The Alexander Polynomial for Tangles

Pensieve Header: This is a commented version of the pA program, computing the Alexander polynomial for w-tangles using half-densities, along with testing and verification of some relations. September 2008, Joint with Jana Archibald.

```
<< KnotTheory`
```

```
Loading KnotTheory` version of August 13, 2008, 14:31:13.4448.
Read more at http://katlas.org/wiki/KnotTheory.
```

The Program

The Circuit Algebra

■ Variable Equivalences

Equal[v1, v2, ...] Equal[w1, w2, ...] ... declares that v1 == v2 == ..., w1 == w2 == ..., etc.
ReductionRules[eqs] produces a list of reduction rules that reduce expressions modulo the given equivalences eqs.

```
ReductionRules[Times[]] = {};
ReductionRules[Equal[a_, b__]] := (# → a) & /@ {b};
ReductionRules[eqs_Times] := Join @@ (ReductionRules /@ List @@ eqs)
```

■ Some Exterior Algebra Operations

W[i1, i2, ...] denotes an exterior product with indices i1, i2, ...
WExpand[expr] sorts all the W' s in expr.
a~WM~b wedge multiplies a and b; WM[a,b,c,...] wedge multiplies a, b, c,...
IM[i, W[...]] inner multiplies W[...] by i. i may be replaced by a list, and W by an arbitrary linear combinations of W's.

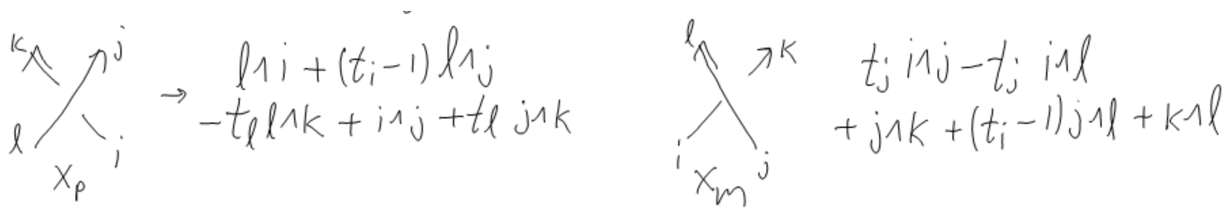
```
WExpand[expr_] := Expand[expr /. w_W :> Signature[w] * Sort[w]];
WM[___, 0, ___] = 0;
a_ ~WM~ b_ := WExpand[Distribute[a ** b] /.
    (c1_. * w1_W) ** (c2_. * w2_W) :> c1 c2 Join[w1, w2]
  ];
WM[a_, b_, c__] := a ~WM~ WM[b, c];
IM[{}, expr_] := expr;
IM[i_, w_W] := If[MemberQ[w, i], -(-1)^Position[w, i][[1, 1]] DeleteCases[w, i], 0];
IM[{is___, i_}, w_W] := IM[{is}, IM[i, w]];
IM[is_List, expr_] := expr /. w_W :> IM[is, w]
```

■ Alexander Half - Densities

AHD[{i1, ...}, W[o1, ...], eqs, p] denotes an Alexander half - density with incoming legs i1, ... (Bosons), outgoing legs o1, ... (Fermions), variable equivalences eqs, and payload the half - density p.
AHD[...]*AHD[...] circuit-multiplies two Alexander half-densities.

```
AHD[is_, -os_, eqs_, p_] := AHD[is, os, eqs, Expand[-p]];
AHD /: Reduce[AHD[is_, os_, eqs_, p_]] :=
  AHD[Sort[is], WExpand[os], eqs, WExpand[p /. ReductionRules[eqs]]];
AHD /: AHD[is1_, os1_, eqs1_, p1_] * AHD[is2_, os2_, eqs2_, p2_] := Module[
  {glued},
  glued = Union[Intersection[is1, List @@ os2], Intersection[is2, List @@ os1]];
  Reduce[AHD[
    Complement[Union[is1, is2], glued],
    IM[glued, os1~WM~os2],
    eqs1 * eqs2 //. eq1_Equal * eq2_Equal /;
       Intersection[List @@ eq1, List @@ eq2] =!= {} :> Union[eq1, eq2],
    IM[glued, p1~WM~p2]
   ]]
 ]
```

