文書に最適化されたキー配列の生成

▼ 研究の成果

- アルファベットからなるドキュメントに最適化されたキー配列の生成を行うプログラムを作成した。
- 生成されたキー配列は、最適化に用いた文書と同類の文書に対して高い性能を発揮した。

▼ 研究用に開発したプログラム

https://github.com/yuichiro-kurose/keyboard

情報分野　１１班　２年８組１１番　黒瀬　諭一朗

各キーの使用頻度（日本語 -QWERTY ）

Q W E R T Y U I O P
A S D F G H J K L _
Z X C V B N M _ _ _

各キーの使用頻度（英語 -QWERTY ）

Q W E R T Y U I O P
A S D F G H J K L _
Z X C V B N M _ _ _

各キーの使用頻度（日本語 - 大西配列）

Q L U _ _ F W R Y P
E I A O _ K T N S H
_ Z X C V G D M J B

赤：最高
黄：高
緑：やや高

先行研究の大西配列では、押しやすい場所によく使うキーがある。

大西配列の制作過程

1．片手の**連続使用**を減らす

▶ 左右交互打鍵で高速化

2．指の**移動**を減らす

▶ 速く楽にタイピング

3．指の**連続使用**を減らす

▶ 流れるようにタイピング

作成したプログラムの概要

1．ドキュメントを読み込む

2．右手と左手の担当キーを決定（グラフ理論を応用）

3．押しやすい場所に使用頻度の高いキーを配置（先行研究の○等地を使用）

4．同じ指の連続使用を回避

5．完成したキー配列を出力

大西配列の制作過程に沿って最適なキー配列を生成するプログラムを作る

右手と左手の担当キーを決定（本研究の目玉！！）

キーを右手と左手に分ける方法は、

$$ {}_{26}C_{13} = 10400600 \fallingdotseq 10^7 \text{（通り）} $$

片手を連続使用した回数が最小となる分け方を見つけるためには、それぞれの分け方における片手の連続回数を数える必要がある。

これを愚直にやると、 10^7N 回の処理が必要。（ N はドキュメントの文字数）

これだと $N = 10^6$ のとき、普通のパソコンでは**約 1 日**の時間を要する。

※ ホームリーダー 10 冊で、約 10^6 文字

▶ **片手の連続回数を事前に処理**することで効率化を図る

頂点集合を {a,b,...,z} とする重み付き完全無向グラフ K_{26} を考える。ただし、

w(u,v) = 「ドキュメント内で uv または vu が現れる回数」

とする。例えば、ドキュメントが「Yuichiro Kurose」のとき、

$$w(a,b) = 0,\ w(k,o) = 1,\ w(o,r) = 2$$

となる。

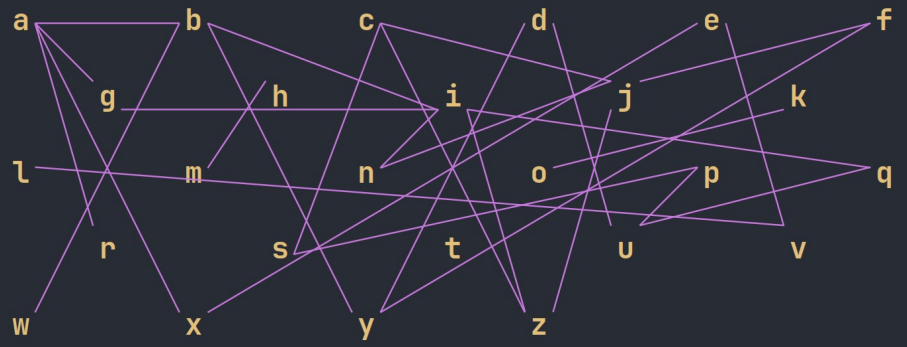
片手の連続使用回数が最小となる分け方は、各グループ内の辺の重みの合計が最小となるように K_{26} を分割する方法に対応する。

この手法では、K_{26} の構築に $N + 26^2$ 回、分割に ${}_{26}C_{13} \fallingdotseq 10^7$ 回、各分割方法における重みの総和の計算に ${}_{13}C_2 = 78$ 回の処理が必要であるから、

（全体の処理の回数）$\fallingdotseq N + 26^2 + 10^7 * 78 \fallingdotseq N + 10^9$

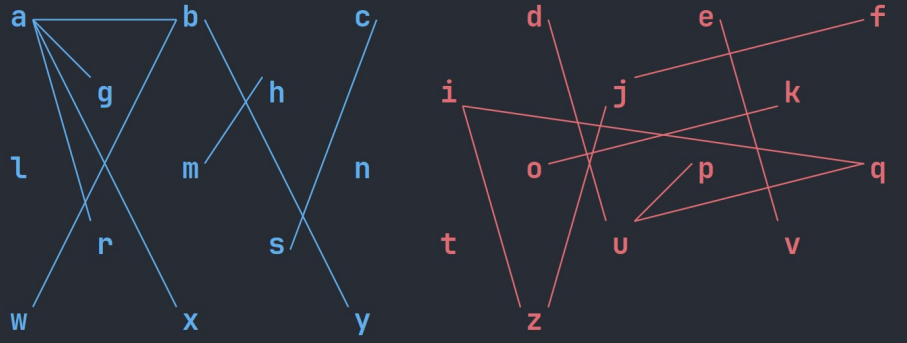
となる。N が 10^9 以下のとき、普通のパソコンでは**約 10 秒**で処理が完了するため、愚直な手法と比較して劇的な高速化を遂げている。

