An Approach to a System of Word Combination

語の承け継ぎの仕組みに関する一研究法

水谷静夫

1　序説

1・1　課題　日本語の語順は英語などより自由だと言われる。しかしこれは文節どうしの排列の場合などの事で，詞と辞とのまたは辞どうしの結合の順序には相当きつい拘束がある。まただ二語の間の結合関係では足りない。たとえば現代東京語で，助動詞ダは単独では動詞を承けないが，ウと合したダロウなら承ける。従って承け継ぎの仕組みを調べるには，三つ以上の連なりにも注目しなければならない。この論文ではそうしたデータが既に得られたとしてこれをまとめて行く方法について一つの試みを示す。実際にはいわゆる助動詞の群を扱うが，論文の課題はデータをまとめる方法自体にある。そしてこの方法は助動詞以外にも，また日本語以外にも適用出来るであろう。

1・2　以下に試みる方法の意義　今までの言語研究では，方法原理が掲げられた場合にも，個々の操作までが形式的な厳密さをもって規定されていたとは限らない。筆者は以下の試みでこの段階まで明確な方法を打ち建てようと思う。これが意義の一つである。§4 の実例だけでは分り切った事に仰々しい方法を振り回すと思われようが，更に複雑な事柄を扱う時にはこの種の形式的に定義された操作の組を使うのが有利である。

この方法の背景にはもう一つの意義がある。語の承け継ぎという文法論の一分野では順序が重要な働きをするが，筆者はこれを順序集合の立場から見て行く。筆者は先に語彙論の基本概念を集合論の見地から定義した。[注1)] それは集合の元(げん)の順序関係を捨象した立場であった。ところで結合の際の元の順序に目をとめれば，この論文で扱う類の問題となる。こう考えると，従来さほど密接

1　国研報告13『総合雑誌の用語』後編（昭33；1958），付録Ｉ。

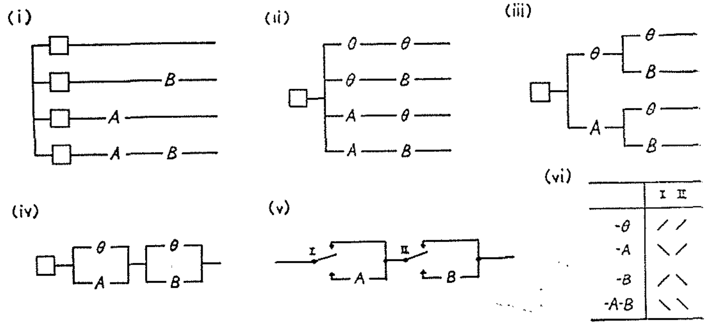
なつながりを方法上は持たなかった語彙論と文法論との相当部分が，集合論の見地から統一的に基礎づけられる可能性を生ずる訳である。

2　語の承け継ぎを電気回路に表わすこと

2・1　着想の由来　周知の通り二値命題論理はスイッチやリレーを使って電気回路に表わせる。命題の真偽をスイッチの開閉の状態に引き当て，導通があれば真，なければ偽と定義する時，選言「P または Q」はスイッチの並列，連言「P かつ Q」は直列に対応する。筆者は初め，この事が語の承け継ぎの問題に応用出来まいかと思った。発話の中である位置を占める言語要素として P か Q かを選ぶことが選言に似，要素 P に Q を連ねることが連言に似るからである。もしこの並行性が確かめられたら，回路の理論を言語研究に適用するのに何の妨げもない。

しかし言語要素の選び・連ねと論理上の選言・連言とは完全には並行しない。連言では命題の順序が問題とならないが，言語要素を連ねるには順序が無視出来ない。選言は「少なくとも一つ」を意味するが，言語要素を選ぶには〔掛け言葉の場合すら〕「どれか一つだけ」である。だから初めの思い着きは改めなければならなかったが，なお似通う点もあるので，それを追及して行った。

次のような模型を考えた：言語要素□は文中に単独でも使えるし，また他の要素 A, B および A と B とがこの順に結合したものを連ねても使えるとする。この承け継ぎの様は第1図（i）のように図示出来る。上から□，□-B,



