

あやとり表記法を入力とするあやとりの平面図生成ツールの開発

小泉 智史[†] 西田 誠幸[‡]
 拓殖大学工学部情報工学科[§]

1 はじめに

ひもに指を絡め、変形を繰り返すことで形を作る遊びとして知られるあやとりは世界中で親しまれている。

これまでに、あやとりを計算機で処理する手法のひとつとして、あやとりの二次元画像による可視化の方法^[1](以下可視化法)が提案されている。この方法は、指とひもの交差点の間の接続関係を持つデータ構造と、指によってひもをとる、指からひもを外すなどの操作、位相を変えない単純化と描画の処理からなる。

一方、あやとりのひもの操作を表現する方法として、String Figure Notation^[2](以下あやとり表記法)がある。これは指の動き、とるひもの位置、ひものとり方からなるひもの操作とその手順を表現する。

本稿では、我々が開発したあやとり表記法を入力とするあやとりの平面図生成ツールについて述べる。これは、あやとり表記法に基づく入力を受け付けて、あやとり可視化法によりあやとりの状態を可視化するツールである。さらに、本ツールの開発を行う上で、あやとり表記法の各操作があやとり可視化法に適用可能かどうか判定を行う必要があるため調査を行った。

2 あやとり表記法

あやとり表記法はあやとりに対する動作を表す表記法である。ひものとり方を絵や言葉で表現すると曖昧さが生じることがある。そこで、決められた表記を定義することで、あやとり表記法を知っていれば誰でも同じあやとりの形を作ることができる。

あやとり表記法では指や手を名詞で表し、名詞の前に接頭語をつけることでその名詞の場所を指定する。さらに指や手の動かし方やひものとり方を動詞で表す。以下に一例を示す。ただし、カッコ内は意味を表す。

re T (親指のひもを外す)
 T mo S pu fFS (親指で人差し指の奥のひもをとる)
 T gr nMS (親指で中指の手前のひもを握る)

3 あやとりの平面図生成ツールの概要

本研究で開発するあやとりの平面図生成ツールは、あやとり表記法で書かれたファイルを入力すると入力に対して構文解析を行い、入力が本ツールに適用可能か判定を行う。適用可能だった場合、入力にしたがっ

て仮想的なあやとりを変形し、単純化処理、最小化処理を行って平面図を描画する。ただし、仮想的なあやとりとそれに対する変形処理、単純化処理、最小化処理は先行研究の可視化法^[1]を踏襲する。以上の処理について次に述べる。

3.1 あやとりのデータ構造

本ツールは内部に仮想的なあやとりを持つ。あやとりのデータ構造はハンドル部とネット部からなる。ハンドル部は指と指にかかるひもを管理し、ネット部はひもと交差点を管理する。特にネット部のデータは、ハンドル部とネット部の境界に位置する点(以下境界点)と交差点からなる点集合と、これらの点を端点とする線分の集合で表す。図1にあやとりのデータ構造の概要を示す。

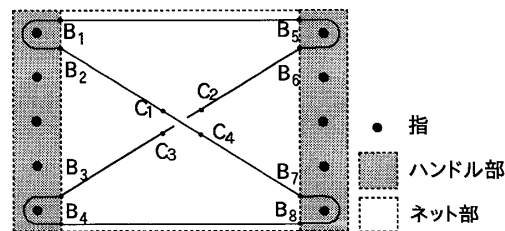


図1 あやとりのデータ構造の概要

一般的なあやとりではひもの長さが一定で、変形操作の過程で指の位置が変化する。だが、ひもの長さをもとに指の座標を移動させるよりも指の位置を固定しひもの長さを可変にするほうが処理が簡単のため、本可視化法では指の位置を固定とし、ひもの長さを可変とする。

3.2 あやとりの変形処理

あやとりの変形処理は、ひもを「とる」や「はずす」といった動作の処理である。あやとりに対する変形操作を表すネット部にハンドル部を置き換え、ハンドル部を再構成することで変形動作を実現する。これをブロックの生成・連結と呼ぶ。図1に対して、親指のひもを外した後に、小指の奥のひもを親指でとるという2段階の動作についてブロックを生成・連結した状態を図2に示す。

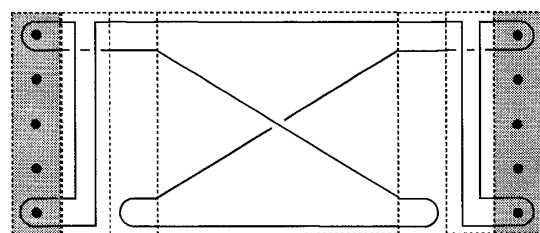


図2 動作ブロックの生成・連結

Development of a Tool for generating Two-dimensional Image of String Figure From String Figure Notation