K_{26} のイメージ図



※　実際は全頂点間が結ばれている

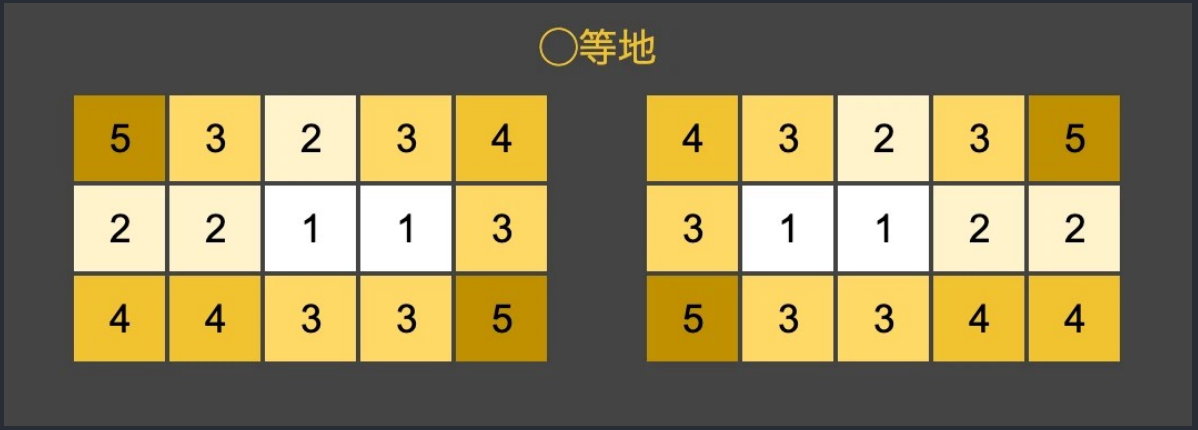
分割後のイメージ図



※　実際は各グループ内の全頂点間が結ばれている

押しやすい場所に使用頻度の高いキーを配置

押しやすさの評価には、大西氏が開発した「◯等地」を使用。◯等地によって定義されたコストを最小化することで、押しやすい場所に使用頻度の高いキーを配置することを実現。



キー配列の生成と評価の手順

1. A Christmas Carol の序盤（約 30000 文字）を用いて、キー配列を生成
2. Dr. Jekyll and Mr. Hyde の序盤（約 10000 文字）を用いて、 QWERTY と比較

QWERTY と太郎（仮）の比較

同じ手の連続使用率

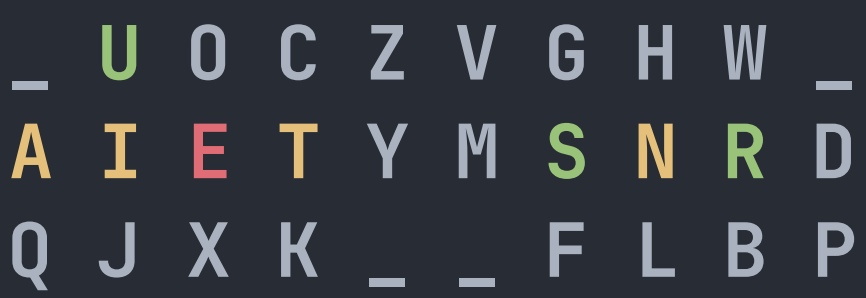
47.8% ▶ 34.2%

同じ指の連続使用率

10.5% ▶ 6.4%

各キーの使用頻度（英語 -QWERTY ）

各キーの使用頻度（日本語 - 太郎（仮））



結論

- アルファベットからなるドキュメントに最適化されたキー配列の生成を行うプログラムを作成した。
- 生成されたキー配列は、最適化に用いた文書と同類の文書に対して高い性能を発揮した。

課題

- 実際に人間にとって使いやすいかは不明。
- QWERTY からの乗り換えが困難である可能性が高い。

展望

- 文書をアルファベットに変換するプログラムを用いれば、アルファベットでない文書にも対応できる可能性が高い。
- 記号に対応できれば、プログラミングの効率化が期待できる。

参考文献

[1] 大西拓磨「ローマ字入力に最適なキー配列を考える（制作編）」
https://note.com/illlillliliill/n/n3b51f4aaf086

[2] Charles Dickens「A Christmas Carol in Prose」
https://www.gutenberg.org/cache/epub/46/pg46.txt

[3] Robert Louis Stevenson「The Strange Case of Dr. Jekyll and Mr. Hyde」
https://www.gutenberg.org/cache/epub/43/pg43.txt

[4] 秋葉拓哉・岩田陽一・北川宜稔『プログラミングコンテストチャレンジブック』株式会社 マイナビ出版