第1図

□-A, □-A-B を表わし，同じ要素は縦にそろえて描いた。また縦線でくくった横線は，そのうちのどれか一つだけが実現する意である。さて□はすべてに共通だからこれをくくり出し，またそこより後に空位の認められる所は θ で埋めて，(ii) を得る。同じ考え方を進めれば(iv)に至る。ここで(v)の位置に切換えスイッチを入れて，□から他端に電気がどの道を通って流れるかを考えれば，上記の四種の連なりの形は，二個のスイッチがそれぞれ上下どちらにスイッチされているかの状態に，一意的に対応する。この事は(vi)の表から確かめられる。表では上〔下〕にスイッチすることを／〔＼〕で示した。

□の後の連なり $\{\theta, A, B, A\text{-}B\}$ において，「，」「－」をそれぞれ＋，×と解釈し，θ を１に引き当てて見よう。この時 $\{\theta, A, B, A\text{-}B\}$ は $\theta+A+B+AB$ と表わされる。これに普通に代数で行う因数分解を施すと，$(\theta+A)(\theta+B)$ を得る。この結果を第１図(v)と見比べれば，＋が選び〔言わば並列〕×が連ね〔言わば直列〕に当る。無論この加法・乗法が普通の加法・乗法と一致するか否かの検討は必要である。それを確かめた上なら，第１図でたどったような直観的な考え方によらず，式の代数的変形によって思考出来ると考えた。

(i)も(iv)も同じ内容を表わすが，承け継ぎの仕組みが見やすい点では(iv)がまさる。そこで(i)から(iv)へと代数的に導く方法は研究に価しよう。このためにはまず選び・連ねという算法の性質を調べる必要があるが，その前に，結果として描く結線図に望ましい性質をあげて置こう。

2・2　結線図に望まれる性質　結果としての図には，考察の範囲内で成り立つ連なりの形を枚挙したのと過不足ない内容を，その仕組みが一層見やすい形で，むだなく示すことが大切である。この意味で次の諸性質を備えた結線図が望ましい。なお1°と2°とは必要条件である。

1°　スイッチの一組の状態と言語要素のある連なりの形とが一意的に対応する。

2°　その状態のすべてにわたれば，あらゆる連なりの形が現れ，かつそれ以外は現れない。

3°　属性的な言語理論との並行を破らない限りで，スイッチを置く段階の数が，更にはスイッチの総個数が，最小であるようにスイッチが配置さ

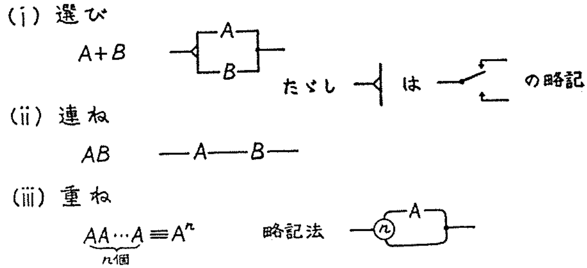
れている。

4° なるべくは，同じ言語要素〔をさす記号〕がただ一つの位置にだけ描かれている。

5° なるべくは，各要素〔をさす記号〕をつなぐ線が交わらないように，つまり平面への展開可能な形に描かれている。

3 結線図を描くための算法

3・1 基本回路 語の承け継ぎを考えるには，§2・1に述べた通り，ある位置に来る要素をそこに立ち得る要素 $\{\theta, a, \cdots, b\}$ の中から一つだけ選ぶ操作選びと，その選ばれた要素を先行の要素〔発話の頭では想定された θ〕に連ねる操作連ねとが必要である。[注2] この二種の操作には第2図の(i)，(ii)という回路が引き当てられる。



第2図

この事にはいささか注釈がいる。選びは exclusive and complete *or* であった。そういう意味の *or* は論理上，二項関係として定義しなければ扱いがあいまいになる。[注3] われわれの選びも二項選択「二つの中の一方だけ」の意味だと約束しよう。三つ以上の要素から一つを選ぶには，二項選択を繰り返すのである。〔この考え方は切換えスイッチやリレーに適する。〕 次の節で述べる通り $a+b+c$ などと書くことはさしつかえないが，意味を確定するには $(a+b)+c$ とするか，$a+(b+c)$ とするかしなくてはならない。連ねもこれに応じて二項演算と解する。なお連ねの特殊な場合には a-a-$\cdots$-a の形がある。例：面白くないな。——いや，面白くなくないよ。——どうして・面白くなくなく

