

中規模なクロスワードパズルの作成方法

関東職業能力開発大学校 加部 通明

A Way of Creating Crossword Puzzles on a Middle Scale

Michiaki KABE

要約 CWP (CrossWord Puzzle: クロスワードパズル) は雑誌、新聞、広報誌等の紙面でよく見かけられる最もポピュラーなパズルである。一般的には、問題が与えられていて、枠に埋込む単語のヒントを基にして単語が交叉する部分の整合性をとりながら、全ての枠に文字を埋込むパズルである。

今回、本稿で取り上げるCWPは与えられた問題を解くのではなく、CWPの問題を作成する方法を考えて、それをプログラミングし実際にパソコン上でCWPの問題を作成しようとするものである。

CWPは理論上制約充足問題の一種である。それ故、この問題を解くアルゴリズムはバックトラック法として存在している。しかし、枠の規模が例えば10×16になると、不定型な枠では所謂「組み合わせ爆発」を起こしてスーパーコンピュータでもバックトラック法を利用して実時間で問題を作成するのが困難となる。

過去に中規模な枠10×16の英語CWPの自動作成をGA (Genetic Algorithm: 遺伝的アルゴリズム) を応用して考察してきたが、完成させることができなかった。そこで、今回同じ大きさ規模のCWPを作成する際に、いきなり最初から100%パソコンによって自動的に作成するのではなく、90%程度をGAによって作成し、それを基に制約を満たす状態に手直しして残りの10%程度を再度GAによって完成させる方法について報告する。

I はじめに

CWPは今日多くの人達に知られている世界規模で普及しているパズルである。当然文字文化が発達したところならどこにでもありと考えられる。我が国では、CWPを専門に取り扱った雑誌が月に30種類ぐらい^(注1)発行されているくらい人気のあるパズルである。

CWPは文字を記入できる枠とできない枠とから構成されており、全体の形はいろいろあると考えられるが、本稿では図1に示すような長形状に枠が配置されているCWPを取り上げる。

文字はある長さで区切られその長さの文字をつなげると意味のある単語となる。その単語が意味のあるものであるか否かは、その単語が辞書に登録されているか否かによって判定される。

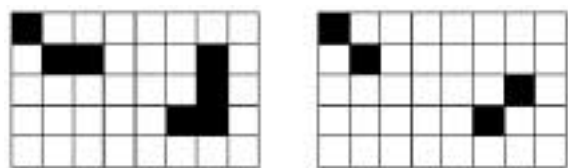
以下、一般のCWPの書き方によって、文字記入枠

を白枠、文字非記入枠を黒枠と呼ぶことにする。そこで、この黒枠を中心に上げて、文字が英語の場合と日本語の場合とを比較して、英語CWPと日本語CWPの違いを考えてみよう。

日本語CWPは黒枠が縦横に連結してなくて孤立しており⁽¹⁾、単語の字数に依らず単語数が豊富である。一方英語CWPは黒枠が縦や横に連結している。また、黒枠全体がある図柄を表したり、中心点に関して対称だったりするものがある^(注2)。そして、字数が少ない単語は個数が少ない。例えば、中学高等学校で習得すべき英単語を中心に考えると、2文字の単語は30個ぐらいしかなく、3文字以上の単語数が300個以上あるのに比べて極端に少ない。従って、外国の英語CWPでは、枠に入れる文字対象を3文字以上の単語に制限しているものが多い。

本稿で取り扱う英語CWPは、日本語CWPと英語

CWPの折衷で2文字以上の単語を対象に黒枠が縦や横に連結しないもの考えることにする。



(1) 英語 CWP (2) 日本語 CWP

図1 長方形状5×8のCWP

II CWPの作成方法

1 数学的な解釈

CWPは単語の文字が重なるという制約を持った順列問題の一種であるが、表現としては組み合わせ問題という言葉がよく使われている。本稿では、CWPを従来から考えられている次のような制約充足問題⁽³⁾として捉えることにする。

[記号の意味]

