Pensieve header: A 4D algebra whose associated knot invariant is Alexander.

In[∘]:=
```
SetDirectory["C:\\drorbn\\AcademicPensieve\\Classes\\21-1350-KnotTheory"];
<< Common.m
```

```
Loading KnotTheory` version of February 2, 2020, 10:53:45.2097.
Read more at http://katlas.org/wiki/KnotTheory.
```

In[∘]:=

$$MT = \begin{pmatrix} & a & b & c & d \\ a & a & b & 0 & 0 \\ b & 0 & 0 & a & b \\ c & c & d & 0 & 0 \\ d & 0 & 0 & c & d \end{pmatrix};$$

```
ℰ_ // m_{i_,j_→k_} := Expand[ℰ] /. Flatten@
    Table[MT⟦α, 1⟧_i MT⟦1, β⟧_j → (MT⟦α, β⟧ /. v : (a | b | c | d) :→ v_k), {α, 2, 5}, {β, 2, 5}];
η_{i_} :=
  a_i +
   d_i;
```

In[∘]:=
```
KBasis[i_] := {a_i, b_i, c_i, d_i};
KBasis[i_, is__] := Flatten@Outer[Times, KBasis[i], KBasis[is]]
```

Associativity of *m*:

In[∘]:=
```
Short[lhs = KBasis[1, 2, 3] // m_{1,2→1} // m_{1,3→1}]
rhs = KBasis[1, 2, 3] // m_{2,3→2} // m_{1,2→1};
lhs == rhs
```

Out[∘]//Short= $\{a_1, b_1, 0, 0, 0, 0, a_1, \ll 51 \gg, 0, 0, 0, 0, c_1, d_1\}$

Out[∘]= True

Units:

In[∘]:=
```
{(η_1 KBasis[2] // m_{1,2→1}) == KBasis[1], (η_1 KBasis[2] // m_{2,1→1}) == KBasis[1]}
```

Out[∘]= {True, True}

In[∘]:=

$$R_{i\_,j\_} := T^{-1/2}\left(a_i a_j + T a_i d_j + (1-T)\, b_j c_i + d_i a_j - T d_i d_j\right);$$

$$\overline{R}_{i\_,j\_} := T^{1/2}\left(a_i a_j + T^{-1} a_i d_j + \left(1-T^{-1}\right) b_j c_i + d_i a_j - T^{-1} d_i d_j\right);$$

$$C_{i\_} := T^{-1/2}\,(a_i - d_i);\quad \overline{C}_{i\_} := T^{1/2}\,(a_i - d_i);$$

Reidemeister 3:

In[∘]:=
```
Short[lhs = R1,2 R4,3 R5,6 // m1,4→1 // m2,5→2 // m3,6→3]
rhs = R2,3 R1,4 R5,6 // m1,5→1 // m2,6→2 // m3,4→3;
lhs == rhs
```

Out[∘]//Short= $\frac{a_1 a_2 a_3}{T^{3/2}} + \ll 29 \gg$

Out[∘]= True

Reidemeister 2b:

In[∘]:=
```
Short[lhs = R1,2 R̄3,4 // m1,3→1 // m2,4→2]
rhs = η1 η2 // Expand;
lhs == rhs
```

Out[∘]//Short= $a_1 a_2 + a_2 d_1 + a_1 d_2 + d_1 d_2$

Out[∘]= True

Naive Reidemeister 2c:

In[∘]:=
```
Short[lhs = R1,2 R̄3,4 // m1,3→1 // m4,2→2]
rhs = η1 η2 // Expand;
Simplify[lhs - rhs]
```

Out[∘]//Short= $a_1 a_2 + 2 b_2 c_1 - 2 T \ll 1 \gg c_1 + \ll 1 \gg + a_1 d_2 + d_1 d_2$

Out[∘]= $-2 \times (-1 + T) b_2 c_1$

Corrected Reidemeister 2c:

In[∘]:=
```
lhs = R1,4 R̄5,2 C̄3 // m2,4→2 // m1,3→1 // m1,5→1
rhs = C̄1 η2 // Expand;
lhs == rhs
```

Out[∘]= $\sqrt{T} a_1 a_2 - \sqrt{T} a_2 d_1 + \sqrt{T} a_1 d_2 - \sqrt{T} d_1 d_2$

Out[∘]= True

$C \bar{C}$:

In[∘]:=
```
C1 C̄2 // m1,2→1
```

Out[∘]= $a_1 + d_1$

Reidemeister 1s:

In[∘]:=
```
{(C̄2 R1,3 // m1,2→1 // m1,3→1) == η1, (C̄2 R̄3,1 // m1,2→1 // m1,3→1) == η1,
 (C2 R̄1,3 // m1,2→1 // m1,3→1) == η1, (C2 R3,1 // m1,2→1 // m1,3→1) == η1}
```

Out[∘]= {True, True, True, True}

The whirl:

In[∘]:=
```
Expand[(C̄1 C̄2 R3,4 C5 C6 // m1,3→1 // m1,5→1 // m2,4→2 // m2,6→2) == R1,2]
```

Out[∘]= True

RVK and Z

RVK, rot, Z modified from 2016-09/OneSmidgen.nb. See also in AP/Projects/SL2Invariant/.

Some details of the code below are at http://drorbn.net/bbs/show?shot=Dror-160920-151350.jpg.