2 更に，連ねの際に前の要素，時には連濁などで後の要素，更には両者が，語形を変えることもある。ただし語形変化の問題は分けて別に扱えるので，今は要素間の順序関係だけを取り上げる。

3 H. Reichenbach : *Elements of Symbolic Logic* (1947), p. 45。

ないさ。これを重ねと名づけ，第2図(iii)の図示法を採ってもよいが，重ねは連ねの一種に過ぎない。

3・2 選び・連ねの代数的性質 ここで便宜上，次の recursive な定義を与えて，項という概念を入れる：θ を含む個々の言語要素および〔§3・4に述べる〕0は項である；これらに選び，連ね，または〔§3・4に述べる〕消しの操作を施して得たもの，それらに更にこの範囲の操作を施して得たものも項である；それ以外に項はない。x と y との選び $x+y$，連ね $x\times y$ 略して xy が持つ性質を抽象的に書けば，下記の通りである。θ は先に導入した記号であるが，式(3・1),(3・2)で改めて定義される。これはある場合には零記号に相当する。こういう要素を認めても，それに関する算法が定められており，その導入法も§3・5の2′で示す通り操作的に規定されるし，また「単独では…，他と結合して…」の言い分けをせずに済むしするから，有用である。なお下記の各式をこれらに当る命題論理の公式と比べれば，性格がはっきりしよう。

(1・1)	$x+y\equiv y+x$	
(1・2)	$xy\not\equiv yx$	ただし $x, y\not\equiv\theta, 0$
(2・1)	$x+x\equiv x$	
(2・2)	$xx\not\equiv x$	ただし $x\not\equiv\theta, 0$
(3・1)	$\theta+x\equiv x+\theta\not\equiv\theta, x$	ただし $x\not\equiv\theta, 0$
(3・2)	$\theta x\equiv x\theta\equiv x$	
(4・1)	$(x+y)+z\equiv x+(y+z)$	
(4・2)	$(xy)z\equiv x(yz)$	
(5)	$(x+y)z\equiv xz+yz$	
(6)	$x(y+z)\equiv xy+xz$	

式(4・1),(4・2)は，これらの場合に括弧をはずして $x+y+z$ xyz と書いてもよいことの根拠となる。注4) また(1・2)では，連ねにおいて θ か0か以

4 音素の連ねを意味と関係させて扱う時には「二目／不為」のような例があるので，(4・2)を許さない方が安全であろう。助動詞の承け継ぎでは，そういう危険な例をまだ思いついていない。文構造の研究では「雨が降る天気じゃない。／雨が降る。天気じゃない。」の例があるが，フルとテンキとの間に，θ に似た働きをする要素として閉じ目の記号があるかないかで区別される。

外の項の交換を禁じているが，これは項の順序が無視出来ないからである。従って次に順序の問題に移らなければならない。

3・3　順序関係　選び・連ねには既成の数理が流用されなかったが，順序には数学用語がそのまま借りられる。しかも順序集合の理論はきわめて有用である。読者の大方はその方面になじみが少なかろうから，一往の紹介をすべきだが，紙幅の制約で出来ない。順序関係の定義にだけ触れて置く。なお以下に定義を省いて使う数学用語には，下に・を打って明らかにする。また論理記号として，「かつ」「ならば」にそれぞれ&と⇒とを使うことにする。

集合Aにおける順序関係≦とは，Aの元の二項関係であって

反射律　$a \leqq a$

反対称律　$a \leqq b$ & $b \leqq a \Rightarrow a = b$

推移律　$a \leqq b$ & $b \leqq c \Rightarrow a \leqq c$

を満たすものである。〔これらは数の大小関係に引き当てて考えると分りよかろう。整数を大小に従って並べれば，そこに順序が立つ。〕また $a<b$ とは，$a \leqq b$ であるが $a=b$ ではない事をさす。この $a<b$ をわれわれの問題では，番号の若い方が前という習慣に従って「a が b より前にある」と読む約束をしよう。反対に「後にある」と読んで進むことも出来，その流儀では上端・下端が直観的に分りよくなるが，他の面の不便もあるので，先のように約束する。

