岡山理科大学紀要 第43号 A pp. 1-6(2007)

交差点数が11以下の絡み目の生成とJones多項式の計算

須藤 加奈子

岡山理科大学大学院理学研究科博士課程応用数学専攻（修了生）

(2007年９月25日受付、2007年11月２日受理)

１．序論

Jones多項式は結び目理論の重要な不変量である。ある2つの絡み目のJones多項式が等しく、同値でない絡み目はまだ知られていない。従ってJones多項式の表を作るのは重要な問題であり、交差点数が10以下の絡み目のJones多項式のよく知られた表はたくさん存在する。

我々は絡み目を帰納的に生成させ、その生成させた絡み目のJones多項式を計算するコンピュータプログラムを開発した。

この論文では、プログラムのアルゴリズムの概要を述べ、そしてそのプログラムを使って得られた交差点数が11の絡み目の一部のリストを与える。

２．組み紐に関する準備

２－１　組み紐の定義

1つの箱の天井と床にそれぞれn個の点を定め、以下固定する。

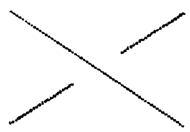
n-組み紐とは、n本の紐で天井の1点と床の1点を箱の中で結んだものである。天井と床のそれぞれのn点に対して、それぞれ1, 2, ..., nと番号を付けることにする。

箱の1つの側面を決め、それを以下固定する。組み紐Kのその側面のxy-平面上に直交射影する写像をPとする。Kの像P(K)の各点vに対して、

- $P^{-1}(v)\cap K$は1点または2点である。
- $P^{-1}(v)\cap K$が2点の時は、vはP(K)の2辺の交差点になっている。

をみたす時、Pの像P(v)を正則射影と呼ぶ。

$P^{-1}(v)\cap K$が2点の時、z座標の小さい点を通るKの辺（以下、下交差線と呼ぶ）の像の交差点の近傍の1部を削除して得られる図形



を正則表示と呼ぶ。また、z座標の大きい点を通るKの辺を上交差線と呼ぶ。

2つの組み紐WとW’が、紐を切ったり繋げたりせず、さらに天井と床の点を移動したりもせずに1方から他方に変形できる時、WとW’は同型であるという。

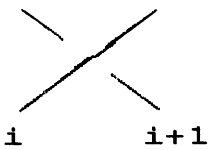
組み紐b_1とb_2に対し、b_1の床の点とb_2の天井の点を番号順に繋げると新たな組み紐bを得る。このbをb_1とb_2の積と定める。

n-組み紐の同型類全体をB_nと表すと、B_nはこの積に関して組み紐群と呼ばれる群になる。

２－２　組み紐の表現

天井の番号i, i+1の点がそれぞれ床のi+1, iの点と結ばれ、交差点は1個だけで、天井のiと床のi+1の点を結ぶ紐がその交差点における下交差線となっている正則表示をもつ組み紐を

σ_iとする。



組み紐群は$\sigma_1, \ldots, \sigma_n$とその逆元$\sigma_1^{-1}, \ldots, \sigma_n^{-1}$で生成される。

2－3　不変量

各々の組み紐Tに対してある数学的な量$\rho(T)$がきまって、$T \cong T'$ ならば$\rho(T) = \rho(T')$となる時、対応ρを組み紐の不変量という。不変量ρが決まると$\rho(T) \neq \rho(T')$ならば、$T \ncong T'$であるとわかる。

3．組み紐の交換関係

3－1　Markov移動とReidemeister移動

任意の組み紐b($\in$B)からbσ_nまたはbσ_n^{-1}への変形をMarkov移動と呼ぶ。

ある組み紐の元とその逆元の積は1になる変形、

$$\sigma_i \sigma_i^{-1} = 1$$

をReidemeisterⅡ移動と呼ぶ。また、

$$\sigma_i \sigma_{i+1} \sigma_i = \sigma_{i+1} \sigma_i \sigma_{i+1}$$

$$\sigma_i^{-1} \sigma_{i+1}^{m} \sigma_i = \sigma_{i+1} \sigma_i^{m} \sigma_{i+1}^{-1}$$

の変形をReidemeisterⅢ移動と呼ぶ。

3－2　合成絡み目

絡み目$K(=K_1 \cup \ldots \cup K_\mu)$と球面$\Sigma(\in S^3)$が2点$v_1, v_2$で交わるものとする。但し、2点$v_1, v_2$は$K_i$の同一成分上にあるものとする。$K_i$は$\Sigma$上に沿って$v_1$と$v_2$を結んだ単純折線により分割される。よってKは、絡み目B_1とB_2に分割される。この時、KをB_1とB_2の合成絡み目（$B_1 \# B_2$）と呼ぶ。