X_k : 辞書に登録されている1つの単語。

W_k : 辞書に登録されている文字数kの単語の集合。

P_k : 単語を埋込むために配置された横又は縦のk番目の単語枠。

P_{mn} : 二つの単語枠 P_m と P_n の重なる位置にある枠。

$L()$: 単語枠の長さ又は単語の文字数。

$P_m(X_m)$: 単語枠 P_m に単語 X_m を埋込んだとき、ぴったり収れば真、でなければ偽となる真理関数。

即ち、

$$L(P_m) = L(X_m) \Rightarrow P_m(X_m) = 1,$$

$$L(P_m) \neq L(X_m) \Rightarrow P_m(X_m) = 0.$$

$P_{mn}(X_m)$: $P_m(X_m)=1$ で枠 P_{mn} に入る単語 X_m の文字。

$P_{mn}(X_m, X_n)$: 2つの単語枠 P_m と P_n が重なる枠で、それぞれの枠に当て嵌まる単語 X_m と X_n との文字が一致すれば真、しなければ偽となる真理関数。

即ち、

$$P_{mn}(X_m) = P_{nm}(X_n) \Rightarrow P_{mn}(X_m, X_n) = 1,$$

$$P_{mn}(X_m) \neq P_{nm}(X_n) \Rightarrow P_{mn}(X_m, X_n) = 0.$$

ところで、

$$P_m(X_m) \cdot P_n(X_n) = 0 \Rightarrow P_{mn}(X_m, X_n) = 0,$$

$$P_{mn}(X_m, X_n) = P_{nm}(X_n, X_m),$$

である。以上の意味から、CWPとは

$$(\exists X_1 \in W_{p_1}) \cdots (\exists X_n \in W_{p_n}) : \bigwedge_k P_k(X_k) \wedge \bigwedge_{mn} P_{mn}(X_m, X_n)$$

(ただし、 $p_k = L(P_k)$)

が真であるような X_1, \dots, X_n を求める制約充足問題と解釈できる。

2 作成の方針

CWPは、枠の大きさ規模が5×5程度の小さい時には、バックトラック法又は人間の勘や経験によって少ない試行回数で問題を作成することができる。従って、枠の規模が大きいCWPを作成するには、それを規模の小さいCWPに変形させて完成させる方法が考えられる。

今回のCWP作成には埋込み型の方法を利用して⁽¹⁾、先ず初めにGAを応用し90%程度の完成度まで持っていく。即ち、単語埋込み規模を当初の規模の1/10程度の量のCWPにする。その後、制約を満たす単語のみを残し、制約を満たさない単語を消去して、制約を満たしている単語が埋込まれた状態を新たなCWPの初期状態とする。そして、それを基に残りの10%程度を同じくGAを応用して全体の完成を目指す。

埋込み型のCWP作成では、次のような手順を行う⁽¹⁾⁽²⁾。その全体の流れを図2に示す。

- ①単語枠の作成。
- ②単語枠の選択（連結性、重複率、存在率が選択の尺度）。
- ③単語の単語枠へ埋込み。

今回実際にCWPを作成する段階では、既に①が終了して、更に辞書が存在していることを仮定している。

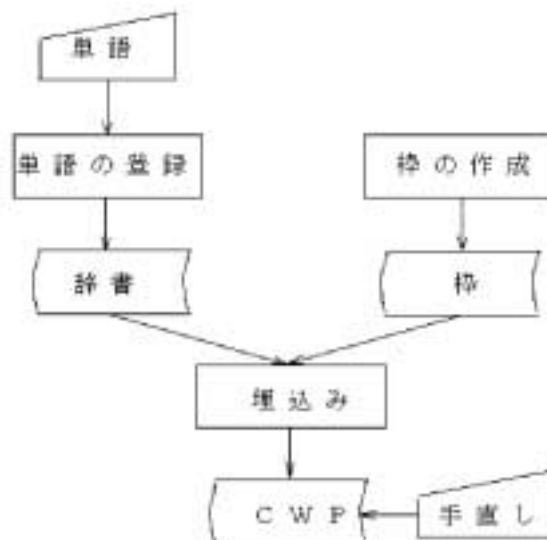


図2 CWP作成の流れ

3 GAとは

生物の進化過程にヒントを得た基本原理（自然淘汰、遺伝子の交叉、突然変異の発生）を基にして、工学的

にモデル化した染色体をこの基本原理により世代交代させて、染色体の進化により最適化問題では最適解を求めようとするものである⁽⁴⁾。また、今回取り上げる制約充足問題では、制約違反数を1つずつ減らして解の発見を行うのにGAが応用される。

