Regular Möbius Knot Tree (RMKT), see Fig. 306.

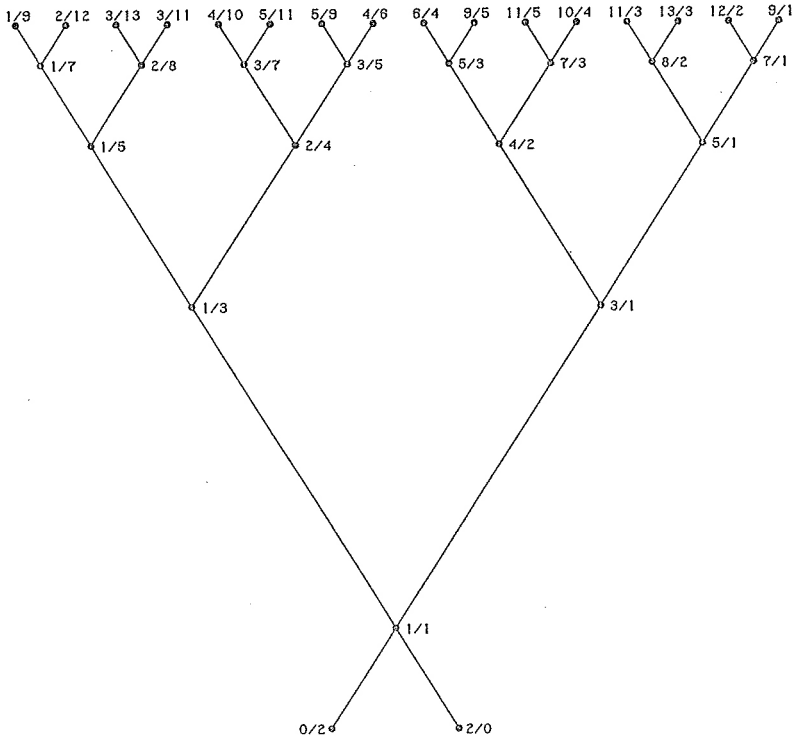


Fig. 306 — The Regular Möbius Knot Tree.

We may obtain the string-run diagram of a Regular Möbius Braid from the string-run diagram of its virtual Regular Cylindrical Braid by deleting the Matthew Walker section. Since p_m and b_m are both either odd or even, we have two types of string-run diagrams. The leftmost diagram in Fig. 307 depicts the type of string-run diagram (solid lines) for a Regular Möbius Braid with p_m and b_m both **odd**; the fourth diagram from the left depicts the type of string-run diagram (solid lines) for a Regular Möbius Braid with p_m and b_m both **even**. The uppermost solid horizontal line is the lowermost solid horizontal line after it has received a rotation of 180° about the vertical centre-line of the diagram as axis of rotation. The virtual Regular Cylindrical braid incorporates the dotted Matthew Walker section. This Matthew Walker Section has one of the two depicted codings.

A Regular Möbius Braid with an aesthetically acceptable weaving pattern requires a coding which has **Evert-Lateral** equivalency. In Fig. 308 two column-coded and two row-coded examples of such a coding are depicted. Besides the restriction that p_m and b_m must both have the same parity, row-coding demands the further condition that b_m must be a multiple of the number of rows in a coding-block. Hence for the leftmost

bottom coding in Fig. 308 b_m must be a multiple of 4, and for the rightmost bottom coding in Fig. 308 b_m must be a multiple of 5.

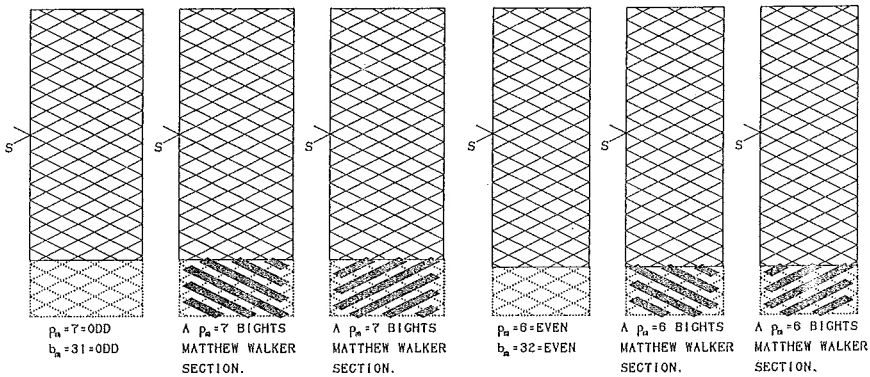


Fig. 307 — The string-run diagrams for p_m and b_m both odd, and both even.