■ The Basic Components




```
AT[Xp[i_, j_, k_, l_]] := AHD[{i, l}, W[j, k], (t_i == t_k) (t_j == t_l),
   W[l, i] + (t_i - 1) W[l, j] - t_l W[l, k] + W[i, j] + t_l W[j, k]
  ];
AT[Xm[i_, j_, k_, l_]] := AHD[{i, j}, W[k, l], (t_i == t_k) (t_j == t_l),
  t_j W[i, j] - t_j W[i, l] + W[j, k] + (t_i - 1) W[j, l] + W[k, l]
 ]
```

■ The Overall Invariant

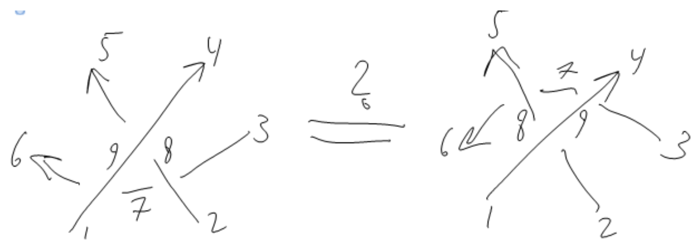
This part of the program is modeled after the "faster Jones" program of http://katlas.org/wiki/The_Jones_Polynomial#How_is_the_Jones_polynomial_computed?

The definition "AT[cd_CircuitDiagram] := Times @@ (AT /@ cd)" would work too, only slower.

```
AT[cd_CircuitDiagram] := AT[cd, {}, 1];
AT[cd_CircuitDiagram, inside_, ahd_] := Module[
   {pos = First[Ordering[Length[Complement[List @@ #, inside]] & /@ cd]]},
   AT[
    Delete[cd, pos],
    Union[inside, List @@ cd[[pos]]],
    ahd * AT[cd[[pos]]]
   ]
  ];
AT[CircuitDiagram[], _, ahd_] := ahd
```

Some Relations

■ Reidemeister 3



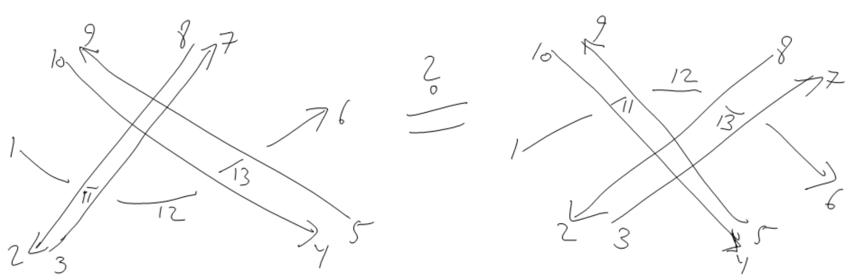
```
res1 = AT /@ {
   CircuitDiagram[Xp[7, 9, 6, 1], Xp[3, 8, 7, 2], Xp[8, 4, 5, 9]],
   CircuitDiagram[Xp[3, 4, 7, 9], Xp[7, 5, 6, 8], Xp[2, 9, 8, 1]]
  }
```

$$\{\mathrm{AHD}[\{1,2,3\}, W[4,5,6], (t_1 == t_4 == t_9)\,(t_2 == t_5 == t_8)\,(t_3 == t_6 == t_7), -W[1,2,3] + W[1,2,4] - t_3 W[1,2,4] + t_1 W[1,2,5] - t_1 t_3 W[1,2,5] + t_1 t_2 W[1,2,6] - W[1,3,4] + t_2 W[1,3,4] - t_1 W[1,3,5] + t_1 t_2 W[1,4,5] - t_1 t_2 t_3 W[1,4,5] - t_1 t_2 W[1,4,6] + t_1 t_2^2 W[1,4,6] - t_1^2 t_2 W[1,5,6] + W[2,3,4] + t_1 W[2,4,5] - t_1 t_3 W[2,4,5] + t_1 t_2 W[2,4,6] - t_1 W[3,4,5] + t_1^2 t_2 W[4,5,6]],$$
$$\mathrm{AHD}[\{1,2,3\}, W[4,5,6], (t_1 == t_4 == t_9)\,(t_2 == t_5 == t_8)\,(t_3 == t_6 == t_7), -W[1,2,3] + W[1,2,4] - t_3 W[1,2,4] + t_1 W[1,2,5] - t_1 t_3 W[1,2,5] + t_1 t_2 W[1,2,6] - W[1,3,4] + t_2 W[1,3,4] - t_1 W[1,3,5] + t_1 t_2 W[1,4,5] - t_1 t_2 t_3 W[1,4,5] - t_1 t_2 W[1,4,6] + t_1 t_2^2 W[1,4,6] - t_1^2 t_2 W[1,5,6] + W[2,3,4] + t_1 W[2,4,5] - t_1 t_3 W[2,4,5] + t_1 t_2 W[2,4,6] - t_1 W[3,4,5] + t_1^2 t_2 W[4,5,6]]\}$$

```
Equal @@ (Last /@ res1)
```

```
True
```