In[•]:=
```
RVK::usage =
  "RVK[xs, rots] represents a Rotational Virtual Knot with a list of n Xp/Xm crossings
    xs and a length 2n list of rotation numbers rots. Crossing
    sites are indexed 1 through 2n, and rots⟦k⟧ is the rotation
    between site k-1 and site k. RVK is also a casting operator
    converting to the RVK presentation from other knot presentations.";
```

In[•]:=
```
RVK[pd_PD] := Module[{n, xs, x, rots, front = {1}, k},
   n = Length@pd; rots = Table[0, {2 n}];
   xs = Cases[pd, x_X :> { Xp[x⟦4⟧, x⟦1⟧]  PositiveQ@x
                           Xm[x⟦2⟧, x⟦1⟧]  True       ];
   For[k = 1, k ≤ 2 n, ++k,
    Echo@{k, front, rots};
    If[FreeQ[front, -k],
     front = Flatten@Replace[front, k → (xs /. {
            Xp[k, l_] | Xm[l_, k] :> {l + 1, k + 1, -l},
            Xp[l_, k] | Xm[k, l_] :> (++rots⟦l⟧; {-l, k + 1, l + 1}),
            _Xp | _Xm :> {}
           }), {1}],
     Cases[front, k | -k] /. {k, -k} :> --rots⟦k⟧;
    ]
   ];
   RVK[xs, rots] ];
RVK[K_] := RVK[PD[K]];
```

In[•]:=
```
RVK[PD[X[4, 2, 5, 1], X[2, 6, 3, 5], X[6, 4, 7, 3]]]
```

```
» {1, {1}, {0, 0, 0, 0, 0, 0}}
» {2, {5, 2, -4}, {0, 0, 0, 0, 0, 0}}
» {3, {5, -5, 3, 6, -4}, {0, 0, 0, 0, 1, 0}}
» {4, {5, -5, 7, 4, -6, 6, -4}, {0, 0, 0, 0, 1, 0}}
» {5, {5, -5, 7, 4, -6, 6, -4}, {0, 0, 0, -1, 1, 0}}
» {6, {5, -5, 7, 4, -6, 6, -4}, {0, 0, 0, -1, 0, 0}}
```

Out[•]=
```
RVK[{Xp[1, 4], Xp[5, 2], Xp[3, 6]}, {0, 0, 0, -1, 0, 0}]
```

In[•]:=
$$\text{rot}_{i\_}[n\_] := \text{rot}_i[n] = \begin{cases} \eta_i & n == 0 \\ C_\$\ \text{rot}_i[n-1] \ // \ m_{i,\$\to i} & n > 0 \\ \overline{C}_\$\ \text{rot}_i[n+1] \ // \ m_{i,\$\to i} & n < 0 \end{cases}$$

```
In[◦]:= Z[K_] := Z[RVK@K];
Z[rvk_RVK] := Module[{ξ, done, st, c, χ, i, j, k},
  ξ = 1; done = {}; st = Range[2 Length[rvk〚1〛]];
  Do[
   {i, j} = List @@ c;
   χ = c /. {_Xp :→ R_{i,j}, _Xm :→ R̄_{i,j}};
   Do[χ = (rot_0[rvk〚2, k〛] χ) // m_{0,k→k}, {k, {i, j}}];
   ξ *= χ;
   Do[
    If[MemberQ[done, k + 1], ξ = ξ // m_{k,k+1→k}; st = st /. k + 1 → k];
    If[MemberQ[done, k - 1], ξ = ξ // m_{st〚k-1〛,k→st〚k-1〛}; st = st /. k → st〚k - 1〛],
    {k, {i, j}}];
   done = done ⋃ {i, j},
   {c, rvk〚1〛}
  ];
  Factor@ξ
 ]
```

```
In[◦]:= K = Knot[8, 17]; Factor@Alexander[K][T]
z = Z[K]
```

```
In[◦]:= RVK[pd_PD] := Module[{n, xs, x, rots, front = {1}, k},
   n = Length@pd; rots = Table[0, {2 n}];
   xs = Cases[pd, x_X :→ { Xp[x〚4〛, x〚1〛]   PositiveQ@x
                           Xm[x〚2〛, x〚1〛]   True        ];
   For[k = 1, k ≤ 2 n, ++k,
    If[FreeQ[front, -k],
     front = Flatten@Replace[front, k → (xs /. {
            Xp[k, l_] | Xm[l_, k] :→ {l + 1, k + 1, -l},
            Xp[l_, k] | Xm[k, l_] :→ (++rots〚l〛; {-l, k + 1, l + 1}),
            _Xp | _Xm :→ {}
           }), {1}],
     Cases[front, k | -k] /. {k, -k} :→ --rots〚k〛;
    ]
   ];
   RVK[xs, rots]];
RVK[K_] := RVK[PD[K]];
```

```
In[◦]:= Timing@Union@Table[Simplify[Alexander[K][T] η_1 == Z[K]], {K, AllKnots[{3, 9}]}]
```

```
In[◦]:= ZF[K_] := Z@ThinPosition@K;
```

```
In[◦]:= Timing@Union@Table[Simplify[Alexander[K][T] η_1 == ZF[K]], {K, AllKnots[{3, 9}]}]
```

```
In[◦]:= Timing@Union@Table[Simplify[Alexander[K][T] η_1 == ZF[K]], {K, AllKnots[{3, 10}]}]
```

```
In[ ]:= Timing[ZF[GST48]]
```