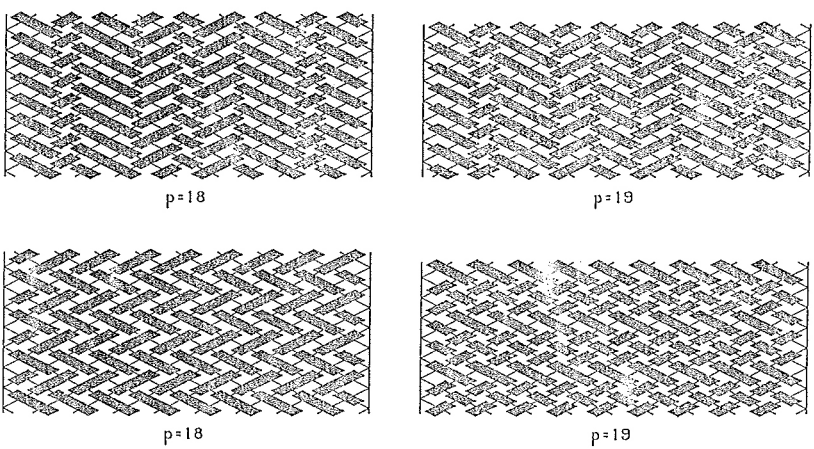
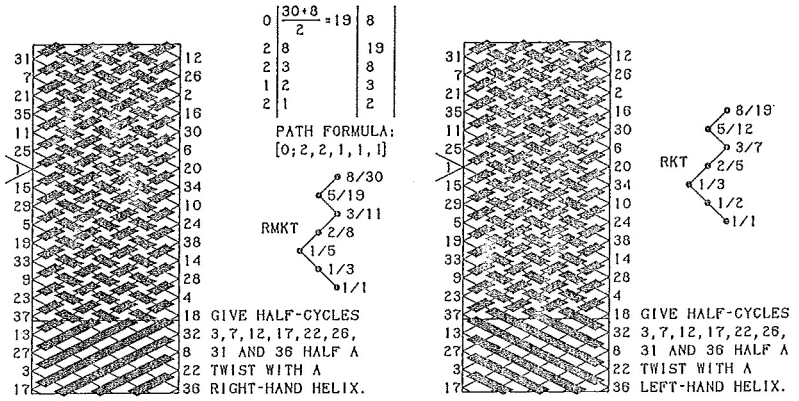


Fig. 308 — Examples of Evert-Lateral coding equivalency.

It will thus be evident that a Regular Möbius Braid cannot have a Gaucho-coding, nor can it have a Casa-coding when p_m is odd. It can, however, have a Headhunter's-coding, or when p_m is even a Casa-coding.

Fig. 309 shows the grid-diagrams with their associated instructions of the virtual Regular Cylindrical Braids which represent Regular Möbius Knots with $p_m = 8$ parts and $b_m = 30$ bights. After braiding the left-hand virtual Regular Cylindrical Braid, the band should be given a half twist with a right-hand helix, while the right-hand virtual Regular Cylindrical Braid should be given after braiding a half twist with a left-hand helix. Each braid can then be tightened to a correctly finished Regular Möbius Knot.



1)	FREE RUN.	20)	2u-2o.
2)	FREE RUN.	21)	2o-u-o.
3)	FREE RUN.	22)	3u-o.
4)	FREE RUN.	23)	2o-2u.
5)	FREE RUN.	24)	2u-2o-u.
6)	u.	25)	2o-2u-o.
7)	o.	26)	5u.
8)	u.	27)	2o-2u-o.
9)	o.	28)	2u-2o-u.
10)	u-o.	29)	2o-2u-o.
11)	o-u.	30)	u-o-u-2o-u.
12)	2u.	31)	6o.
13)	o-u.	32)	u-o-u-2o-u.
14)	u-o.	33)	o-u-o-2u-o.
15)	o-u.	34)	u-o-u-o-u-o-u.
16)	2u-o.	35)	o-u-o-u-o-u-o.
17)	3o.	36)	7u.
18)	2u-o.	37)	o-u-o-u-o-u-o.
19)	2o-u.	38)	u-o-u-o-u-o-u.

1)	FREE RUN.	20)	2u-2o.
2)	FREE RUN.	21)	2o-2u.
3)	FREE RUN.	22)	4o.
4)	FREE RUN.	23)	2o-2u.
5)	FREE RUN.	24)	2u-2o-u.
6)	u.	25)	2o-2u-o.
7)	o.	26)	2u-3o.
8)	o.	27)	4u-o.
9)	o.	28)	2u-2o-u.
10)	u-o.	29)	2o-2u-o.
11)	o-u.	30)	u-o-u-2o-u.
12)	2o.	31)	o-5u.
13)	o-u.	32)	2o-u-2o-u.
14)	u-o.	33)	o-u-o-2u-o.
15)	o-u.	34)	u-o-u-o-u-o-u.
16)	2u-o.	35)	o-u-o-u-o-u-o.
17)	3u.	36)	7o.
18)	2u-o.	37)	o-u-o-u-o-u-o.
19)	2o-u.	38)	u-o-u-o-u-o-u.

Fig. 309 — Casa-coded $p_m/b_m = 8/30$ Regular Möbius Knots.