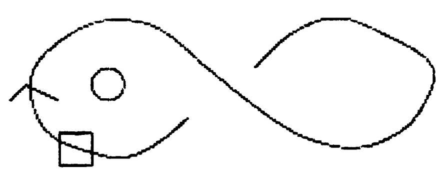
3－3　組み紐と絡み目の関係

組み紐の天井の番号と同じ床の番号を組み紐の外側で絡まないように結ぶと絡み目となる。これを閉じた組み紐と呼ぶ。

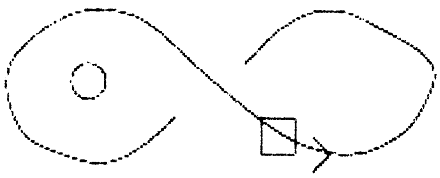
逆に以下のようにして絡み目が全同位によって閉じた組み紐の形に変形できることがわかる。

絡み目の紐と紐の間の余白の領域に、ある1点を定め、その1点を軸とする。各成分上に1点を定め、そこを始点として時計回りに進む。もし反時計回りに回ったらその部分を軸の反対側に移動する。この作業を全ての成分について始点に戻るまで繰り返す。そうすると閉じた組み紐になる。

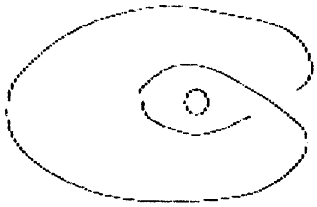
例えば



の中の○を軸とし、□の中の点を始点とする。始点から時計回りに進む。そうすると下の図の□で反時計回りに回ることになる。



そこで下の図のように軸の反対側に移動すると、閉じた組み紐になる。



3－4　組み紐の自動生成と計算

正整数Kを1つ定める。紐数をn(<K)、交差点数がK-nである組み紐を以下のように生成していく。

積 $\sigma_{P_1}^{\varepsilon_1}\sigma_{P_2}^{\varepsilon_2},\ldots,\sigma_{P_{K-n}}^{\varepsilon_{K-n}}$ であらわされる組み紐を $\varepsilon_1,\varepsilon_2,\ldots,\varepsilon_{K-n},P_1,P_2,\ldots,P_{K-n}$ に関する辞書式順序に生成する。nとKを順に増やしていけば全ての組み紐をもれなく生成することができる。

K=1の場合はn=0となり、n=2, K=3の場合は自明な絡み目となるので、初期値をK=4, n=2と定める。

例えばK=5, n=3の場合は

$$\sigma_1\sigma_1,\ \sigma_1\sigma_1^{-1},\ \sigma_1\sigma_2,\ \sigma_1\sigma_2^{-1},\ \sigma_1^{-1}\sigma_1,\ \sigma_1^{-1}\sigma_1^{-1},\ \sigma_1^{-1}\sigma_2,\ \sigma_1^{-1}\sigma_2^{-1},\ \sigma_2\sigma_1,\ \sigma_2\sigma_1^{-1},$$

$$\sigma_2\sigma_2,\ \sigma_2\sigma_2^{-1},\ \sigma_2^{-1}\sigma_1,\ \sigma_2^{-1}\sigma_1^{-1},\ \sigma_2^{-1}\sigma_2,\ \sigma_2^{-1}\sigma_2^{-1}$$

の順に生成する。

組み紐が生成された順番に参考文献2), 4)の定義に従って、Bracket多項式、正規化Bracket多項式、Jones多項式を計算する。ただしこのとき、参考文献1)の定義に従って、Reidemeister移動やMarkov移動などで交差点数が減らせる組み紐はすでに計算済みなので除外する。また、合成絡み目の上述の不変量は簡単に計算することができるので除外する。

3－5　今までの結果と今後の目標

コンピュータを使ってKが23以下で、K-n≦11となるような組み紐を生成し、それらのJones多項式の計算をした。同値なものを除くとそれらのJones多項式は実際に異なっていて、いくつかの知られた結果と一致していることを確認した。今後はさらに計算を続け、生成された組み紐の同値性の確認を自動化することが課題である。