■ Virtual Double Delta



```
res2 = AT /@ {
   CircuitDiagram[Xp[1, 2, 11, 8], Xm[11, 3, 12, 7], Xp[12, 4, 13, 10], Xm[13, 5, 6, 9]],
   CircuitDiagram[Xp[1, 4, 11, 10], Xm[11, 5, 12, 9], Xp[12, 2, 13, 8], Xm[13, 3, 6, 7]]
  }
```

```
{AHD[{1, 3, 5, 8, 10}, W[2, 4, 6, 7, 9],
  (t2 == t8) (t3 == t7) (t4 == t10) (t5 == t9) (t1 == t6 == t11 == t12 == t13),
  t3 t5 W[1, 2, 3, 4, 5] - t3 t5 W[1, 2, 3, 4, 9] + t3 t5 W[1, 2, 3, 5, 10] - t3 t5 W[1, 2, 3, 9, 10] -
   t3 t5 W[1, 2, 4, 5, 7] - t3 t5 W[1, 2, 4, 7, 9] + t3 t5 W[1, 2, 5, 7, 10] + t3 t5 W[1, 2, 7, 9, 10] +
   t3 t5 W[1, 3, 4, 5, 8] + t3 t5 W[1, 3, 4, 8, 9] - t3 t5 W[1, 3, 5, 8, 10] - t3 t5 W[1, 3, 8, 9, 10] -
   t3 t5 W[1, 4, 5, 7, 8] + t3 t5 W[1, 4, 7, 8, 9] - t3 t5 W[1, 5, 7, 8, 10] + t3 t5 W[1, 7, 8, 9, 10] -
   t2 t4 W[2, 3, 4, 5, 6] + t2 t5 W[2, 3, 4, 5, 7] - t1 t2 t5 W[2, 3, 4, 5, 7] - t3 t5 W[2, 3, 4, 5, 8] +
   t1 t3 t5 W[2, 3, 4, 5, 8] + t2 t4 W[2, 3, 4, 5, 9] - t1 t2 t4 W[2, 3, 4, 5, 9] -
   t2 t5 W[2, 3, 4, 5, 10] + t1 t2 t5 W[2, 3, 4, 5, 10] - t2 t4 W[2, 3, 4, 6, 9] +
   t2 t5 W[2, 3, 4, 7, 9] - t1 t2 t5 W[2, 3, 4, 7, 9] - t3 t5 W[2, 3, 4, 8, 9] +
   t1 t3 t5 W[2, 3, 4, 8, 9] + t2 t5 W[2, 3, 4, 9, 10] - t1 t2 t5 W[2, 3, 4, 9, 10] +
   t2 t4 W[2, 3, 5, 6, 10] - t2 t5 W[2, 3, 5, 7, 10] + t1 t2 t5 W[2, 3, 5, 7, 10] +
   t3 t5 W[2, 3, 5, 8, 10] - t1 t3 t5 W[2, 3, 5, 8, 10] - t2 t4 W[2, 3, 5, 9, 10] +
   t1 t2 t4 W[2, 3, 5, 9, 10] + t2 t4 W[2, 3, 6, 9, 10] - t2 t5 W[2, 3, 7, 9, 10] +
   t1 t2 t5 W[2, 3, 7, 9, 10] + t3 t5 W[2, 3, 8, 9, 10] - t1 t3 t5 W[2, 3, 8, 9, 10] -
   t2 t4 W[2, 4, 5, 6, 7] + t3 t5 W[2, 4, 5, 7, 8] - t1 t3 t5 W[2, 4, 5, 7, 8] - t2 t4 W[2, 4, 5, 7, 9] +
   t1 t2 t4 W[2, 4, 5, 7, 9] + t2 t5 W[2, 4, 5, 7, 10] - t1 t2 t5 W[2, 4, 5, 7, 10] +
   t2 t4 W[2, 4, 6, 7, 9] - t3 t5 W[2, 4, 7, 8, 9] + t1 t3 t5 W[2, 4, 7, 8, 9] + t2 t5 W[2, 4, 7, 9, 10] -
   t1 t2 t5 W[2, 4, 7, 9, 10] - t2 t4 W[2, 5, 6, 7, 10] + t3 t5 W[2, 5, 7, 8, 10] -
   t1 t3 t5 W[2, 5, 7, 8, 10] - t2 t4 W[2, 5, 7, 9, 10] + t1 t2 t4 W[2, 5, 7, 9, 10] +
   t2 t4 W[2, 6, 7, 9, 10] - t3 t5 W[2, 7, 8, 9, 10] + t1 t3 t5 W[2, 7, 8, 9, 10] + t2 t4 W[3, 4, 5, 6, 8] -
   t2 t5 W[3, 4, 5, 7, 8] + t1 t2 t5 W[3, 4, 5, 7, 8] + t2 t4 W[3, 4, 5, 8, 9] - t1 t2 t4 W[3, 4, 5, 8, 9] -