Note that the procedure at the left in Fig. 309 leads through a virtual multi overhand knot (it has two half twists with a right-hand helix), whereas the procedure at the right in Fig. 309 does not lead through a virtual multi overhand knot.

Although we can braid any Regular Möbius Knot by means of a virtual Regular Cylindrical Knot, a procedure which has theoretical advantages with respect to its path in the RMKT, it is from the practical point of view not a good method. The reason being threefold:

(1). The half twists in the Matthew Walker section makes the braiding process more cumbersome, because we have to pay careful attention that they do not accidentally disappear.

(2). The Matthew Walker coding of the crossings in the Matthew Walker section makes the braiding process, through the half-cycles affected, more cumbersome.

(3). At the end of the braiding process much slack will have to be taken out.

It will thus be obvious that, if possible, a much more practical braiding process should be used. A process which is direct rather than indirect through some virtual

braid-form. Fortunately such a process does exist, and what is more, the procedures involved fit in beautifully with our braiding procedures for Regular Cylindrical Knots and Braids. This is of course not surprising since there is clearly some close relationship between Regular Möbius Braids and Regular Cylindrical Braids, a relationship well demonstrated by their respective evolution trees.

Let's take a closer look at the grid-diagram of a Regular Möbius Knot, and let's first look at the case where both p_m and b_m are **odd**.

We shall start with the grid-diagram of the virtual Regular Cylindrical Knot having $p = p_m = 7$ and $b = \frac{p_m+b_m}{2} = 19$, which represents a 2-pass Headhunter's-coded Regular Möbius Knot with $p_m = 7$ and $b_m = 31$. This diagram with its braiding instructions is presented in Fig. 310.

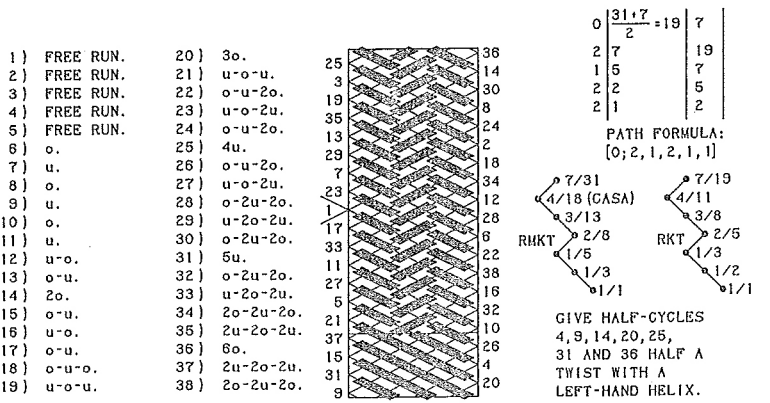


Fig. 310 — The virtual Regular Knot $p/b = 7/19$ which represents a Regular Möbius Knot $p_m/b_m = 7/31$.

Let's redraw this grid-diagram with the string-run only (dotted lines) in the Matthew Walker section; see Fig. 311. Take the left-hand diagram and follow half-cycle #1, then follow half-cycle #2, then half-cycle #3 till we reach the upper horizontal solid line. We have now to rotate this line through 180° as described on page 356, and continue, from the bottom horizontal solid line, with this half-cycle till we reach the bight-boundary. But in reality the braid **gradually** rotates through the 180°, and hence half-cycle 3 continues to the apparent bight-boundary on the right. In order to show this clearly in a grid-diagram, we rotate our left-hand grid-diagram through the 180° and obtain the second grid-diagram from the left. Thus half-cycle #3 runs from the apparent left-hand bight-boundary of the leftmost grid-diagram to the upper horizontal solid line and continues from the bottom horizontal solid line of the second grid-diagram from the left to the apparent right-hand bight-boundary of this grid-diagram. Next in this second grid-diagram from the left, half-cycle #4 runs from the apparent right-hand bight-boundary to the apparent left-hand bight-boundary. Then follows half-cycle #5 in this diagram, next half-cycle #6, then half-cycle #7 till the upper horizontal line (all in the second grid-diagram from the left). This half-cycle continues from the bottom horizontal solid line of the leftmost grid-diagram to the apparent right-hand

bight-boundary of this grid-diagram. Then follows half-cycle #8 which runs from the apparent right-hand bight-boundary (leftmost grid-diagram) to the apparent left-hand bight-boundary of this grid-diagram. And so on.