3－6　プログラムを使って得られた交差点数が11の絡み目のリスト

組み紐 $\sigma_{P_1}^{\varepsilon_1}\sigma_{P_2}^{\varepsilon_2},\ldots,\sigma_{P_{K-n}}^{\varepsilon_{K-n}}$ を紐数と積 $\varepsilon_i P_i$ (i=1, ..., K-n) の列で表したもののリストを以下に挙げる。例えば 4:1, 1, 2, 2, -1, 3, -2, 3, 3, 2, 2 は紐数 4 の組み紐

$$\sigma_1\sigma_1\sigma_2\sigma_2\sigma_1^{-1}\sigma_3\sigma_2^{-1}\sigma_3\sigma_3\sigma_2\sigma_2$$

を表す。

4:1, 1, 2, 2, -1, 3, -2, 3, 3, 2, 2
4:1, 1, 2, 2, -3, 2, -1, 2, 3, 3, 2

4:1, 1, 2, −3, 2, −1, 2, 3, 3, −2, −2
4:1, 1, 1, 1, 2, 2, −3, 2, 1, −3, −2
4:1, 1, 1, 1, −2, 1, −3, 2, −3, −3, −2
4:1, 1, 2, 2, 2, 2, −3, 2, −1, −3, 2
4:1, 1, 2, −3, 2, 2, 2, −1, 2, −3, −3
4:1, 1, 1, 1, 2, −3, 2, 2, −1, 2, −3
4:1, 1, 1, 2, 2, −3, 2, 1, −3, −3, −2
4:1, 1, 1, 2, −3, −3, −3, 2, −1, 2, −3
4:1, 1, 2, 2, 2, −3, 2, −1, −3, 2, 2
4:1, 1, 1, 2, 2, −3, 2, −1, −3, −3, 2
4:1, 1, 2, 2, −3, −3, 2, 2, −1, −3, 2
4:1, 1, 1, −2, 1, 1, −3, 2, −3, −3, −2
4:1, 1, 1, 1, 2, −3, 2, 2, −1, 2, −3
4:1, 1, 2, 2, −1, −3, 2, 2, −3, −3, 2
4:1, 1, 2, 2, 2, −3, 2, −1, 2, −3, −3
4:1, 1, 1, 2, −3, 2, −1, 2, 2, 2, −3
4:1, 1, 1, 2, −3, 2, 2, 2, 1, −3, −2
4:1, 1, 1, 2, 2, 2, −1, −3, −3, 2, −3
4:1, 1, 2, −3, 2, 2, −3, 2, −1, 2, −3
4:1, 1, 1, 2, −3, 2, 2, −1, −3, 2, −3
4:1, 1, 2, −3, 2, −3, 2, −1, 2, −3, −3
4:1, 1, 1, 2, −3, 2, −1, −3, −3, 2, −3
4:1, 1, 1, 2, −3, 2, −1, 2, 2, −3, 2
4:1, 1, 2, 2, −3, 2, −1, 2, −3, −3, 2
4:1, 1, 2, 2, −3, 2, −1, 2, 2, −3, 2
4:1, 1, 2, 2, −3, 2, 2, −1, 2, −3, 2
4:1, 1, 1, 2, −3, 2, −1, −3, 2, −3, 2
4:1, 1, 2, −3, 2, −1, −3, 2, 2, −3, 2
4:1, 1, 2, −3, 2, −1, 2, −3, −3, 2, −3
4:1, 1, 2, −3, 2, −1, 2, −3, 2, −3, 2
4:1, 1, 2, −3, 2, −1, −3, 2, −3, 2, −3
4:1, 1, 1, −2, −2, −1, 3, −2, −1, 3, −2
4:1, 1, 2, 2, 1, −3, 2, −3, −3, 2, 2
4:1, 1, 1, −2, 3, −2, −1, 3, −2, 3, −2
4:1, 1, −2, −2, −1, 3, −2, −1, 3, −2, 3
4:1, 1, −2, 1, −3, −2, 1, −2, 3, 3, −2
4:1, −2, −2, −1, 3, −2, −2, −2, −1, 3, −2
4:1, 1, 2, 2, 1, −3, 2, 2, −3, −3, −2
4:1, 1, 1, −2, −2, −2, 1, 1, −3, 2, −3
4:1, 1, −2, −2, −1, −1, −3, 2, 1, −3, −2
4:1, 1, −2, 1, −2, −2, −1, −1, −3, 2, −3
4:1, 1, 2, 2, −1, −3, 2, −1, 2, −3, −3
4:1, 1, 1, 2, −3, 2, −1, −1, 2, 2, −3
4:1, 1, 1, 2, −3, 2, 1, 1, 2, 2, −3
4:1, 1, 1, 2, −3, 2, 1, 1, −3, −2, −2
4:1, −2, −2, −2, −1, −1, −3, 2, −1, 2, −3
4:1, 1, 1, 1, 2, 2, 1, −3, 2, −3, 2
4:1, 1, 1, 1, 2, −3, 2, 1, −3, −2, −2