   t2 t5 W[3, 4, 5, 8, 10] + t1 t2 t5 W[3, 4, 5, 8, 10] - t2 t4 W[3, 4, 6, 8, 9] + t2 t5 W[3, 4, 7, 8, 9] -
   t1 t2 t5 W[3, 4, 7, 8, 9] - t2 t5 W[3, 4, 8, 9, 10] + t1 t2 t5 W[3, 4, 8, 9, 10] +
   t2 t4 W[3, 5, 6, 8, 10] - t2 t5 W[3, 5, 7, 8, 10] + t1 t2 t5 W[3, 5, 7, 8, 10] +
   t2 t4 W[3, 5, 8, 9, 10] - t1 t2 t4 W[3, 5, 8, 9, 10] - t2 t4 W[3, 6, 8, 9, 10] +
   t2 t5 W[3, 7, 8, 9, 10] - t1 t2 t5 W[3, 7, 8, 9, 10] + t2 t4 W[4, 5, 6, 7, 8] - t2 t4 W[4, 5, 7, 8, 9] +
   t1 t2 t4 W[4, 5, 7, 8, 9] + t2 t5 W[4, 5, 7, 8, 10] - t1 t2 t5 W[4, 5, 7, 8, 10] +
   t2 t4 W[4, 6, 7, 8, 9] - t2 t5 W[4, 7, 8, 9, 10] + t1 t2 t5 W[4, 7, 8, 9, 10] - t2 t4 W[5, 6, 7, 8, 10] +
   t2 t4 W[5, 7, 8, 9, 10] - t1 t2 t4 W[5, 7, 8, 9, 10] - t2 t4 W[6, 7, 8, 9, 10]], AHD[{1, 3, 5, 8, 10},
  W[2, 4, 6, 7, 9], (t2 == t8) (t3 == t7) (t4 == t10) (t5 == t9) (t1 == t6 == t11 == t12 == t13),
  t3 t5 W[1, 2, 3, 4, 5] - t3 t5 W[1, 2, 3, 4, 9] + t3 t5 W[1, 2, 3, 5, 10] - t3 t5 W[1, 2, 3, 9, 10] -
   t3 t5 W[1, 2, 4, 5, 7] - t3 t5 W[1, 2, 4, 7, 9] + t3 t5 W[1, 2, 5, 7, 10] + t3 t5 W[1, 2, 7, 9, 10] +
   t3 t5 W[1, 3, 4, 5, 8] + t3 t5 W[1, 3, 4, 8, 9] - t3 t5 W[1, 3, 5, 8, 10] - t3 t5 W[1, 3, 8, 9, 10] -
   t3 t5 W[1, 4, 5, 7, 8] + t3 t5 W[1, 4, 7, 8, 9] - t3 t5 W[1, 5, 7, 8, 10] + t3 t5 W[1, 7, 8, 9, 10] -
   t2 t4 W[2, 3, 4, 5, 6] + t2 t4 W[2, 3, 4, 5, 7] - t1 t2 t4 W[2, 3, 4, 5, 7] - t3 t4 W[2, 3, 4, 5, 8] +
   t1 t3 t4 W[2, 3, 4, 5, 8] + t3 t4 W[2, 3, 4, 5, 9] - t1 t3 t4 W[2, 3, 4, 5, 9] -
   t3 t5 W[2, 3, 4, 5, 10] + t1 t3 t5 W[2, 3, 4, 5, 10] - t2 t4 W[2, 3, 4, 6, 9] +
   t2 t4 W[2, 3, 4, 7, 9] - t1 t2 t4 W[2, 3, 4, 7, 9] - t3 t4 W[2, 3, 4, 8, 9] +
   t1 t3 t4 W[2, 3, 4, 8, 9] + t3 t5 W[2, 3, 4, 9, 10] - t1 t3 t5 W[2, 3, 4, 9, 10] +
   t2 t4 W[2, 3, 5, 6, 10] - t2 t4 W[2, 3, 5, 7, 10] + t1 t2 t4 W[2, 3, 5, 7, 10] +
   t3 t4 W[2, 3, 5, 8, 10] - t1 t3 t4 W[2, 3, 5, 8, 10] - t3 t4 W[2, 3, 5, 9, 10] +
   t1 t3 t4 W[2, 3, 5, 9, 10] + t2 t4 W[2, 3, 6, 9, 10] - t2 t4 W[2, 3, 7, 9, 10] +
   t1 t2 t4 W[2, 3, 7, 9, 10] + t3 t4 W[2, 3, 8, 9, 10] - t1 t3 t4 W[2, 3, 8, 9, 10] -
   t2 t4 W[2, 4, 5, 6, 7] + t3 t4 W[2, 4, 5, 7, 8] - t1 t3 t4 W[2, 4, 5, 7, 8] - t3 t4 W[2, 4, 5, 7, 9] +
```

