OS IV LISP で実現している。

システムの処理の流れは次のようになっている。以下関数名を [ ] で表わすことにする。システム構成をFig. 3に示す。

1. 感嘆文・疑問文の処理 [Transform]
2. 形態素解析 [Morphemic\_Analysis]
3. 構文解析 [Parser]
4. 訳文生成 [Generation]

#### (1) 感嘆文・疑問文の処理 [Transform]

エスペラントでは感嘆文・疑問文は文法機能の位置関係が平叙文と同じであり、文頭に相関語を付加した形で示される。このため感嘆文・疑問文の要素を取り除いた文への変形は非常に容易である。

**感嘆文の処理** — 入力文の最後が感嘆符「！」ならば、この文が感嘆文であると識別する。そして、入力文の最初の2つの単語 (Kia+形容詞, 又はKiel+副詞) を感嘆文の要素として Short\_Term\_Memory (以下STMと略す) に記録し、感嘆符「！」と共に入力文より削除する。

**疑問文の処理** — 入力文の最後が疑問符「？」ならば、この文が疑問文であると識別する。そして、入力文の最初の単語 (Ĉu, 又は疑問詞) を疑問文の要素としてSTMに記録し、疑問符「？」と共に入力文より削除する。

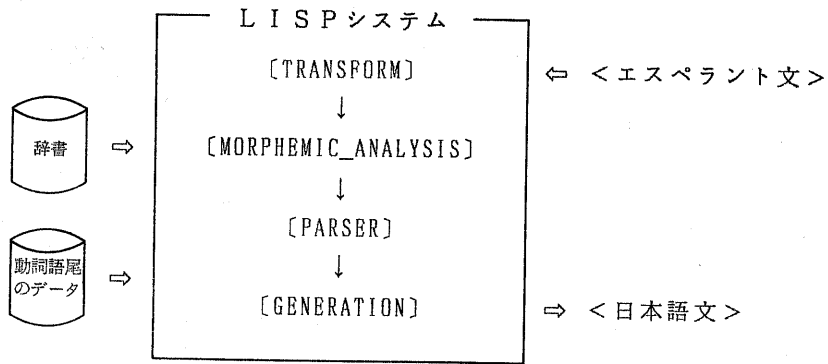


Fig. 3 システム構成

## (2) 形態素解析 [Morphemic\_Analysis]

入力された文の各単語によって辞書を引き、品詞、訳語、などを得る。辞書には基本形が登録されているので、基本形への変形が必要である。この変形の過程で種々の形態素情報が抽出される。

辞書はリスト構造の形をとり、Fig. 4 のように記述されている。辞書は翻訳システム立ち上げ時に、LISPシステム内にローディングされ、属性リストの形で展開される。

エスペラントの単語は規則的な語尾を持ち、不規則変化がまったくなく、Fig. 5 のようにたいへん簡単なアルゴリズムで辞書引きができる。たとえば英語の場合の複雑な語尾分析<sup>5)</sup>と比較すると、いかに簡単であるかわかるであろう。また単語に多品詞性がないのでこのプロセスにおいて品詞が確定される。ここで得られた情報はすべてその単語の属性リストに付加される。

## (3) 構文解析 [Parser]

エスペラントは各単語に多義性がなく、文の構造に決定性がある。このためエスペラント文の解析は構文レベルで行い、利用する知識も非常に限られたものでよい。ここで作成したパーザは、文法規則としては文脈自由型句構造文法による、一方向的解析手法の拡張遷移網 (ATN) にもとづくものである。構文解析の流れをFig. 6 に示す。

パーザはまず重文の処理を行う。並列の文として扱えるものを分割する。そしてその個々の文をATNパーザが処理する。ATNパーザは関数 [Transition] を介してS, NP, VPの各 [Transfer] を関数呼び出しして、再帰的に書き替え規則を適用する。たとえば埋め込み文が現われた場合は、再帰的に [S\_Transfer] が呼び出される。

(AKTIVA (APOS A) (MEAN1 活動的な) (MEAN2 活動的))  
 (TUTE (APOS ADV) (MEAN1 全く))  
 (KAJ (APOS CNJ) (MEAN1 と) (CO-CNJ そして))  
 (LA (APOS DET) (MEAN1 その))  
 (ALFABETO (APOS N) (MEAN1 アルファベット))  
 (ESPERANTO (APOS N) (MEAN1 エスペラント))  
 (FIZIKO (APOS N) (MEAN1 物理学))  
 (INFANO (APOS N) (MEAN1 子供) (PLURAL 子供たち))  
 (MASIN-TRADUKO (APOS N) (MEAN1 機械翻訳))  
 (SINTAKSO-ARBO (APOS N) (MEAN1 構文解析木))  
 (DE (APOS P) (MEAN1 の) (MEAN2 に) (PASSIVE によって))  
 (EN (APOS P) (MEAN1 の中で) (MEAN2 の中) (GOAL の中に))  
 (CU (APOS Q) (MEAN1 か))  
 (KE (APOS Q) (MEAN1 と言うことを) (REL VP) (REL-MEAN のを))  
 (KIA (APOS Q)  
   (MEAN1 どんな)  
   (INTERJECTION なんて)  
   (REL NP)  
   (REL-MEAN NIL))  
 (KIEL (APOS Q)  
   (MEAN1 どのように)  
   (INTERJECTION どんなに)  
   (REL VP)  
   (REL-MEAN のように))  
 (ESTI (APOS VB) (MEAN1 で) (EOAW す))  
 (AMI (APOS VT) (MEAN1 愛) (EOAW する))

