ISS-A-038

焼きなまし法を用いた大きな覆面算の生成

浦川 大輝 †　　山本 修身 ‡
名城大学 理工学部 †　　名城大学 情報工学部 ‡

1　はじめに

覆面算は，筆算について 0 から 9 までの各数字をそれぞれ対応する別の文字に置き換えられたものが与えられ，それを元の筆算に復元する計算パズルである．覆面算の例を図 1 に示す．以降，覆面算の文字で構成された筆算を「問題」と呼び，問題から復元される数字で構成された筆算を「解」と表す．問題を解く際の制約に「同じ文字には同じ数字が入る」，「異なる文字には異なる数字が入る」，「左端の文字に 0 は入らない」がある [1]．

本研究では，図 1 のような複数の英単語で構成された加算の筆算を扱う．覆面算の問題を作成するうえで英単語数を増やそうとすると，問題全体に使用される文字の種類数は 10 より大きくなり，問題として成立しなくなりやすい．本研究の目的は，問題に使用する単語数を 3 より大きくして，より大きな覆面算を作成することである．

本研究は焼きなまし法 [2] を用いる．焼きなまし法はある状態から局所探索をランダムに行いながら，より良い解が見られない場合でも新しい状態に移ることで局所最適解ではなく最適解を得るアルゴリズムである．

2　覆面算の生成処理アルゴリズム

覆面算の問題には任意の英単語群から選ばれた単語を使用し，以降，この単語群のことを「英単語リスト」と表し，記号は S とする覆面算生成処理の流れを図 2 に示す．まず任意の単語数を設定し，S から単語をその数だけ抽出し，文字数の少ない順にソートする．以降，設定した単語数を n とし，ソートしたものを $\{w_1, w_2, \cdots, w_n\}$ と表す．次に $\{w_1, w_2, \cdots, w_n\}$ に使用される文字の種類数が 10 以下であり，w_{n-1} と w_n の文字数の差が 1 以内のときそれを問題候補とする．最後に $w_1 + w_2 + \cdots + w_{n-1} = w_n$ を解き一意解を持てば，それは覆面算であり出力する．

S からランダムに n 単語取り出し，それが問題候補になるか逐一確かめる方法は現実的ではない．焼きなまし法により，取り出した n 単語から新しい n 単語を確率的に生成する．評価値は，文字の種類数が 10 のとき最大になるように設定し，状態遷移時に入れ替えられる単語は，ランダムに選ばれた単語または最も使用頻度の低い文字を含む単語とし，新しく入る単語は S からランダムに選ばれたものとする．

S E N D	9 5 6 7
＋M O R E	＋ 1 0 8 5
M O N E Y	1 0 6 5 2

図 1　覆面算の問題（左）とその解（右）．

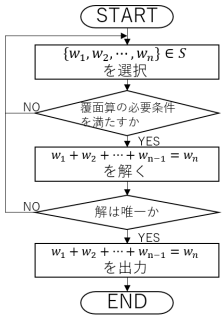

図 2　覆面算生成処理の流れ．

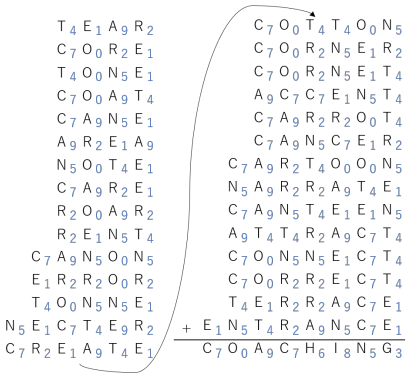

図 3　単語数 30 の覆面算．

3　アルゴリズムの実行結果

本研究は CEFR-J Wordlist[*1]の 4 文字以上の名詞と動詞，合計 4,524 単語を十分問題を作成できるため S として使用した．ランダムに単語を取り出して逐一確認する方法，焼きなまし法において，ランダムに選ばれた単語を入れ替える方法，最も使用頻度の低い文字を含む単語を入れ替える方法のそれぞれで 100 万回試行した．[*2]$n = 3$ のときは先 2 つの手法の方がより多くの問題候補を生成でき，$n \leq 4$ のときは焼きなまし法で特定の単語を入れ替える手法の方がより多くの問題候補を生成できるとわかった．また生成した問題候補についてそれぞれを覆面算ソルバーにかけることで，$n = 3$ から 30 までの覆面算の生成に成功した．$n = 30$ の覆面算は図 3 に示す．

4　まとめ

使用する文字は 10 種類で英単語リストから単語を抽出し，$n \leq 4$ のとき，最も使用頻度の低い文字を含むの単語を入れ替える手法が最も問題候補の生成数が多くなることがわかった．$n = 30$ までの覆面算の生成に成功した．

参考文献

[1] 大槻 兼資：パズルで鍛えるアルゴリズム力．技術評論社，2022 年 5 月．
[2] 池田 智悟，窪田 耕明：シミュレーテッドアニーリング概説．同志社大学，知的システムデザイン研究室第 29 回月例発表会資料，2000 年 4 月．

On generation of large size alphametic puzzles using simulated annealing
† Taiki Urakawa, Meijo University
‡ Osami Yamamoto, Meijo University

*1 『CEFR-J Wordlist Version 1.6』東京外国語大学投野由紀夫研究室（URL:http://www.cefr-j.org/download.html#cefrj_wordlist より 2022 年月ダウンロード）．
*2 使用した PC の CPU は Intel(R)Core(TM)i7-8565U であり，使用言語は JavaScript，実行環境は Node.js である．