$t_1 t_3 t_4 W[2, 4, 5, 7, 9] + t_3 t_5 W[2, 4, 5, 7, 10] - t_1 t_3 t_5 W[2, 4, 5, 7, 10] +$
$t_2 t_4 W[2, 4, 6, 7, 9] - t_3 t_4 W[2, 4, 7, 8, 9] + t_1 t_3 t_4 W[2, 4, 7, 8, 9] + t_3 t_5 W[2, 4, 7, 9, 10] -$
$t_1 t_3 t_5 W[2, 4, 7, 9, 10] - t_2 t_4 W[2, 5, 6, 7, 10] + t_3 t_4 W[2, 5, 7, 8, 10] -$
$t_1 t_3 t_4 W[2, 5, 7, 8, 10] - t_3 t_4 W[2, 5, 7, 9, 10] + t_1 t_3 t_4 W[2, 5, 7, 9, 10] +$
$t_2 t_4 W[2, 6, 7, 9, 10] - t_3 t_4 W[2, 7, 8, 9, 10] + t_1 t_3 t_4 W[2, 7, 8, 9, 10] + t_2 t_4 W[3, 4, 5, 6, 8] -$
$t_2 t_4 W[3, 4, 5, 7, 8] + t_1 t_2 t_4 W[3, 4, 5, 7, 8] + t_3 t_4 W[3, 4, 5, 8, 9] - t_1 t_3 t_4 W[3, 4, 5, 8, 9] -$
$t_3 t_5 W[3, 4, 5, 8, 10] + t_1 t_3 t_5 W[3, 4, 5, 8, 10] - t_2 t_4 W[3, 4, 6, 8, 9] + t_2 t_4 W[3, 4, 7, 8, 9] -$
$t_1 t_2 t_4 W[3, 4, 7, 8, 9] - t_3 t_5 W[3, 4, 8, 9, 10] + t_1 t_3 t_5 W[3, 4, 8, 9, 10] +$
$t_2 t_4 W[3, 5, 6, 8, 10] - t_2 t_4 W[3, 5, 7, 8, 10] + t_1 t_2 t_4 W[3, 5, 7, 8, 10] +$
$t_3 t_4 W[3, 5, 8, 9, 10] - t_1 t_3 t_4 W[3, 5, 8, 9, 10] - t_2 t_4 W[3, 6, 8, 9, 10] +$
$t_2 t_4 W[3, 7, 8, 9, 10] - t_1 t_2 t_4 W[3, 7, 8, 9, 10] + t_2 t_4 W[4, 5, 6, 7, 8] - t_3 t_4 W[4, 5, 7, 8, 9] +$
$t_1 t_3 t_4 W[4, 5, 7, 8, 9] + t_3 t_5 W[4, 5, 7, 8, 10] - t_1 t_3 t_5 W[4, 5, 7, 8, 10] +$
$t_2 t_4 W[4, 6, 7, 8, 9] - t_3 t_5 W[4, 7, 8, 9, 10] + t_1 t_3 t_5 W[4, 7, 8, 9, 10] - t_2 t_4 W[5, 6, 7, 8, 10] +$
$t_3 t_4 W[5, 7, 8, 9, 10] - t_1 t_3 t_4 W[5, 7, 8, 9, 10] - t_2 t_4 W[6, 7, 8, 9, 10]]\}$

```
{1, -1}.(Last /@ res2) /. {t5 → t4, t3 → t2}
```

```
0
```