Fig. 4 辞書の記述例

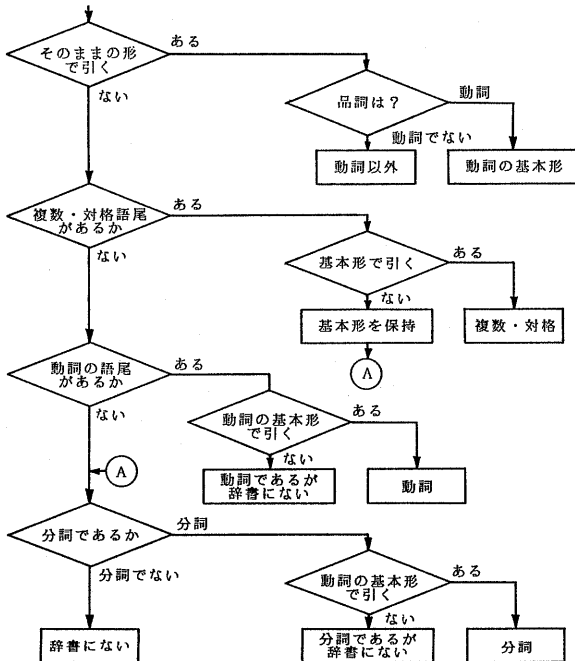


Fig. 5 辞書引きアルゴリズム

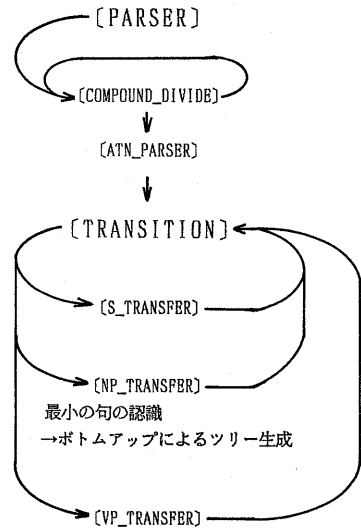


Fig. 6 構文解析の流れ

エスペラント文の名詞句の構造は、その中に含まれる句の位置的自由度が高いことが特徴である。このため英語のように語順の体系による構文解析ではなく、名詞句は特別の処理をする。まず、名詞句は並列に並ぶ最小の句を認識する。ここで単語の品詞名からアークを選ぶのであるが、このとき初期状態においてアークの選択に制約をつけない。このように認識された句から部分的にボトムアップにより句を再構成する。またこの解析途中で訳語選択に関する情報を単語の属性リストに付加する。

#### (4) 訳文生成 [Generation]

訳文生成は次の二つのプロセスによって行われる。訳文生成の流れをFig. 7 に示す。

[Generator] — 構文解析の結果からエスペラントの語順を日本語の語順になるように生成する。[Generator] はデータ駆動によって行われる。構文解析で得た句構造標識の範疇記号から生成規則を選び、日本語の語順になるように生成する。このときNPとVPの間に適切な助詞を付加する。エスペラントではしばしば倒置が用いられる。倒置であると判断すると、関数 [Inversion] が起動され、助詞を付加する位置を適切な位置へ移動する。このように語順によるニュアンスを日本語に反映するようにした。STMに何らかの情報があれば、そのような情報を処理する関数を起動する。

[Translator] — 日本語の語順になったエスペラントの各単語を、動詞の支配のもとで日本語の訳語で置き換える。多くの場合、すでに訳語は選択されており、単語の属性リストによって訳語を得る。動詞の語尾は動詞語尾専用のデータから得る。ここで、対格には適切な助詞を付加する。