今回使用したGAは基本的な形のものを採用した。即ち、2個の親により2点交叉を行い、世代交代では各染色体の適応度の大きさに比例した確率で次世代に残し、かつ最大適応度の染色体を必ず1つ残すというエリート保存・ルーレット選択である。

4 GAのCWPへの応用

まず、使用するGAの項目がCWPのどの項目に対応するかを定義し、その後、単語枠に単語を埋込む時にGAをどのようにCWPへ応用するのかを具体的な操作で説明しよう。その操作内容を図3に示す。

[項目の定義]

遺伝子：辞書に登録されている単語に割り付けられた一連の番号。

染色体：ある規準で選んだ枠順に埋込まれた単語の番号を並べた番号列。

適応度：制約条件を満たしている単語の個数を総単語枠数で割った数値で、1以下の非負数^(注3)。

[GAの操作]

(1) 初期処理

重複率の高い順に選んだ単語枠に埋込み操作を行い、16個の親の染色体を生成する。

(2) 子の生成

16個の親から交叉させる2つの親を8組作り、1つの組の親に対して2点交叉で交配させて子を2つ作る。その際、致死遺伝子を生じないように、同じ単語番号が出現したときには2番目の番号を乱数を用いて $P_k(X_k)$ が真となるような X_k を探す。(図4)

(3) 突然変異

親1から親7までに突然変異を次の発生方法で説明する方法により起こさせる。

(4) 適応度計算と淘汰

全ての染色体の適応度の計算を行いそれを基にして、最大適応度の染色体1個をエリートとして必ず保存し、残りをルーレット選択により親1から親15までを決定する。(図5)

(5) 終了判定

操作(2)(3)(4)を規定回数を超えるまで繰り返して行う。

[突然変異の発生方法] (図6)

①突然変異を起こさせる遺伝子座の対象 (全体の1/2)