Just for fun, let' s try that again with all crossings flipped :

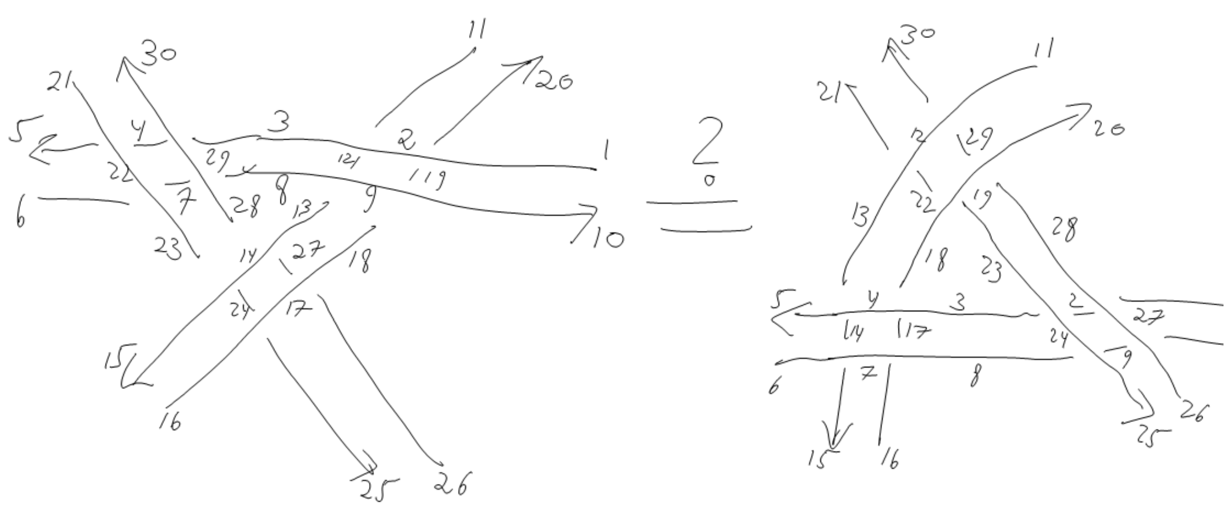
```
VDDFlipped = {
   CircuitDiagram[Xp[1, 2, 11, 8], Xm[11, 3, 12, 7], Xp[12, 4, 13, 10], Xm[13, 5, 6, 9]],
   CircuitDiagram[Xp[1, 4, 11, 10], Xm[11, 5, 12, 9], Xp[12, 2, 13, 8], Xm[13, 3, 6, 7]]
  } /. {Xp[i_, j_, k_, l_] :→ Xm[l, i, j, k], Xm[i_, j_, k_, l_] :→ Xp[j, k, l, i]}
```

```
{CircuitDiagram[Xm[8, 1, 2, 11], Xp[3, 12, 7, 11], Xm[10, 12, 4, 13], Xp[5, 6, 9, 13]],
 CircuitDiagram[Xm[10, 1, 4, 11], Xp[5, 12, 9, 11], Xm[8, 12, 2, 13], Xp[3, 6, 7, 13]]}
```

```
res3 = (AT /@ VDDFlipped);
{1, -1}.(Last /@ res3) /. {t5 → t4, t3 → t2}
```

```
0
```