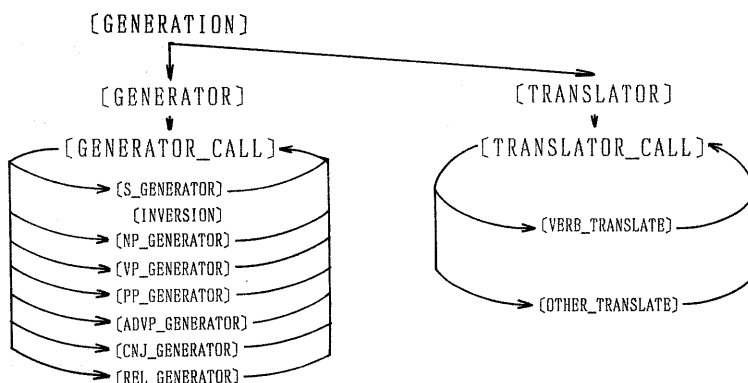


Fig. 7 訳文生成の流れ

#### 4. 訳文例

##### 平叙文の例

((Esperanto estas tre utila lingvo por maŝin-traduko))  
= S ==== NP === NP === (ESPERANTO)  
= VP === V ===== (ESTAS)  
= NP === NP === NP === (TRE UTILA LINGVO)  
= PP === P ===== (POR)  
= NP === (MAŜIN-TRADUKO)  
= (エスペラントは機械翻訳にとってたいへん有用な言語です。)

((Mi ne vizitis lin hieraŭ))  
= S ===== NP === NP === (MI)  
= VP === V ===== (NE VIZITIS)  
= NP === NP === (LIN)  
= ADVP = (HIERAŬ)  
= (私は彼をきのう訪問しませんでした。)

##### 感嘆文・疑問文の例

((Kia bela floro ĝi estas !))  
= (それはなんてすばらしい花でしょう。)

((Kio estas ŝia laboro ?))  
= (彼女の職業は何ですか。)

##### 倒置の例

((Li ĉiam povis alparoli al ŝi en la biblioteko))  
= (彼はその図書館の中で彼女にいつも話しかけることができた。)

((En la biblioteko ĉiam li povis alparoli al ŝi))  
= (その図書館の中で彼は彼女にいつも話しかけることができた。)

((En la biblioteko al ŝi li ĉiam povis alparoli))  
= (彼女にその図書館の中で彼はいつも話しかけることができた。)

##### 重文・複文の例

((Mi sciigis ke dio donis diversajn lingvojn al homoj ,))  
(( kaj homoj kreis diversajn kulturojn per la lingvoj))  
= (私は神が人々に様々な言語を与えたのを知った。そして人々はその言語によって様々な文化を創造した。)

##### 作文テキストからの例文<sup>6)</sup>

((Estas bela vetero))  
= (すばらしい天気です。)

((Mi iris tien multfoje))  
= (私はそこに何度も行った。)

((Ŝi estas ĉiam maltrankvila pri la sanstato de sia edzo))  
= (彼女は自分の夫のその健康についていつも不安です。)

((La gasto alvenos post tridek minutoj))  
= (その来客は三十分あとで到着するでしょう。)

((La fabriko ĉiam akceptas vizitantojn))  
= (その工場は訪問者をいつも受け入れる。)

## 5. おわりに

以上、エスペラントから日本語への翻訳については、エスペラントの特長を生かして作った簡単なシステムで、比較的よい訳文が得られることを説明した。ヨーロッパとくに東欧圏で研究されているエスペラントを出力する自動翻訳システムと組合せれば、たとえばハンガリー語とかポーランド語などから日本語への翻訳が容易に行えるであろう。

ところで日本語からエスペラントへの翻訳については、日本語の解析が複雑であるが、すでに日英翻訳などで十分に研究されている日本語の言語処理の手法を借用することにすれば、エスペラント文の生成については非常に容易にできるはずである。従来われわれが検討してきた日本語からエスペラントへの翻訳プログラムを、この立場から大幅に作り直す作業が、現在進行中である。

## 参考文献

- 1) 勝守, 福田: エスペラントを仲介言語とする機械翻訳の試み, 情報処理学会, 自然言語処理研究会資料44-6 (1984). 1981~1983年度中のわれわれの研究報告はこの文献に示してある.
- 2) 勝守, 平田, 牧野, 名知, 若山: 電気関係学会東海支部連合大会論文集 p.395 (1985)
- 3) H. D. Maas, *Aŭtomata Tradukado en kaj el Esperanto, Lingvo-Kibernetiko, Tübingen*, p.75 (1982). SUSY(Saarbrücker Übersetzungs-System)はSaarland 大学で1972年に作成された. 当初はロシア語からドイツ語への翻訳を目的とした.
- 4) A. P. M. Witkam, *Distributed Language Translation, BSO, Utrecht* (1983)
- 5) T. Winograd, *Understanding Natural Language, Academic Press Inc.* (1972)  
または淵ほか訳, 言語理解の構造, 産業図書 (1976) p.97 図20を参照.
- 6) 阪直, エスペラント初級・中級の作文, 日本エスペラント学会 (1984)