4:1, 2, 2, -3, 2, 2, -1, -1, -3, 2, -3
4:1, 1, 2, 2, -1, -3, 2, -1, 2, -3, -3
4:1, 1, -2, -2, -1, -3, 2, -1, -1, -3, -2
4:1, 1, 1, 2, 2, 1, 1, -3, 2, -3, -3
4:1, 1, -2, 1, 1, -2, 1, -3, 2, -3, -3
4:1, 1, -2, -2, -2, -1, -1, -1, 3, -2, 3
4:1, 1, 2, 2, 1, -3, -3, -2, 1, -2, -3
4:1, 1, -2, -2, -2, 1, -3, -2, 1, -2, -3
4:1, 1, -2, 3, -2, -2, 1, 3, 2, 2, 3
4:1, 2, 2, 2, 1, -3, 2, 2, -3, 2, -3
4:1, 1, -2, 1, -2, 1, 3, 2, 2, 2, 3
4:1, 1, 2, 2, 1, -3, -3, 2, -3, 2, -3
4:1, 1, -2, -2, 1, 3, -2, -1, -1, 3, -2
4:1, 1, 1, -2, -2, -1, -3, 2, -1, -3, -2
4:1, 1, -2, -2, 3, -2, -1, -1, -1, 3, -2
4:1, 1, 2, -3, 2, 2, -1, -1, -1, 2, -3
4:1, 1, 2, -3, 2, -1, -1, -1, 2, -3, -3
4:1, -2, 1, -2, -2, -2, -2, -1, 3, -2, 3
4:1, 1, 1, -2, -2, 1, -3, 2, 1, -3, -2
4:1, 1, 1, -2, 1, 1, -3, 2, 1, -3, -2
4:1, 1, 2, 2, 1, -3, -2, -2, -2, -3, -3
4:1, 1, 2, 2, 1, -3, 2, 2, 2, -3, -3
4:1, 1, 2, 2, 1, -3, 2, 2, 2, -3, -3
4:1, 1, 2, -3, 2, 1, -3, -2, -2, -2, -2
4:1, 1, 1, 2, -3, 2, -1, -1, -1, 2, -3
4:1, 1, -2, -2, -1, -1, -3, 2, -1, -3, -2
4:1, 1, 2, 2, -3, -3, 2, -1, 2, 2, 3
4:1, 1, 1, -2, 3, -2, -1, 3, -2, 3, -2
4:1, 1, 2, 2, 2, -3, -3, -2, 1, -2, -3
4:1, 1, 1, -2, 1, -2, 1, 1, -3, 2, -3
4:1, 1, -2, -2, -1, -3, 2, -1, -1, -3, -2
4:1, 1, -2, 1, -2, 1, -3, 2, 1, -3, -2
4:1, 1, 1, 2, -3, 2, 1, -3, 2, -3, 2
4:1, 1, -2, 1, -2, 1, -3, 2, 2, 2, -3
4:1, 1, 2, 2, -1, -3, 2, -1, 2, -3, 2
4:1, 1, 2, 2, -3, 2, -1, 2, -1, -3, 2
4:1, 2, 2, -3, 2, 1, -3, 2, -3, 2, -3

参考文献

1) Kanako Sutou : Elimination of certain crossings of braids. Preprint.
2) Kunio murasugi : Jones Polynomials and classical conjectures in knot theory, Topology, Vol. 26, No. 2, pp. 187-194 (1986)
3) Larry Wall, Tom Christiansen, Jon Orwant : Programming Perl Third Edition, O'REILLY (2002)
4) Louis H. Kauffman : KNOT AND PHYSICS, World Scientific (1993)

Generation and calculation of Jones polynomials of knots with eleven crossings

Kanako SUTOU

Graduate School of Science,
Okayama University of Science,
1-1 Ridai-cho, Okayama 700-0005, Japan

(Received September 25, 2007; accepted November 2, 2007)

The Jones polynomial is an important invariant in knot theory. Until now, any two knots of which Jones polynomials equal to each other and of different types are not known. Thus it is an important problem to make tables of Jones polynomials. There are some well known tables of Jones polynomials for knots with ten or less than ten crossings.

We developed computer programmes to generate knots inductively and to calculate Jones polynomials of generated knots. In this article, we outline algorithm of the programmes and give a list of some knots with eleven crossings obtained with the programmes.

Keywords: knot; Jones polynomial; computer program.