- **Double Delta**



```
res4 = AT /@ {
   CircuitDiagram[
    Xm[19, 1, 20, 2], Xp[11, 3, 12, 2], Xp[3, 30, 4, 29], Xm[4, 21, 5, 22],
    Xp[6, 23, 7, 22], Xm[7, 28, 8, 29], Xm[12, 8, 13, 9], Xp[18, 10, 19, 9],
    Xm[27, 13, 28, 14], Xp[23, 15, 24, 14], Xm[24, 16, 25, 17], Xp[26, 18, 27, 17]
   ],
   CircuitDiagram[
    Xp[1, 28, 2, 27], Xm[2, 23, 3, 24], Xm[17, 3, 18, 4], Xp[13, 5, 14, 4],
    Xm[14, 6, 15, 7], Xp[16, 8, 17, 7], Xp[8, 25, 9, 24], Xm[9, 26, 10, 27],
    Xm[29, 11, 30, 12], Xp[21, 13, 22, 12], Xm[22, 18, 23, 19], Xp[28, 20, 29, 19]
   ]
  }
```

A very large output was generated. Here is a sample of it:

```
{AHD[{1, 6, 11, 16, 21, 26}, W[5, 10, 15, 20, 25, 30],
  (t_1 == t_2 == t_3 == t_4 == t_5) (<<1>>) <<2>> (<<1>>) (<<1>>),
  -t_6^2 t_16^2 t_26^2 W[1, 5, 6, 11, 15, 21] + t_6^2 t_16^3 t_26^2 W[1, 5, 6, 11, 15, 21] + <<3582>> +
   t_1 t_6 t_11^2 t_21^2 W[10, 15, 20, 25, 26, 30] - t_1^2 t_6 t_11^2 t_21^2 W[10, 15, 20, 25, 26, 30]], <<1>>}
```

Show Less | Show More | Show Full Output | Set Size Limit...

```
{1, -1}.(Last /@ res4) /. {t_6 → t_1, t_16 → t_11, t_26 → t_21}
```

```
0
```

Testing

```
ReductionRules[Equiv[t[1], t[2]] Equiv[t[2], t[3]]]
```

```
ReductionRules[Equiv[t[1], t[2]], Equiv[t[2], t[3]]]
```

```
ReductionRules[Equiv[t[1], t[2]] Equiv[t[2], t[3]] Equiv[t[5], t[6]]]
```

```
ReductionRules[Equiv[t[1], t[2]], Equiv[t[2], t[3]], Equiv[t[5], t[6]]]
```

```
WExpand[W[1, 2] + W[2, 1]]
```

```
0
```

```
WM[W[1, 2] + W[3, 4], 3 W[2, 5]]
```

```
3 W[2, 3, 4, 5]
```

```
{IM[3, W[2, 3, 5]], IM[4, W[2, 3, 5]]}
```

```
{-W[2, 5], 0}
```

```
{IM[{4, 2}, W[3, 4] + 2 W[3, 2, 4]], IM[{2, 4}, W[3, 4] + 2 W[3, 2, 4]]}
```

```
{2 W[3], -2 W[3]}
```

```
Reduce[AHD[{2, 1}, W[4, 3], t[1] == t[2], t[2] W[5, 6]]]
```

```
AHD[{1, 2}, W[3, 4], t[1] == t[2], -t[1] W[5, 6]]
```

```
AHD[{1}, W[2], 1, W[1]] AHD[{2}, W[1], 1, W[2]]
```

```
AHD[{}, W[], 1, -W[]]
```

```
PD[Mirror[Knot[3, 1]]]
```