を乱数によって5個決定する。

②①で決定された各遺伝子座に対して、遺伝子座を持つ単語枠情報によりその単語枠と同長の未使用単語の候補を乱数で選出する。

③②で選出された単語が制約条件を満たすか否かを判定し、もし満たさなければ突然変異を中止する。

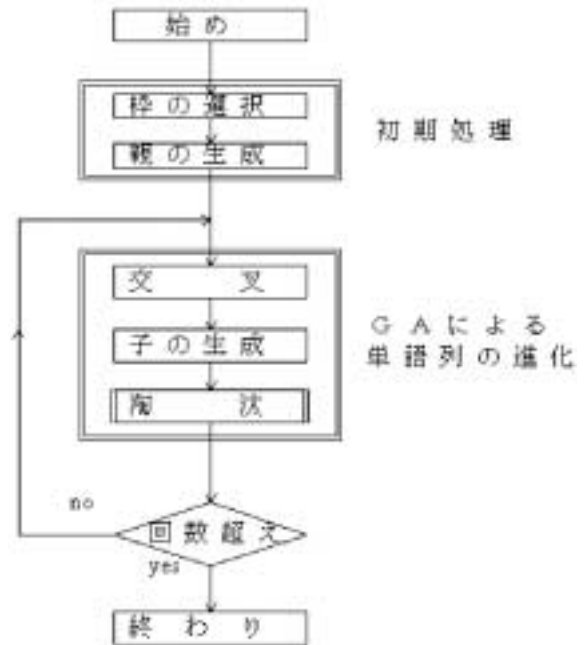


図3 GAによる単語の埋込み



図4 遺伝子の2点交叉と致死遺伝子の解消法

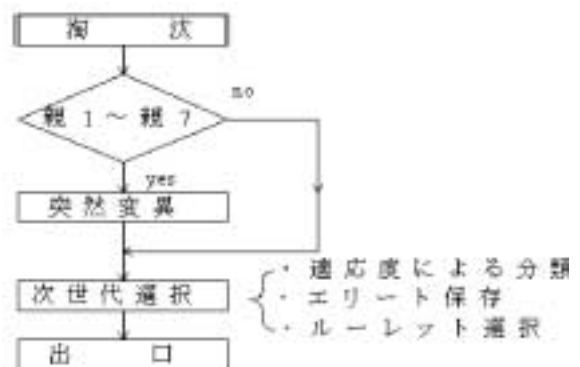


図5 淘汰と突然変異

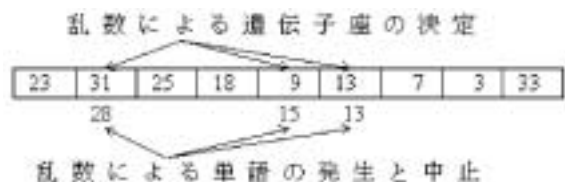


図6 突然変異の発生方法

III システム実験

1 実験内容

パソコン上でCWPを作成する場合、パソコン画面を目一杯使用するとすればCWPの形状はパソコン画面の制約から自ずと決まってくる。つまり、普通パソコン画面自体は縦横の比が3:4であるが、GUI環境を考慮すると、実際に使用できるワークエリアの形状は縦横の比がおおよそ5:8の長方形になる。よって、枠の大きさが縦横5×8の長方形の辺を整数倍した規模のCWPを考えることにする。

今回は枠の大きさ規模が10×16で表1に示すような内容のCWPを当初作成することを試みる。その後、II章で説明したCWPの作成方法に従って、その完成を目指す。

2 パソコンによる実験

GUIと対話性に優れたJavaプログラムを作成して、次のような手順に沿ってそれをパソコン上で実行した。

[GA諸設定]

- 一世代数：親16個、子16個
- 突然変異率：制約や単語使用等の条件により変動^(注4)
- 世代交代数：20,000世代
- 淘汰方法：エリート保存・ルーレット選択

[実験環境]

- OS：Windows 98 Second Edition
- CPU：Pentium(r)II
- メモリ：63.0MB

[CWP作成手順]

- ①枠を作成するプログラムにより10×16規模の枠を作成する。(図7-1、表1)
- ②GAを応用したプログラムにより90%程度の完成度のCWPを作成する。(図7-2)
- ③②で作成したCWPを制約を満たしている単語だけを残し残りを消去して、新たな初期状態を作り上げる。同じ状態を繰り返す場合には、単語枠を少し手直しする。(図7-3)

④③で仮に決定したCWPに②で行ったGAを応用し再度残りの部分のCWPを完成させるまで③、④又は②、③、④等を繰り返して行く。(図7-4)

表1 10×16の枠情報

項目	2	3	4	5	6	7	合計
枠の長さ	2	38	15	7	7	3	44
単語の種類	26	248	917	999	986	948	4144
存在率	13	27	61	143	1401	3160	
重複数	7	25	44	26	29	15	72

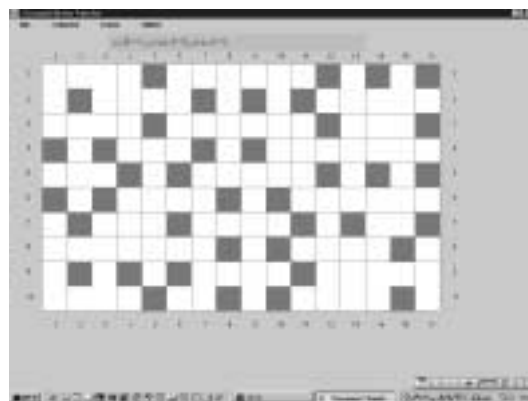


図7-1 初期状態



図7-2 GAによる自動生成



図7-3 9割程度の完成度



図7-4 完成

3 実験結果

20,000世代終了するまでのパソコン稼働時間は約3分であるが、現実上ほとんど問題はない。何故かと言えば、残りの10%の部分を仕上げの方が総和で10倍ぐらい時間がかかるからである。

