

CLL Version 1.1 におけるlujvoが一意に分解できることの証明

fi'e la .sozysozbot. (<http://twitter.com/sosobotpi>)

-1章 注

以下の文章は、「2017年5月1日現在 <https://goo.gl/fGZvAE> にて一般公開している、lujvo作成アルゴリズムの解説」の補足文書である。0章～3章は上記リンク上にある。以降、文章中で断りなく「lujvo作成アルゴリズムの解説」の内容に触れることがある。

4章 一意性の証明

以下、「(0個以上のcmavo) + lujvo」が「(0個以上のcmavo) + 2個以上のrafsiの列」に一意に分解できることの構成的な証明を行う。なお、以下C\Cで「語頭に立てない二重子音」を表す。

4-1 lujvoの先頭部分に関する考察

まず、3章の全パターンを列挙して調べることで、lujvoの先頭部分としてありうる形が限られることが分かる。（2番目のrafsiが何であれ、必ずCで始まり、その次がyでないことは保障されていることに注意）
便宜上、それぞれの形に番号を振り、先頭のrafsiを強調表示する。

パターン	形
case1	① CCVCyC - ② CVC/CyC - ③ CCVC (y以外の文字)-
case2	④ CVVCCV
case3	⑤ CVVrC (y以外の文字)- ⑥ CVVnr (y以外の文字)-
case4 (2-2が適用されるとき)	⑦ CVCyC (y以外の文字)-
case4 (3-1が適用されるとき)	⑧ CVCyCVCCVC ..CVCCV ⑨ CVCyCVC ..CVCy-
case4 (それ以外のとき)	⑩ CVC/C (y以外の文字)-

これにより、まず、2-1で書いた主張「lujvoの2文字目には絶対にyが来ない」が示され、また「lujvoの3文字目にも絶対にyが来ない」ことも分かる。また、異なる第一rafsiに由来するlujvoの形が重複することがなく、lujvoの形から一意に第一rafsiが判定できることがわかる。

さらに、⑤⑥⑧⑨では第一rafsiの後ろにハイフンが挿入されているが、このハイフンが仮に存在しなかったとしても他種のrafsi列と「は」混同することがない。⁸ゆえに、第一rafsiを取り除いた後の文字列について同様の議論を行うことで⁹、lujvoを構成するrafsi列は一意に決定できる。

4-2 cmavo: 0個 vs. 1個以上

次に、「lujvo」が「(1個以上のcmavo) + lujvo」と一致することはないことを示す。そのために、一致する例が存在するとして矛盾を導く。

一致する例が存在するならば、4-1より、「(1個以上のcmavo) + lujvo」と解釈した時のcmavoの形としてありうるのは1つのCV (②⑦⑧⑨⑩) か1つのCVV (④⑤⑥) である。しかし、②④⑤⑥⑦⑧⑨⑩のいずれの場合でも、「cmavo + lujvo」との解釈は成り立たないことを示す。

②④⑤⑥⑦⑧⑨の場合は単純である。

パターン	理由
②	CVを取り除くと3文字目がyで、これが4-1の結論と矛盾
④	CVVCCVのCVVの内部にはアクセントがあるので、cmavoにはなれない
⑤⑥	CVVを取り除くとlujvoはrCまたはnrで始まることとなるが、これはどちらもCCではない
⑦⑧⑨	CVを取り除くと2文字目がyで、これが4-1の結論と矛盾

さて、問題は⑩である。⑩の先頭からCVを取り除いた形がlujvoとなる可能性を否定しなければならない。⑩は、3-1のtosmabruテストに該当しない場合なので、

- 1. [(1個以上のCVC) + CVCCV] であるが、くっつけた時一部の子音結合がC\Cである
- 2. [(2個以上のCVC) + y + (1個以上の任意のrafsi)] であるが、y以前の一部の子音結合がC\Cである
- 3. y を含まず、[(1個以上のCVC) + CVCCV] でもない
- 4. y を含んでいるが、[(2個以上のCVC) + y + (1個以上の任意のrafsi)] ではない

という4通りの場合があり得る。

まず1.の場合、先頭からCVを取り除くとCCVCCV..C\CV..CVCCVといった形になる。左から1つずつrafsiを削っていくとき、4-1.よりその削り方は一意に決定できる（CCVという形で必ず削られる）が、すると途中でC\CV-という形に遭遇してしまう。当然これはrafsiとして許されないので、こう分解することは許容されない。2.も同様であり、CCVCCV..C\CV..CCVCy-という形を左から削っていくとC\CV-に遭遇してしまう。