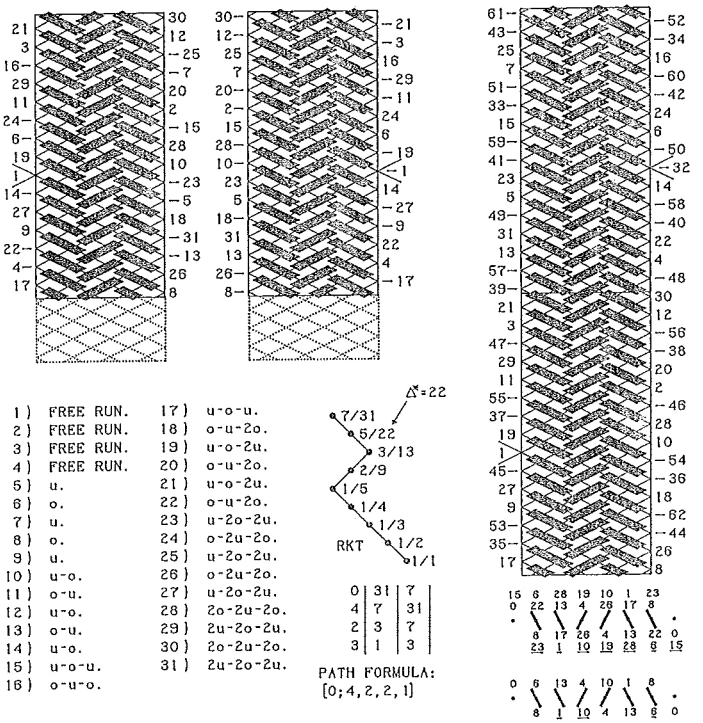


Fig. 311 — The Regular Möbius Knot $p_m/b_m = 7/31$ from Fig. 310.

It is as if two strings are being laid down simultaneously; one string lays down the half-cycles

$$\#1\,;\ \#2\,;\ \#3\,;\ \#4\,;\ \#5\,;\ \#6\,;\ \#7\,;\ \text{etc.}$$

and the other string lays down simultaneously the half-cycles

$$\#-1\,;\ \#2-\,;\ \#-3\,;\ \#4-\,;\ \#-5\,;\ \#6-\,;\ \#-7\,;\ \text{etc.}$$

However the string which lays down the half-cycles #1 ; #2 ; etc. is **real**, but the string which lays down the half-cycles #−1 ; #2− ; etc. is **imaginary** since these half-cycles are in fact the half-cycles of the real string.

It will now be obvious that the string-run in the Regular Möbius Braid can much easier be followed in a grid-diagram which consists of the second grid-diagram from the left stacked on top of the leftmost grid-diagram. Such a diagram is depicted on the right-hand side of Fig. 311. It is the grid-diagram of a Regular Cylindrical Braid with $p/b = p_m/b_m$ in which two strings are being laid down simultaneously. One string starts with half-cycle #1 and the other string starts simultaneously with half-cycle

$\# -1 = 32$. The half-cycles $\#2$ and $\#2- = 33$ are being laid down simultaneously, as are the half-cycles $\#3$ and $\# -3 = 34$; $\#4$ and $\#4- = 35$; $\#5$ and $\# -5 = 36$; etc. The string which lays down the half-cycles $\#1$; $\#2$; etc. is **real**, but the string which lays down the half-cycles $\#32$; $\#33$; etc. is **imaginary**, nevertheless in the Regular Möbius Braid the half-cycles of the **real string-run** intersect in actual fact the **imaginary string-run**. In order to handle this situation in a simple way, we do **as if** the imaginary string-run starts at the left-hand bight-boundary with half-cycle $\#31^*$. This is thus an imaginary half-cycle of higher order, and consequently the *real string-run* does **not** intersect this particular half-cycle.

With Euclid's algorithm we can readily calculate the path formula for this $p/b = p_m/b_m = 7/31$ Regular Cylindrical Braid (a Regular Knot since g.c.d.$(7,31) = 1$), and hence we can readily find its associated Δ^*-value, which in this case is equal to 22. With this Δ^*-value we can construct the algorithm-diagram for the *real* string-run in the usual manner with its complementary cyclic bight-number scheme which has the i-value sequence $0, 22, 13, 4, 26, 17, 8, 30, 21, 12, 3, 25, 16, 7, 29, 20, 11, 2, 24, 15, 6, 28, 19, 10, 1, 23, 14, 5, 27, 18, 9$.

This gives in the algorithm diagram the i-value sequence $0, 22, 13, 4, 26, 17, 8$.

With this Δ^*-value, the complementary cyclic bight-number scheme associated with the *imaginary* string-run, including the imaginary half-cycle of higher order, has from the Standing End bight-point of the *real* string-run the i-value sequence $\frac{b_m-1}{2} = 15, 6, 28, 19, 10, 1, 23, 14, 5, 27, 18, 9, 0, 22, 13, 4, 26, 17, 8, 30, 21, 12, 3, 25, 16, 7, 29, 20, 11, 2, 24$.

This gives in the algorithm diagram the i-value sequence $15, 6, 28, 19, 10, 1, 23$.

Both these i-value sequences are set off in the algorithm diagram for the Möbius braid, but since a real half-cycle cannot intersect the imaginary half-cycle of higher order, we underline, for the half-cycles which run from lower right to upper left, the i-values associated with the imaginary string-run.

