

インターロックと移動方向を考慮した平面分割パズルの生成

本郷元優^{†1} 北直樹^{†2} 斎藤隆文^{†3}
東京農工大学工学部知能情報システム工学科[†]

1. はじめに

インターロッキングパズルは、パズルを完成させると最後に組み込んだピースを除いた、全てのピース同士がかみ合っ固定されるという性質を持つ。

このパズルを設計するためには、完成した形状において全てのピースが互いに拘束され、かつデッドロックが生じないことを確認する必要がある。そのためこのパズルの手作業による設計は膨大な労力を必要とする。そこでインターロッキングパズルを自動生成するアルゴリズムが開発された[1][2]。

しかし従来手法では、インターロッキングパズル生成における制約の多さから、パズル作成者が生成時に与えられる入力条件は、ピースの数やモデルの形状に限られていた。これにより、ユーザはパズルの分解手順に関与することが困難であり、必ずしもユーザの意図と合致したパズルが生成されるとは限らない。

そこで本研究では、パズルの分解手順をユーザが制御可能にするために、ピースの移動方向をユーザが指定する二次元平面インターロッキングパズルの生成手法を提案する。その結果ユーザはパズルの分解手順に関与することができるようになった。

2. 関連研究

インターロッキングパズル生成アルゴリズムの関連研究には、Song らの研究[1]と Chen らによる研究[2]が挙げられる。

Song らの研究[1]では、ボクセルモデルにおけるピース生成時に、種となるボクセルに対し移動方向を確保するためのアンカーを近傍のボクセルから設定する。

また、Chen らによる研究[2]では、デッドロック等の原因でピースの生成が正常に達成されない際に、全てのピースが正常に構築または分解できる地点までバックトラック処理を行うことでパズルの生成を実現している。

Generation of planar partitioned puzzles with interlocks and specified movement directions

^{†1}HONGO MOTOMASA

^{†2}KITA NAOKI,

^{†3}SAITO TAKAFUMI

[†]Tokyo University of Agriculture and Technology, Department of Electrical Engineering and Computer Science

本研究では、Song らの研究[1]の移動方向制御を参考に、ピース周囲に壁を設定し移動方向を制御する。また、Chen らによる研究[2]で利用されたピース生成失敗時のバックトラック処理を、提案手法において参考にした。

3. 提案手法

本研究ではピース生成前にパズルモデルの周囲に動かない壁を設定し、各ピースの移動方向に存在する壁に穴を空けることで、ピースの任意移動方向を確保する。これにより、ピースの移動方向をユーザが指定することが可能な二次元平面におけるインターロッキングパズルの生成手法を提案する。

パズルに含まれるピース数は、ユーザ入力の指定移動方向を羅列した分解手順の段階数 d とする。生成されるピースを図 1 に示す。生成順に $P_i(1 \leq i \leq d)$ と表す。

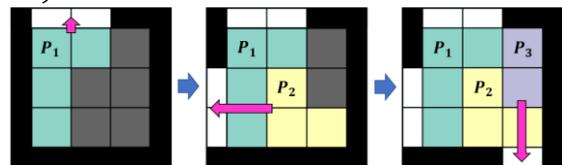


図 1 $d = 3$ [上, 左, 下]生成例

ピース生成は以下の順で実行される。

3.1 種セルの決定

インターロックと分解手順の一意性を確保するため、ピースを生成するための種セルは、 $i > 1$ の時、 P_i の進行方向に P_{i-1} が存在する必要がある。

したがって P_i の種セルは以下の条件を満たすセルから選定される。

- $i = 1$: 進行方向に壁が存在するセル。
- $i > 1$: 進行方向で P_{i-1} と接したセル。

3.2 ピースの拡張

種セルは隣接するセルをピースとして自身に追加し、拡張を行っていく。

パズルの組み立てを行う際、極端に大きいピースの存在はピース配置の大きな手がかかりとなり、パズルの難易度を損なう。したがってパズルの難易度を保障するため、ピースの拡張範囲を限定する。ピースに含まれるセル数は分解手順の段階数 d と、パズルモデルの総セル数 S を用いた、セル数 L の範囲内で拡張される。

$$2 \leq L \leq \lceil 1.1 \times S/d \rceil$$

3.3 移動性テスト

ピースが生成されると、移動方向を確保するために移動性テストが行われる。

P_i の進行方向上の、他ピースでないセルを自身(図 2a, 2b 黄部)に追加し(図 2a, 2b 橙部), 追加したセルと自身を最短経路で接続する(図 2c 橙部)。この時、進行方向に存在する他ピースでないセルを自身に追加し、壁セルは削除する(図 2d)。