3.の場合、元々の分解方法では途中で「CVCCVでも4文字rafsiでもCVCでもないrafsi」に遭遇することとな

る。そのrafsiの左には1つ以上のCVC rafsiがあるので、lujvoの先頭からCVを削った場合

- ⑪ **CVC/CVV-** [CVVが後続する時]
- ⑫ **CVC/CCV-** [CCV, CCVC, CCVCVが後続する時]
- ⑬ **CVC/CVC\CV** [CVC\CVが後続する時]

から先頭のCVが（直接的に、またはCCVCCV..CCVの最後2文字として吸収されて）失われた場合にrafsi列として成立しないことを示せばよい。まず、上での3文字目と4文字目の2つのCがC\Cである場合、先頭のCVを失うとrafsi列と見ることはできない。次に、CCVV-, CCCV-で始まるrafsi列はないので⑪⑫は除外でき、CCVC\CVという形はCCVを削るしかなくC\CVが露出し不適。以上より3.も否定できる。

最後に4.の場合。2.の場合を除いていることから、[CVC + (CVCが0個以上) + (CVC以外が1個以上) + (CVCが0個以上) + **y** + (1個以上の任意のrafsi)]として表せる。つまり、「CVC+CVV」「CVC+CCV」

「CVC+CCVC」「CVC+CVC/C」のいずれかの組み合わせが列中に存在するが、最初の3つは先頭のCVが直接的または間接的に削られた際CCVV-またはCCCV-というrafsi列としてあり得ないものとなり、

「CVC+CVC/C」は2-1より「CVC+CVC/C+ **y**」であり、先頭のCVが消えるとCCV + C/Cy-というあり得ない形となる。

以上より、「lujvo」が「(1個以上のcmavo) + lujvo」と一致することはないことが示された。

4-3 最後に

最後に、2種類の「(0個以上のcmavo) + lujvo」が一致することはないことを示す。「(m個以上のcmavo) + lujvo1」からなるAと、「(n個以上のcmavo) + lujvo2」Bから導き出される文字列が一致したと仮定して矛盾を導く。

まず、m=nの場合を考える。この場合、明らかにcmavo列は一致していて、ゆえにlujvo1とlujvo2も一致する。次にm>nのとき、A,Bの先頭から共にn個のcmavoを削ると「((m-n)個のcmavo) + lujvo1」と「lujvo2」が一致することになるが、これは4-2に反する。AとBを入れ替えればm<nの場合も示せる。

以上より、一意性が示された。

5章 おまけ1: lujvoの得点の話

lujvoには「得点」が割り振られている。同じrafsi列から作られたlujvoの中でも、得点が最も低いものが「好ましい」（原文: preferable）とされている。

得点の計算式自体はCLL4.12に載っているが、厳密に1の位まで得点を計算する必要があることはほとんどなく、以下の目安を覚えておけば十分であろう。

1. 短いlujvoが勝ち（ただし、アポストロフィは0.5字相当）
2. 引き分けならハイフン字（**y**, **n**, **r**）が少ない方が勝ち

3. それでも引き分けならCVC < CCV < Cw

6章 おまけ2: CLLの自己矛盾

公式の説明CLL4.6は自己矛盾を含む。というのも、

Each gismu always has at least two rafsi forms; one is the gismu itself (used only at the end of a lujvo), and one is the gismu without its final vowel (used only at the beginning or middle of a lujvo).

という記述ををbroda, brode, brodi, brodo, broduに当てはめれば、-brod-はこの5種のgismuを表すrafsiになるはずだが、そうするとその直前にある

each such rafsi represents only one gismu.

に矛盾するからだ。

ちなみに、CLL4.15を見ると、

The following gismu were not made by the gismu creation algorithm.

という文章の後にbrodVが紹介されているので、brodVが公式見解上gismuであることは確実である。詳しくは、<https://goo.gl/zM7M9u> を参照のこと。

脚注

8: 後述する通り、万が一ハイフンがない場合cmavo + lujvoと「は」混同することに注意

9: ただし、lujvo末には5文字rafsiが登場するという点に留意する必要がある