われわれの問題で順序づけをする集合の元となるものは，言語要素に限る。ただし順序の観点を入れた時には，「犬でしたでしょう」の二つのデスは別の元と見，また「犬である。／犬であれ。」の二つのアルのように順序に関し＝が成り立てば同じ元と見る。一方，選びなどの操作が施される集合の元には，あらゆる項が許される。この二種の集合ははっきり区別しなければならない。なお言語要素に何々を認めるかという事は，ここの立言とは全く別問題である。

3・4　演算子「消し」の導入　結線図を描くためには，選び・連ねのほかにもう一つ，操作消しを入れると重宝である。これは言語表現主体の立場でも意味があるが，研究者の立場で一層役に立つ。〔その有様は§4・4に見られよう。〕論理上「かつ…でない」に当るかも知れない演算子消し－は，算法上次の性質 (7) ないし(11)を持つものと定義される。また項0も式 (7)，(12・1)お

よび (12・2) を満たすものと定義され，論理上は「偽である」つまり成り立たないことと解せられよう。

(7) $x-x\equiv 0$ すなわち $-x\equiv 0-x$

(8) $-(-x)\equiv x$

(9) $-(x+y)\equiv -x-y$

(10) $x(y-z)\equiv xy-xz$

(11) $(x-y)z\equiv xy-yz$

(12・1) $0+x\equiv x+0\equiv x$

(12・2) $0x\equiv x0\equiv 0$

これらの式で＋と×との実質的な内容は幾分広げられるが，形式上には影響がない。

3・5 結線図を描く手続き 仕組みを調べる範囲につき，その言語系で成り立つすべての連なりの形が既知で，何々を言語要素と認めるかも定まっているとする。そうすれば結線図にまとめ上げる手続きには，次の二通りがある。第一法は項の数が少ないか仕組みが単純かの場合に有利である。すなわち

1° データの連なりの形を項としてすべて含み他は含まない式を，選び＋と連ね×とを使って書き上げる。この際新たに元 θ を導入してよいが，その結果が導入前の式と等価である事が，先の式 (3・1), (3・2) によって証明されなければならない。

2° 上述の式を§3・2, §3・4の算法によって，図示に便利な形に変える。

3° その式を基本回路の組合せから成る図に移す。

先の第１図の例ではこの手続きを踏んだ。この行き方は初等代数の因数分解同様，場合によっては特殊なくふうも必要で，一般的とは言えない。第二法は一般的な行き方である。具体的には§4 で実演する。

1′ θ を除く言語要素の間の順序関係を調べる。〔この際，こうして得た順序集合に最小(大)元があるか，極小(大)元は何々か，また適当に取った部分順序集合に下(上)端があるか等を確かめて置く。〕

2′ この順序をくずさずに下記の形の$\mathfrak{S}$式を作り，これを§3・2の算法によって展開し整頓する。

$$\mathfrak{S}\equiv\sum_{h}\prod_{i}\cdots\sum_{j}\prod_{k}(\theta+\sum_{l}a_{hi\cdots jkl})$$

ここに $\sum_{\nu}a_{\cdots\nu\cdots}$, $\prod_{\nu}a_{\cdots\nu\cdots}$ はそれぞれ ν のすべてにわたる $a_{\cdots\nu\cdots}$ の選

び・連ね。

順序をくずさないとは，Π に関する添え字を左から拾って作った数字の列を，〔こうして得た最長のものより短いものには，最長のものと桁がそろうまで右側に0を書き加え，以上の数字列が〕十進記数法による〔必要があれば，この原理による限り何進法でもよいが〕自然数を表わすと見なした場合に，次の条件を満たす事である：$a_\alpha < a_\beta \Rightarrow \alpha < \beta$.