It is as if we lay down 1 imaginary half-cycle of higher order, $b = b_m$ real half-cycles, and $b = b_m$ imaginary half-cycles, hence a total of $(2b+1) = (2b_m+1)$ half-cycles. Consequently since $(2b+1) = (2b_m+1) =$ odd, the maximum i-value in the i-value sequence is $\frac{(2b_m+1)-3}{2} = 15$. Thus the actual applicable complementary cyclic bight-number scheme has the i-value sequence $0, \underline{6}, 13, 4, \underline{10}, \underline{1}, 8, \underline{14}, \underline{5}, 12, 3, \underline{9}, \underline{0}, 7, \underline{13}, \underline{4}, 11, 2, \underline{8}, 15, 6, \underline{12}, \underline{3}, 10, 1, \underline{7}, 14, 5, \underline{11}, \underline{2}, 9$. The underlining of the values concerned is deleted for the half-cycles from lower left to upper right. Note that each value occurs twice, exept $\frac{(2b_m+1)-3}{2} = 15$ of course.

Thus the algorithm diagram for our case with $p = p_m = 7$ has the i-value sequence $0, \underline{6}, 13, 4, \underline{10}, \underline{1}, 8$, and after entering the coding for the intersection-columns, we can read off the half-cycle algorithms for the real string-run.

To braid the Möbius braid in accordance with the half-cycle algorithms obtained, we put in the first revolution (the circumference of the cylinder) of the string half a twist and then follow the same apparent surface of the string. This will ensure that we automatically obtain the required half twists in the string with as end result a correctly braided Regular Möbius Knot.

In order to get a good understanding of the process which determines the half-cycle algorithms, we shall give a few examples involving the algorithm diagram for the 2-pass Headhunter's-coded Regular Möbius Knot with $p_m/b_m = 7/31$, depicted in Fig. 311.

Half-cycle 1:

Half-cycle 1 is always a free run. Half-cycle 1 runs from lower left to upper right.

Half-cycle 2:

Half-cycle 2 is associated with bight-number $i = \frac{2-2}{2} = 0$. Hence from the algorithm diagram we have to read off, for half-cycle 2, the consecutive crossing-movements in accordance with the coding for the bight-numbers which are equal to $i = 0$. Half-cycle 2 runs from lower right to upper left, hence in the algorithm diagram we read the lower line from right to left. This gives us: no crossings, hence: free run.

Half-cycle 3:

Half-cycle 3 is associated with bight-number $i = \frac{3-3}{2} = 0$. Hence from the algorithm diagram we have to read off, for half-cycle 3, the consecutive crossing-movements in accordance with the coding for the bight-numbers which are equal to $i = 0$. Half-cycle 3 runs from lower left to upper right, hence in the algorithm diagram we read the upper line from left to right. This gives us: no crossings, hence: free run.

Half-cycle 4:

Half-cycle 4 is associated with bight-number $i = \frac{4-2}{2} = 1$. Hence from the algorithm diagram we have to read off, for half-cycle 4, the consecutive crossing-movements in accordance with the coding for the bight-numbers which are less than or equal to $i = 1$. Half-cycle 4 runs from lower right to upper left, hence in the algorithm diagram we read the lower line from right to left. However, $\underline{i = 1}$ is underlined, hence should be neglected for this half-cycle only (since it is a crossing of half-cycle 4 with the imaginary half-cycle of higher order). This gives us: no crossings, hence: free run.

Half-cycle 5:

Half-cycle 5 is associated with bight-number $i = \frac{5-3}{2} = 1$. Hence from the algorithm diagram we have to read off, for half-cycle 5, the consecutive crossing-movements in accordance with the coding for the bight-numbers which are less than or equal to $i = 1$. Half-cycle 5 runs from lower left to upper right, hence in the algorithm diagram we read the upper line from left to right. This gives us: *u*.

Half-cycle 6:

Half-cycle 6 is associated with bight-number $i = \frac{6-2}{2} = 2$. Hence from the algorithm diagram we have to read off, for half-cycle 6, the consecutive crossing-movements in accordance with the coding for the bight-numbers which are less than or equal to $i = 2$. Half-cycle 6 runs from lower right to upper left, hence in the algorithm diagram we read the lower line from right to left. This gives us: *o*.

Half-cycle 7:

Half-cycle 7 is associated with bight-number $i = \frac{7-3}{2} = 2$. Hence from the algorithm diagram we have to read off, for half-cycle 7, the consecutive crossing-movements in accordance with the coding for the bight-numbers which are less than or equal to $i = 2$. Half-cycle 7 runs from lower left to upper right, hence in the algorithm diagram we read the upper line from left to right. This gives us: *u*.

Half-cycle 8:

Half-cycle 8 is associated with bight-number $i = \frac{8-2}{2} = 3$. Hence from the algorithm diagram we have to read off, for half-cycle 8, the consecutive crossing-movements in accordance with the coding for the bight-numbers which are less than or equal to $i = 3$. Half-cycle 8 runs from lower right to upper left, hence in the algorithm diagram we read the lower line from right to left. This gives us: *o*.