[†]Satoshi KOIZUMI

[‡]Seikoh NISHITA

[§]Dept. of Computer Science, Faculty of Engineering, Takushoku University

あやとり表記法には、PalmString(掌のひも)という概念がある。例として、図3(5)の図のうち、縦に伸びているひもがPalmStringである。あやとり表記法では、PalmStringが存在するときはPalmStringと掌の間のひもに対して変形を行うことはほとんどない。指定がない限り、PalmStringと交差し、中央にのびるひもに対して処理を行うと定義されている。ところが、先行研究の可視化法では指にかかるひもしか変形処理ができないため、あやとり表記法の定義通りに変形処理が行えない。そこで、本ツールではPalmStringを含む場合でも、適切な変形処理ができるように拡張した。

3.3 簡単化処理と最小化処理

簡単化処理は指にかかるひもは変えずに、不必要な交差点を削除することであやとりのデータを簡単にする処理である。一方、最小化処理はあやとりの平面図を描画するために各端点の座標を定める処理である。ただし、可視化法では境界点の座標は固定であるため、交差点の座標のみを変更し、ひもの長さが最小になるように座標を求める。

4 可視化法に対するあやとり表記法の適用可能性

あやとり表記法の全ての表記が、あやとり可視化法に適用できるわけではない。そのため、予め適用できるものとできないものを分けておく必要がある。そこで、全ての表記に対して調査を行った。その結果、適用できないものは、大きく3つに分けられることがわかった。

1つ目は指の位置に関する表記で、にぎる(gr)や押し下げる(de)がそれに当たる。可視化法では指の位置は固定のため適用できない。

2つ目はひもの位置を明確に指定しない表記で、交差しているひも(XS)がそれに当たる。ひもの位置を明確に指定しない操作は可視化法には適用できない。

3つ目は指にかかっていないひもに対する操作である。可視化法では、境界点を端点として持たない線分に対する操作は処理できない。これは、変形処理がハンドル部の置き換えによって処理されるためである。

本ツールでは、PalmStringについて拡張を行ったことで、境界点を線分として持たないひもに対する処理の一部ができるようになった。

これまでに適用可能性について、あやとり表記法の全ての表記について調査を行い、その結果を本ツールの適用可能性判定処理に使用した。

5 あやとり平面図生成ツールの実装

本ツールは、Java SE Development Kit(JDK) 6を用いて実装した。本ツールの出力の様子を図3に示す。(1)のOpeningAはあやとりの始めに作る代表的な形のひとつである。以下の処理を(1)OpeningAに対して順番に行った過程の出力結果である。

1. OpeningA
2. T mo S pu fFS(親指で人差し指の奥のひもをとる)

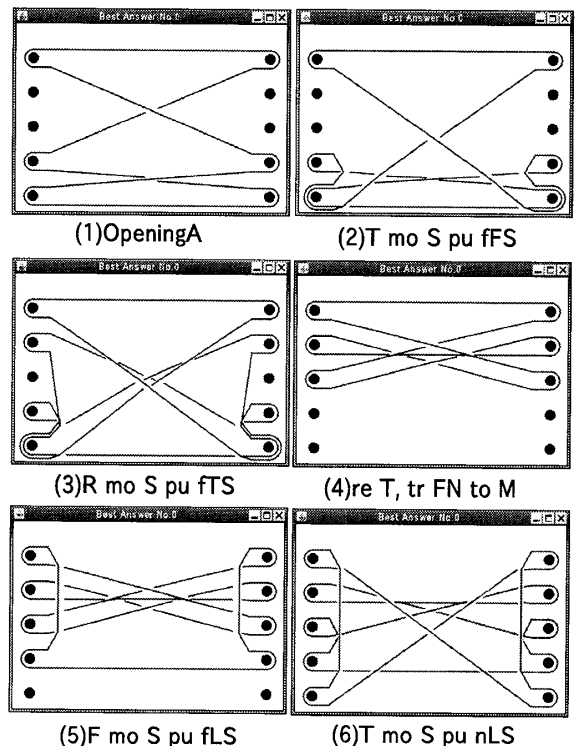


図3 あやとりの平面図生成ツールの出力の様子

3. R mo S pu fTS(薬指で親指の奥のひもをとる)
 4. re T(親指のひもを外す)
tr FN to M(人差し指のひもを中指に移す)
 5. F mo S pu fLS(人差し指で小指の奥のひもをとる)
 6. T mo S pu nLS(親指で小指の手前のひもをとる)
- 3と6の変形処理のときにPalmStringが存在しているが、あやとり表記法の定義通りに可視化することができた。

6 おわりに

本稿では、あやとり表記法を入力とするあやとりの平面図生成ツールについて述べた。これは、あやとり表記法に基づく入力を受け付けて、あやとり可視化法によりあやとりの状態を可視化するツールである。本ツールでは入力をあやとり表記法としている点を除き、基本的に先行研究の可視化法を踏襲している。さらに、PalmStringに注目し処理を拡張することで指に直接かかっていないひもに対する処理の一部が可能となった。

参考文献

- [1] 山田雅之, RHAHMAT BUDIARTO, 伊藤英則, 世木博久
あやとりにおけるひも図形変形過程の表現とその処理, 情報処理学会論文誌 Vol.35 No.3(1994)
- [2] Eric Lee, St. David, AZ
"A Shorthand Notation for Recording STRING FIGURE"
<http://www.alysion.org/figures/notation.htm>