3′ 上記の方法で$\mathfrak{S}$式から得た各項をデータの連なりの形と比べる。

もし両者に過不足がなければ，$\mathfrak{S}$式を基本回路の組合せから成る図に移す。〔この形の結線図を $\mathfrak{S}$ 図と呼ぼう。〕

もし後者にはない項が前者にあれば，これを選びでつなぎ，また更にそれを便利な形に変えて，X を作る。その上で $\mathfrak{S}-X$ を作る。次に $\mathfrak{S}$ 図を描き，または（$\mathfrak{S}-X$）式を適当に変形して得る（$\mathfrak{S}'-X'$）式による $\mathfrak{S}'$ 図〔詳しくは§4・4を見よ〕を描き，この結線を改めて X または X' の項が現れないようにする。

〔もし前者にはない項が後者にあれば，$\mathfrak{S}$式の作り誤りである。〕

なお結線図として簡潔な形が必ずしも式の上で簡潔とは言えないこと，論理回路の場合と同様である。またこうして描いた結線図が§2・2の必要条件を満たす事の証明は略す。

4 助動詞群についての実例

4・1 考察の範囲 試みとして動詞を承ける助動詞の群のいわゆる相互承接の仕組みを取り上げるが，言語要素を§4・2に示すデータの範囲に限る。言語理論としては，このような限られた範囲で，しかも助詞などを除いて扱うのは十分でない。ただこの論文の課題が方法論にあるので，今は許されたい。

4・2 データ 動詞の例としてここでは「書く」を採ろう。そしてカクを連なりの頭とする次のデータを使う注5)。

---

5 下記のデータは考察の範囲内の要素に対して，実際に話し書きする時これらの形なら使い他は使わないと意識している，筆者自身の意識に基づいて作った。ただし形容詞と言われるナイの重ねは省く。

1	かく	12	かかず	23	かかなくない
2	かこう	13	かかない	24	かかなくなかろう
3	かくだろう	14	かかなかろう	25	かかなくないだろう
4	かくであろう	15	かかないだろう	26	かかなくないであろう
5	かくなら	16	かかないであろう	27	かかなくないなら
6	かくまい	17	かかないなら	28	かかなくなかった
7	かいた	18	かかなかった	29	かかなくなかったろう
8	かいたろう	19	かかなかったろう	30	かかなくなかっただろう
9	かいただろう	20	かかなかっただろう	31	かかなくなかったであろう
10	かいたであろう	21	かかなかったであろう	32	かかなくなかったなら
11	かいたなら	22	かかなかったなら		

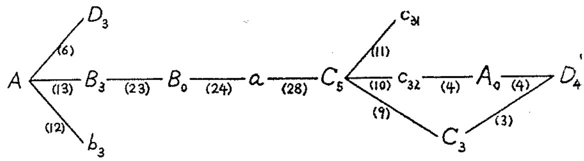
ここでは「書く。／書く(人)／書いて／書けば／書け。」等，後に助動詞を継がない場合を「かく」で代表させた。他も同様。上のデータにつき言語要素を次のように定める。〔その要素の活用形と認めるものも併記する。活用表は紙幅の制約で省略。また要素カクには記号 A を当てる。A 以外の要素の中に助動詞でないものがあるとの意見も出ようが，ともかくこの範囲を扱うのである。〕

記号	要素	活用形	備考
A_0, a	アル	ある, あら, あろ, あり, あっ, あれ	aは「なk」を承ける場合
B_0	ナイ	ない, なく, なk, なけれ	いわゆる形容詞
B_3	ナイ	ない, なく, なk, なけれ	いわゆる助動詞
b_3	ズ	ず	
C_3	ダ	だ, だろ, だっ	
c_{31}	ナ	な, なら	
c_{32}	デ	で	
C_5	タ	た, たら, たろ, たっ	連濁の「だ」を含む
D_3	マイ	まい	
D_4	ウ	う	「よう」を含む

この記号によれば，たとえば連なり31は　$AB_3B_0aC_5c_{32}A_0D_4$　と書かれる。

4・3　要素の順序づけ　集合 $S \equiv \{A, A_0, \cdots, D_4\}$ の元である各要素の間の順序関係を，§3・3にのっとって確かめよう。結果は第3図のようになる。ある要素 x から線を折れ曲らずに右へたどって要素 y に行きつければ，$x<y$ で

ある。なおその線の下にしるした番号は線の両端の要素の順序関係を定めるデータの番号である。この図が集合Sの順序関係を再現している事は，データに一々当って確かめれば分る。



