

Pensieve header: RVK and Z. Continues pensieve://Projects/SL2Portfolio2/.

RVK and Z

RVK, rot, Z 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.";
```

```
RVK[pd_PD] := PP_RVK@Module[{n, xs, x, rots, front = {0}, 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 = 0, k < 2 n, ++k, If[k == 0 ∨ FreeQ[front, -k],
    front = Flatten@Replace[front, k -> {xs /. {
      Xp[k + 1, L_] | Xm[L_, k + 1] => {L, k + 1, 1 - L},
      Xp[L_, k + 1] | Xm[k + 1, L_] => {++rots[[L]],
        {1 - L, k + 1, L}},
      _Xp | _Xm => {}
    }}, {1}],
    Cases[front, k | -k] /. {k, -k} => --rots[[k + 1]]];
  ];
  RVK[xs, rots] ];
```

```
RVK[K_] := RVK[PD[K]];
```

```
rot[i_, 0] := dηi;
rot[i_, n_] := Module[{j},
  rot[i, n] = If[n > 0, rot[i, n - 1] kCj, rot[i, n + 1]  $\overline{kC_j}$ ] // kmi,j→i];
```

```

Z[K_] := Z[RVK@K];
Z[rvk_RVK] := (*Z[rvk] ==*)
Monitor[PP["Z"]@Module[{todo, n, rots,  $\xi$ , done, st, cx,  $\xi_1$ , i, j, k, k1, k2, k3},
  {todo, rots} = List@@rvk;
  AppendTo[rots, 0];
  n = Length[todo];
   $\xi$  =  $d\eta_0$ ;
  done = {0};
  st = Range[0, 2 n + 1];
  While[{ } != ($M = todo),
    {cx} = MaximalBy[todo, Length[done  $\cap$  {#[[1]], #[[2]], #[[1]] - 1, #[[2]] - 1}] &, 1];
    {i, j} = List@@cx;
     $\xi_1$  = Switch[Head[cx],
      Xp, ( $kR_{i,j}$   $\overline{kKink_k}$ ) //  $km_{j,k \rightarrow j}$ ,
      Xm, ( $\overline{kR_{i,j}}$   $kKink_k$ ) //  $km_{j,k \rightarrow j}$ 
    ];
     $\xi_1$  = (rot[k, rots[[i]]]  $\xi_1$ ) //  $km_{k,i \rightarrow i}$ ; rots[[i]] = 0;
     $\xi_1$  = ( $\xi_1$  rot[k, rots[[i + 1]]]) //  $km_{i,k \rightarrow i}$ ; rots[[i + 1]] = 0;
     $\xi_1$  = (rot[k, rots[[j]]]  $\xi_1$ ) //  $km_{k,j \rightarrow j}$ ; rots[[j]] = 0;
     $\xi_1$  = ( $\xi_1$  rot[k, rots[[j + 1]]]) //  $km_{j,k \rightarrow j}$ ; rots[[j + 1]] = 0;
     $\xi$  *=  $\xi_1$ ;
    If[MemberQ[done, i],  $\xi$  =  $\xi$  //  $km_{i,i+1 \rightarrow i}$ ; st = st /. st[[i + 2]]  $\rightarrow$  st[[i + 1]]];
    If[MemberQ[done, i - 1],  $\xi$  =  $\xi$  //  $km_{st[[i]], i \rightarrow st[[i]]}$ ; st = st /. st[[i + 1]]  $\rightarrow$  st[[i]]];
    If[MemberQ[done, j],  $\xi$  =  $\xi$  //  $km_{j,j+1 \rightarrow j}$ ; st = st /. st[[j + 2]]  $\rightarrow$  st[[j + 1]]];
    If[MemberQ[done, j - 1],  $\xi$  =  $\xi$  //  $km_{st[[j]], j \rightarrow st[[j]]}$ ; st = st /. st[[j + 1]]  $\rightarrow$  st[[j]]];
    done = done  $\cup$  {i - 1, i, j - 1, j};
    todo = DeleteCases[todo, cx]
  ];
  CF /@ ( $\xi$  /. { $x_0 \rightarrow x$ ,  $y_0 \rightarrow y$ ,  $a_0 \rightarrow a$ })
], $M]

```