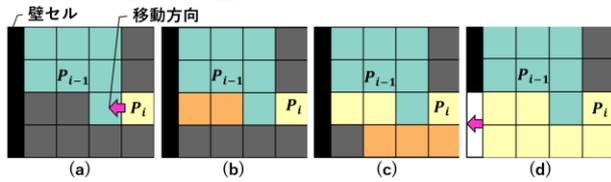


図 2 移動性テスト 手順

3.4 バックトラック

3.1 節から 3.3 節の工程を繰り返す際、入力条件を満たすピースまたはパズルが生成されない場合バックトラック処理を行う。直前のピース拡張または種セル選択をやり直し、再度生成を行うことで、ピース生成を続行する。

3.5 一意性フィルタ

3.1 節から 3.4 節の工程を終え出力されるパズルには、2 つのピースを同時に移動することによって、分解手順の一意性を満たすことが不可能となるパターンが存在する。分解手順が[緑,黄,紫,赤]の順になるよう生成されたパズル(図 3 左)において、紫と赤のピースを同時に移動させることで、意図した分解手順と異なる順序で分解することが可能になる(図 3 右)。

このような分解手順の一意性を満たさないパターンを提示しないようにするため、生成されたパズルにおいて、連続する 2 つのピース $[P_i, P_{i+1}] (i \leq d - 1)$ を 1 つのピースとして認識する新しいパズルを生成する。

$[P_1, P_2], [P_2, P_3] \dots [P_{d-1}, P_d]$ と組み合わせるパターンを変更することで、1 つのパズルに対して $d - 1$ 個の新しいパズルが生成される。これら全てのパズルに対して、分解手順が一意で定まることを確かめた場合に、生成結果として提示する。

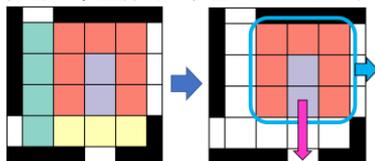


図 3 同時移動により、分解手順に従わない例

4. 生成結果

多様な形状に対して提案手法が適用可能であることを示すため、正方形と長方形のパズルを生

成する。

提案手法を用いて、入力方向を[上, 左, 下, 右]の順に指定し、可能な限りパターンを生成させた結果、下記の 4×4 の正方形では図 4 を、 3×5 の長方形では図 5 を生成した。いずれのパターンにおいても、パズルの分解手順が[緑,黄,紫,赤]の順に一意に定まることが確認された。

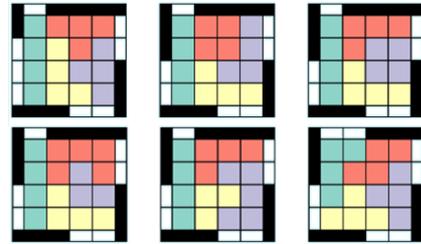


図 4 4×4 [上, 左, 下, 右]入力結果

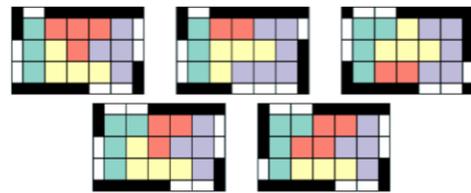


図 5 3×5 [上, 左, 下, 右]入力結果

5. おわりに

本研究では、ピースの移動方向をユーザが指定可能なインターロッキングパズルの生成手法を提案した。しかし現在のピース生成においては壁セルがどのように変化するか、ユーザが指定することはできない。

この問題に対処する場合、現在 1 ピースにつき 1 回の移動回数を複数回許容する手法が考えられる。ピースが複雑な軌道を描くことが可能になることで、壁の変化箇所をユーザが指定しても、デッドロックの生じないインターロッキングパズルを生成可能になることが期待できる。しかし現在の手法で複数回移動を許容する場合、ピースが 1 セル分移動するごとに 3.1 節から 3.3 節の工程を繰り返す必要がある。この際の計算量を考えた時、より計算量を削減したピース生成手法が必要であると考えられる。

参考文献

- [1] Peng Song, Chi-Wing Fu, Daniel Cohen-Or, Recursive Interlocking Puzzles, ACM Transactions on Graphics, 2012, volume 31, issue 6, 128:1-128:10.
- [2] Rulin Chen, Ziqi Wang, Peng Song, Bernd Bickel, Computational Design of High-level Interlocking Puzzles, ACM Transactions on Graphics, 2022, volume 41, issue 4, 150:1-150:15.