第3図

4・4　$\mathfrak{S}$式から結線図へ　Aは集合Sの最小元で，これを基にして助動詞群の承け継ぎを考えるのだから，$\mathfrak{S}$式を書き上げるに当って，便宜上これを略す。この事は，集合SからAを除きθを入れた新たな集合S′を取り上げたことになる。さて注6)

$$\mathfrak{S}\equiv(\theta+B_3)(\theta+B_0)(\theta+a)(\theta+C_5)\{(\theta+c_{31})+(c_{32}A_0+C_3+\theta)D_4\}$$
$$+(\theta+b_3)+(\theta+D_3).$$

この式でB_3, b_3, D_3は順序集合S′の極小元である事に注意。次に$\mathfrak{S}$式を展開して整理する。結果は下の表の通りで，そこでは見やすいように＋を略し，第二～五列には第一列に連ねられる部分の形だけを書いた。またそれぞれの項につけた番号は先にデータにつけたのと照応する。これがない項には消しの操作を施す訳である。なおθの項が三つ出るが，式(2・1)によって一つだけが残る。

1 θ	5 $-c_{31}$	4 $-c_{32}A_0D_4$	3 $-C_3D_4$	2 $-D_4$	6 D_3
7 C_5	11 〃	10 〃	9 〃	8 〃	
a	〃	〃	〃	〃	
aC_5	〃	〃	〃	〃	
B_0	〃	〃	〃	〃	
B_0C_5	〃	〃	〃	〃	
B_0a	〃	〃	〃	〃	
B_0aC_5	〃	〃	〃	〃	
13 B_3	17 〃	16 〃	15 〃	〃	12 b_3
B_3C_5	〃	〃	〃	〃	
B_3a	〃	〃	〃	14 〃	
18 B_3aC_5	22 〃	21 〃	20 〃	19 〃	

(次のページに続く)

6　$\mathfrak{S}$式の形は本文以外のもあり得る。しかしどの形から出発しても等価な回路に達する。

23 B_3B_0	27 $-c_{31}$	26 $-c_{32}A_0D_4$	25 $-C_3D_4$	$-D_4$
$B_3B_0C_5$	〃	〃	〃	〃
B_3B_0a	〃	〃	〃	24 〃
28 $B_3B_0aC_5$	32 〃	31 〃	30 〃	29 〃

求める仕組みに対応する式を$\mathfrak{M}$，また$\mathfrak{S}$式の｛　｝の部分をPで表わそう。そうすれば$\mathfrak{S}$式と上の表とから

$$(13)\qquad \mathfrak{M}\equiv\mathfrak{S}-a(\theta+C_5)P-B_0(\theta+a)(\theta+C_5)P-B_3(\theta+B_0)C_5P -B_3(\theta+B_0)a(P-D_4)-B_3(\theta+\mathrm{B}_0)D_4.$$