Half-cycle 9:

Half-cycle 9 is associated with bight-number $i = \frac{9-3}{2} = 3$. Hence from the algorithm diagram we have to read off, for half-cycle 9, the consecutive crossing-movements in accordance with the coding for the bight-numbers which are less than or equal to $i = 3$. Half-cycle 9 runs from lower left to upper right, hence in the algorithm diagram we read the upper line from left to right. This gives us: u.

Half-cycle 10:

Half-cycle 10 is associated with bight-number $i = \frac{10-2}{2} = 4$. Hence from the algorithm diagram we have to read off, for half-cycle 10, the consecutive crossing-movements in accordance with the coding for the bight-numbers which are less than or equal to $i = 4$. Half-cycle 10 runs from lower right to upper left, hence in the algorithm diagram we read the lower line from right to left. This gives us: $u - o$.

Half-cycle 11:

Half-cycle 11 is associated with bight-number $i = \frac{11-3}{2} = 4$. Hence from the algorithm diagram we have to read off, for half-cycle 11, the consecutive crossing-movements in accordance with the coding for the bight-numbers which are less than or equal to $i = 4$. Half-cycle 11 runs from lower left to upper right, hence in the algorithm diagram we read the upper line from left to right. This gives us: $o - u$.

Half-cycle 12:

Half-cycle 12 is associated with bight-number $i = \frac{12-2}{2} = 5$. Hence from the algorithm diagram we have to read off, for half-cycle 12, the consecutive crossing-movements in accordance with the coding for the bight-numbers which are less than or equal to $i = 5$. Half-cycle 12 runs from lower right to upper left, hence in the algorithm diagram we read the lower line from right to left. This gives us: $u - o$.

Half-cycle 13:

Half-cycle 13 is associated with bight-number $i = \frac{13-3}{2} = 5$. Hence from the algorithm diagram we have to read off, for half-cycle 13, the consecutive crossing-movements in accordance with the coding for the bight-numbers which are less than or equal to $i = 5$. Half-cycle 13 runs from lower left to upper right, hence in the algorithm diagram we read the upper line from left to right. This gives us: $o - u$.

Half-cycle 14:

Half-cycle 14 is associated with bight-number $i = \frac{14-2}{2} = 6$. Hence from the algorithm diagram we have to read off, for half-cycle 14, the consecutive crossing-movements in accordance with the coding for the bight-numbers which are less than or equal to $i = 6$. Half-cycle 14 runs from lower right to upper left, hence in the algorithm diagram we read the lower line from right to left. However, $i = 6$ is underlined, hence should be neglected for this half-cycle only (since it is a crossing of half-cycle 14 with the imaginary half-cycle of higher order). This gives us: $u - o$.

Half-cycle 15:

Half-cycle 15 is associated with bight-number $i = \frac{15-3}{2} = 6$. Hence from the algorithm diagram we have to read off, for half-cycle 15, the consecutive crossing-movements in accordance with the coding for the bight-numbers which are less than or equal to $i = 6$. Half-cycle 15 runs from lower left to upper right, hence in the algorithm diagram we read the upper line from left to right. This gives us: $u - o - u$.

Half-cycle 16:

Half-cycle 16 is associated with bight-number $i = \frac{16-2}{2} = 7$. Hence from the algorithm diagram we have to read off, for half-cycle 16, the consecutive crossing-movements in accordance with the coding for the bight-numbers which are less than or equal to $i = 7$. Half-cycle 16 runs from lower right to upper left, hence in the algorithm diagram we read the lower line from right to left. This gives us: $o - u - o$.

Half-cycle 17:

Half-cycle 17 is associated with bight-number $i = \frac{17-3}{2} = 7$. Hence from the algorithm diagram we have to read off, for half-cycle 17, the consecutive crossing-movements in accordance with the coding for the bight-numbers which are less than or equal to $i = 7$. Half-cycle 17 runs from lower left to upper right, hence in the algorithm diagram we read the upper line from left to right. This gives us: $u - o - u$.

Half-cycle 18:

Half-cycle 18 is associated with bight-number $i = \frac{18-2}{2} = 8$. Hence from the algorithm diagram we have to read off, for half-cycle 18, the consecutive crossing-movements in accordance with the coding for the bight-numbers which are less than or equal to $i = 8$. Half-cycle 18 runs from lower right to upper left, hence in the algorithm diagram we read the lower line from right to left. This gives us: $o - u - 2o$.

Half-cycle 19:

Half-cycle 19 is associated with bight-number $i = \frac{19-3}{2} = 8$. Hence from the algorithm diagram we have to read off, for half-cycle 19, the consecutive crossing-movements in accordance with the coding for the bight-numbers which are less than or equal to $i = 8$. Half-cycle 19 runs from lower left to upper right, hence in the algorithm diagram we read the upper line from left to right. This gives us: $u - o - 2u$.