CWP作成手順は実際には単純な順番で行われることは余りない。{}で手順の流れを、()で手順の繰り返しを表すとすれば、CWP完成までの様々な手順を表記することが出来る。例えば、もっとも単純な部類の手順には、{①②(③④)}があり、かなり複雑な部類の手順には{①(②)((③)④)}∪{①((②)(③))(④)}等一回の一連の流れでは完成できずに一連の流れを複数回行うこともある。

枠の規模を小さくしても未完成部分では完成したところの枠との制約があり、その制約を満たしながら、残りを完成させることはそんなに単純な作業ではないことが分かる。完成部分と未完成部分との制約数を如何に減らすかが、本方法の質的な問題点である。

IV 終わりに

今回、中規模な10×16のCWPを一応作成したが、同じ規模でも単語枠数や枠の重複数を増やしたりすることによって、作成の難易度が高くなる。しかし、極度に重複数を増やすと面白みがなくなる。例えば、大きさ規模5×8で全ての枠が重なる場合、単語枠数が13個、重複枠数が40個となり、この場合、重複度が大きくなりすぎて、もう虱潰しに調べる組み合わせ論的な方法以外、他の方法が存在しないかも知れない。

今後は、更に大規模な20×32のCWPの作成に挑戦するつもりであるが、その時の作成方法は単純でない変形されたGAを応用したいと考えている。

[注]

(注1)実際に確認するのは難しいが、例えば http://www.familybook.jp/book_teiki/03.html からある程度CWP関連雑誌の種類のを把握することができる。

(注2)インターネットにはCWP関連情報が記載されているウェブサイトが数多く存在する。その中でも外国のCWPについてよく纏められているサイトに

<http://www.crossword-puzzles.co.uk>

がある。そこに載せられているCWPコンパイラは圧巻である。定型な枠であるが、素早くCWPを作成する。その仕組みは明らかでないが、作成の様子を見ると組み合わせ方式と思われる。

(注3)第一目標では、90%程度の完成度を持ったCWPの作成を目指すので、その完成度の割合が最大適応度0.9程度になる。そして、最大適応度を持つ染色体は必ず次世代に選択される。それがエリート保存と呼ばれる選択方法である。

(注4)II章4の[突然変異の発生方法]で説明した通り、親1~親7の染色体に対しては必ず突然変異を行うので、その時の突然変異率は全個体数が32個だから7/32となる。しかし、突然変異で生じた遺伝子が制約を満たさない致死遺伝子の場合、その遺伝子は破棄され元の遺伝子が残るので実際には突然変異が行われない状況と同じになる。つまり、突然変異率を全個体数に対する突然変異の操作が行われた個体数と定義すれば、その値は0.219である。しかし、前述の通り、実際の結果を観れば突然変異の破棄が考えられるので、突然変異率は0.22未満の値になり、1回の世代交代における実際の突然変異率はプログラムで表示しない限り分らないし、また、その値は各世代ごとに変動する。

因みに、20,000世代交代における1世代あたりの実際に行われた平均突然変異率は0.001であった。

[参考文献]

- (1) 加部通明、生方俊典、クロスワードパズルの作成手法、情報処理学会第50回全国大会、1995年、2分冊P157-P159
- (2) 加部通明、生方俊典、クロスワードパズルの遺伝的アルゴリズムによる作成手法、情報処理学会第52回全国大会、1996年、2分冊P133-P134
- (3) Stuart C. Shapiro et al. Encyclopedia of Artificial Intelligence, Wiley, 1990, P205-211
- (4) 安居院猛、長尾智晴、ジェネティックアルゴリズム、昭晃堂、1993年