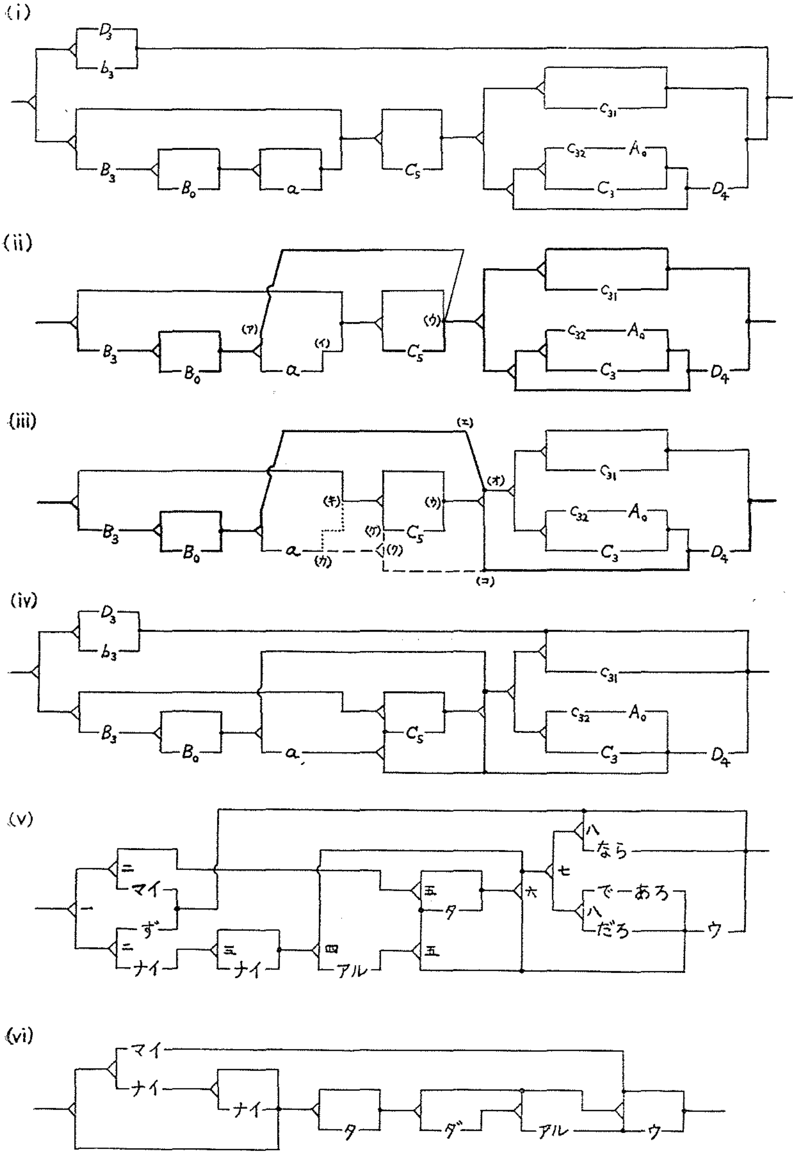
右辺第一行の二番目と三番目との項を$\mathfrak{S}$に追い込めば，

$$(14)\qquad \mathfrak{M}\equiv(D_3+b_3)+\{\theta+B_3(\theta+B_0)(\theta+a)\}(\theta+C_5)P-B_3(\theta+B_0)C_5P -B_3(\theta+B_0)a(P-D_4)-B_3(\theta+B_0)D_4$$

を得る。同様にして式(14)の第一行末項をもその前の項に追い込んでみると，その結果新たに得られる項は　$[(\theta+C_5)+B_3(\theta+B_0)\{\theta+a(\theta+C_5)\}]P$　となり，これをどう変形してもθ以外の要素が一回しか現れない形にする見込みが立たない。こうなったら，式の変形をここの段階で止め，あとは図の上で結線を改めることにする。従って$\mathfrak{S}$図の代りに式(14)右辺の第二項までを表わす$\mathfrak{S}'$図を使い〔第4図(i)〕，ここから消し演算子に導かれる項を消して行く。

まず(14)式第一行の末項を消す。このためには，第4図(ii)に示したように(ア)から(イ)への線を取り去り，その代り(ア)から(ウ)につなげばよい。これで末項が消せた事は，それに当る道筋が図の太線に見る通り切れていて，導通がない事で確かめられる。〔なお図の(ii)，(iii)ではD_3+b_3を略して描いた。〕次に式(14)の第二行末項を消そう。これには少々細工がいる。その前の項と見比べると，$B_3(\theta+B_0)$の承け方がD_4とそれ以外とで異なり，言わば相補う様をしている。そこで(ウ)の後に三段にはいっているスイッチの組合せ方を，これに応じて(iii)のように改める。その上で(エ)から(ウ)につないだ線を(オ)につけ替える。これで$B_3(\theta+B_0)D_4$が消せた事は，前と同様にして分る。[注7)] 残った項を消すには，(カ)〜(キ)の線を，新たに入れたスイッチ(ク)を介して(ケ)と(コ)とにつなげばよい。以上の結果を清書すると(iv)になる。これが式(14)に対応する結線図であ

7　スイッチの状態にかかわらず，上下の向きにスイッチをまたいでは電気が流れない。



第4図

る事は，上述の描き方からして明らかである。

なお言語理論としての美しさの点では，(iv)と等価な回路(v)の方が整っていると言えよう。ただしこれに対応する式は(14)より複雑な形を取る。〔その式は略す。〕念のため（v）が前述のデータにおける仕組みを表わす事を，第1図(iv)と同様な仕方で示そう。第4図(v)に漢数字で示したのは，スイッチの段階である。同段のスイッチは何個あっても，リレーを使えば連動する事に注意。また下の表のアラビア数字は，その行のスイッチの状態が代表する連なりのデータ番号である。行中の空欄は，上下どちらにスイッチされていてもよい事を表わす。