And so on.

$\star\star$ *Prove that for* p_m *and* b_m **odd** *with g.c.d.* $(p_m, b_m) = 1$, *the bight-number* i *associated with the half-cycle (of the imaginary string-run) which coincides with the Standing End half-cycle of the real string-run has the value* $\frac{b_m-1}{2}$.

An arbitrary intersection-column in the algorithm diagram carries $i = a^*$ *for the half-cycles, associated with the imaginary string-run, which run from left to right, and carries* $i = b^*$ *for the half-cycles, associated with the imaginary string-run, which run from right to left:*

$$\begin{array}{ccccccccccccccc} \frac{b_m-1}{2} & . & . & . & . & \dots & . & a^* & . & \dots & . & . & . & & \\ . & . & . & . & . & \dots & . & . & . & \dots & . & . & . & . & \\ & . & . & . & . & \dots & . & b^* & . & \dots & . & . & . & \frac{b_m-1}{2} \end{array}$$

Prove that $a^* + b^* = b_m - 2$.

For Regular Möbius Knots with p_m and b_m **even**, and g.c.d. $(2p_m, [p_m + b_m]) = 2$, the procedure is in essence similar, but the Regular Cylindrical Braid with $p = p_m$ and $b = b_m$ is however a Semi Regular Knot which requires two strings in its construction since the g.c.d. $(p, b) = 2$.

An example of such a Regular Möbius Knot is shown in Fig. 312, depicting the 2-pass Herringbone-coded Regular Möbius Knot with $p_m/b_m = 10/32$.

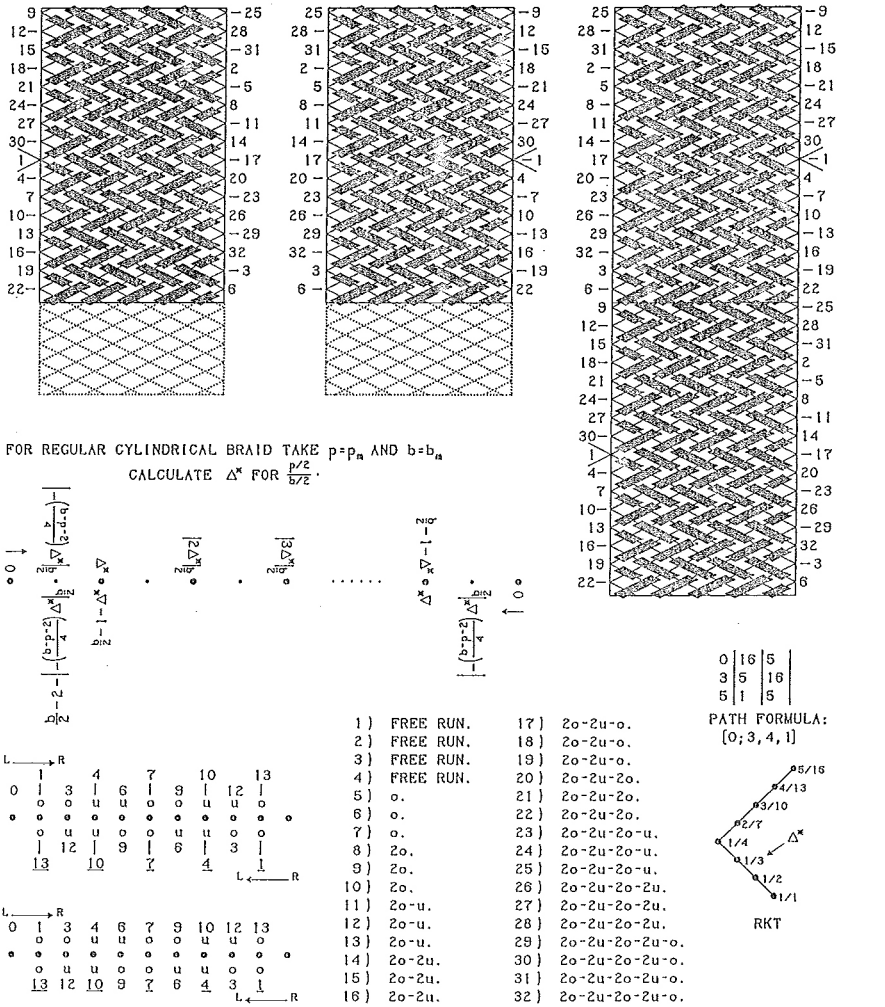


Fig. 312 — A 2-pass Herringbone-coded Regular Möbius Knot with $p_m/b_m = 10/32$.

The *real* string-run and the *imaginary* string-run belong each to a Regular Knot with $p^*/b^* = \frac{p_m}{2}/\frac{b_m}{2} = 5/16$.

With Euclid's algorithm we again calculate the path formula for this $p^*/b^* = \frac{p_m}{2}/\frac{b_m}{2} = 5/16$ Regular Knot, and hence its associated Δ^*-value, which in this case is equal to 3, is readily found.