	一	二	三	四	五	六	七	八
1	／	／			／	／	／	／
5	／	／			／	／	／	＼
4	／	／			／	／	＼	／
3	／	／			／	／	＼	＼
2	／	／			／	＼		
7	／	／			＼	／	／	／
11	／	／			＼	／	／	＼
10	／	／			＼	／	＼	／
9	／	／			＼	／	＼	＼
8	／	／			＼	＼		
6	／	＼						

	一	二	三	四	五	六	七	八
12	＼	／						
13	＼	＼	／	／			／	／
17	＼	＼	／	／			／	＼
16	＼	＼	／	／			＼	／
15	＼	＼	／	／			＼	＼
18	＼	＼	／	＼	／	／	／	／
22	＼	＼	／	＼	／	／	／	＼
21	＼	＼	／	＼	／	／	＼	／
20	＼	＼	／	＼	／	／	＼	＼
19	＼	＼	／	＼	／	＼		
14	＼	＼	／	＼	＼			

	一	二	三	四	五	六	七	八
23	＼	＼	＼	／			／	／
27	＼	＼	＼	／			／	＼
26	＼	＼	＼	／			＼	／
25	＼	＼	＼	／			＼	＼
28	＼	＼	＼	＼	／	／	／	／
32	＼	＼	＼	＼	／	／	／	＼
31	＼	＼	＼	＼	／	／	＼	／
30	＼	＼	＼	＼	／	／	＼	＼
29	＼	＼	＼	＼	／	＼		
24	＼	＼	＼	＼	＼			

ついでながら，要素 a を立てず，「なかろ，なかっ」をナイの，また「で，なら」もダの活用形と認め，更に「ず」をも助動詞ナイの活用系列に組み込めば，どうなるか，この時は式(14)から $\theta+a$ の項を取り去り，$c_{31}\equiv c_{32}\equiv c_3$，$b_3\equiv B_3$ と置いて，同様な手続きを踏めばよい。結果は第4図（vi）である。ここに文法学説の相違が結線図の上に目で見える形に示された。(vi)は(v)より簡潔であるが，その代り連ねの際の語形について但し書が増す。

5　結語

以上で課題について一通り論じ終えた。この理論の完成のためには，更に二つの仕事が残っている。一つは，扱う言語現象の範囲を広げ，それにこの方法を適用して経験的な有効さを確かめることである。もう一つは，この稿での論述がまだ粗いから更に論理的に整え，算法に公理論的な裏づけを与えること，

また結線図の最適条件などを形式的に明らかにすることである。

最後にこの方法による所産の，言語理論への寄与について事細かに述べたいが，許された紙幅が尽きているので，読者の賢察に待つ。

〔追記〕　算法の公理論的検討を，本稿の提出後に進めたところ，§3・2の式(2・1)を無条件に他の式と同列に置くのはよくない事が分った。更に整備した形で算法を示すべきであるが，既にこの論文集の編集が終った模様なので今その部分を書き替える訳には行かない。この問題をめぐっては述べるべき事も多いから，別の機会に詳しく示すことにする。

ただし今その見通しを述べて置けば，ここで扱った代数系は，乗法の単位元を有し，乗法について非可換な環をなすものとして基礎づけられると思う。更にこの代数系に順序の概念を導入するに当っては，なお考えるべき点があるが，筆者はこれを十分に明らかにするには至らない。かような代数系の理論は，既成の数学では取り上げていないと思うので，今後の問題となろう。

また式(2・1)の使用を禁じてしかもこれを使ったと同等の効果をあげる事についても，試案はあるが，まだ発表の段階ではないので保留する。

以上のように論理的に整備されてはいない点が出て来たが，その事は本稿の着想を根本からゆすぶるものではなく，また§4に実演した通り言語事実の再現に成功を収めているから，そうした不備は今後の努力で取り除いて行きたい。