The imaginary string-run does not go through the bight-point of the Standing End of the real string-run, but goes through the bight-point immediately above it. The half-cycle through this bight-point is associated with bight-number:

$$i = \left|-\left(\frac{b_m - p_m - 2}{4}\right)\Delta^*\right|_{\frac{b_m}{2}} = \left|-\left(\frac{32 - 10 - 2}{4}\right)3\right|_{16} = 1\,.$$

With the Δ^*-value for the Regular Knot with $\frac{p_m}{2}/\frac{b_m}{2} = 5/16$ we can construct the algorithm diagram for the *real* string-run in the usual manner, and obtain the i-value sequence $0, \cdot, 3, \cdot, 6, \cdot, 9, \cdot, 12, \cdot$.

For the *imaginary* string-run, including the imaginary half-cycle of higher order, we obtain the i-value sequence $\cdot, \left|-\left(\frac{b_m - p_m - 2}{4}\right)\Delta^*\right|_{\frac{b_m}{2}} = \underline{1}, \cdot, \underline{4}, \cdot, \underline{7}, \cdot, \underline{10}, \cdot, \underline{13}$.

Both these i-value sequences are combined which results in the i-value sequence $0, \underline{1}, 3, \underline{4}, 6, \underline{7}, 9, \underline{10}, 12, \underline{13}$.

This i-value sequence is then set off in the algorithm diagram for the Möbius braid, but since a real half-cycle cannot intersect the imaginary half-cycle of higher order, we only delete the underlining of the i-values associated with the imaginary string-run for the half-cycles which run from lower left to upper right.

We like to stress again that the 2-pass Herringbone-coded Regular Möbius Knot $p_m/b_m = 10/32$ is produced by the real string-run, and that the imaginary string-run is in fact laid down by the real string-run, but that in the Semi Regular Knot $p/b = 10/32$ the imaginary string-run is laid down by a separate string.

Note that the coding along every odd numbered half-cycle of the real string-run is $//\backslash\backslash//\backslash\backslash/$, and that the coding along every even numbered half-cycle of the real string-run is $\backslash//\backslash\backslash//\backslash\backslash$.

We can either enter these coding sequences or their associated *under* and *over* sequences for the half-cycle direction concerned. After entering these sequences in the algorithm diagram, we can read off the half-cycle algorithms for the real string-run.

In order to braid the Möbius braid in accordance with the half-cycle algorithms obtained, we put in the first revolution (the circumference of the cylinder) of the string half a twist and then follow the same apparent surface of the string. This will ensure that we automatically obtain the required half twists in the string with as end result a correctly braided Regular Möbius Knot.

$\star\star$ *Prove that for* p_m *and* b_m **even** *with g.c.d.* $(2p_m, [p_m + b_m]) = 2$, *the bight-number* i *associated with the half-cycle (of the imaginary string-run) which goes through the bight-point immediately above the bight-point of the Standing End half-cycle of the real string-run has the value* $\left|-(\frac{b_m - p_m - 2}{4})\Delta^*\right|_{\frac{b_m}{2}}$.

An arbitrary intersection-column in the algorithm diagram carries $i = a^*$ *for the half-cycles, associated with the imaginary string-run, which run from left to right, and carries* $i = b^*$ *for the half-cycles, associated with the imaginary string-run, which run from right to left:*

$$\begin{array}{ccccccccccccccc}
\left|-(\frac{b_m - p_m - 2}{4})\Delta^*\right|_{\frac{b_m}{2}} & . & & \cdots & & a^* & & \cdots & & . & & . & \\
. & \updownarrow & . & \updownarrow & . & \cdots & . & \updownarrow & . & \cdots & . & \updownarrow & . & \updownarrow & . \\
 & . & & . & & \cdots & & b^* & & \cdots & & . & \left|-(\frac{b_m - p_m - 2}{4})\Delta^*\right|_{\frac{b_m}{2}}
\end{array}$$

Prove that $a^* + b^* = \frac{b_m}{2} - 2$.

The Braider should be well aware of the pitfalls associated with the braiding procedures for Möbius braids. Especially braiders who work with round braiding material, tend to make the error of omitting the necessary half twists in the string when braiding a Regular Möbius Braid by means of its associated virtual Regular Cylindrical Braid. We have seen that in such cases the braider does **not** obtain a Regular Möbius Braid, but will finish up with a Regular Cylindical Braid which can be transformed into a **false Möbius band** (the band retains its **separate** north, south, east and west surface lines!). This neglect of half twist is also the standard procedure found in the topological knot theory. Hence the fact that the topological knot theory is a bogus theory as far as knots and braids are concerned can readily be demonstrated with a simple Regular Möbius Braid. The topological knot theory is **only** as a purely mathematical subject of any value, and hence of no value to knots and braids. Unfortunately it are the academically brainwashed bogus mathematicians who would like us to believe differently.