KnotTheory::loading : Loading precomputed data in PD4Knots`.

```
PD[X[4, 2, 5, 1], X[6, 4, 1, 3], X[2, 6, 3, 5]]
```

```
Times @@ (AT /@ (PD[X[4, 2, 5, 1], X[6, 4, 7, 3], X[2, 6, 3, 5]] /. X → Xp))
```

AHD[{1}, W[7], $t_1 == t_2 == t_3 == t_4 == t_5 == t_6 == t_7$,
$-t_1$ W[1] + t_1^2 W[1] − t_1^3 W[1] + t_1 W[7] − t_1^2 W[7] + t_1^3 W[7]]

```
pd = PD[TorusKnot[5, 4]]
```

```
PD[X[17, 25, 18, 24], X[10, 26, 11, 25], X[3, 27, 4, 26], X[11, 19, 12, 18], X[4, 20, 5, 19],
 X[27, 21, 28, 20], X[5, 13, 6, 12], X[28, 14, 29, 13], X[21, 15, 22, 14], X[29, 7, 30, 6],
 X[22, 8, 23, 7], X[15, 9, 16, 8], X[23, 1, 24, 30], X[16, 2, 17, 1], X[9, 3, 10, 2]]
```

```
n = 2 Length[pd];
pd = pd /. {
   X[1, i_, n, j_] :→ X[n + 1, i, n, j], X[i_, 1, j_, n] :→ X[i, n + 1, j, n],
   X[n, i_, 1, j_] :→ X[n, i, n + 1, j], X[i_, n, j_, 1] :→ X[i, n, j, n + 1]
  }
```

```
PD[X[17, 25, 18, 24], X[10, 26, 11, 25], X[3, 27, 4, 26], X[11, 19, 12, 18], X[4, 20, 5, 19],
 X[27, 21, 28, 20], X[5, 13, 6, 12], X[28, 14, 29, 13], X[21, 15, 22, 14], X[29, 7, 30, 6],
 X[22, 8, 23, 7], X[15, 9, 16, 8], X[23, 31, 24, 30], X[16, 2, 17, 1], X[9, 3, 10, 2]]
```

```
AT[CircuitDiagram @@ pd /. X → Xp]
```

AHD[{1}, W[31], $t_1 == t_2 == t_3 == t_4 == t_5 == t_6 == t_7 == t_8 == t_9 == t_{10} == t_{11} == t_{12} == t_{13} == t_{14} == t_{15} ==$
$t_{16} == t_{17} == t_{18} == t_{19} == t_{20} == t_{21} == t_{22} == t_{23} == t_{24} == t_{25} == t_{26} == t_{27} == t_{28} == t_{29} == t_{30} == t_{31}$,
$-t_1^2$ W[1] + t_1^3 W[1] − t_1^6 W[1] + t_1^8 W[1] − t_1^{10} W[1] + t_1^{13} W[1] − t_1^{14} W[1] + t_1^2 W[31] −
t_1^3 W[31] + t_1^6 W[31] − t_1^8 W[31] + t_1^{10} W[31] − t_1^{13} W[31] + t_1^{14} W[31]]

```
Test[pd_PD] := Module[
   {n = 2 Length[pd]},
   Cancel[
    (Last[AT[
        CircuitDiagram @@ pd /. {
           X[1, i_, n, j_] :→ X[n + 1, i, n, j], X[i_, 1, j_, n] :→ X[i, n + 1, j, n],
           X[n, i_, 1, j_] :→ X[n, i, n + 1, j], X[i_, n, j_, 1] :→ X[i, n, j, n + 1]
          } /. x_X :→ If[PositiveQ[x], Xp @@ x, Xm @@ x]
       ]] /. {W[n + 1] → 1, W[1] → 0, t1 → t})
     / Alexander[pd][t]
   ]
  ];
Test[L_] := Test[PD[L]]
```

```
Test[Knot[5, 2]]
```

t^3

```
Test /@ AllKnots[{3, 9}]
```

$\{t^2, t^3, t^3, t^3, t^4, t^4, t^3, t^4, t^4, t^4, t^4, t^4, t^4, t^5, t^5, t^5, t^5, t^5, t^5, t^5,$
$t^4, t^4, t^5, t^4, t^5, t^5, t^4, t^5, t^5, t^4, t^5, t^4, t^4, t^5, t^5, t^5, t^5, t^5, t^5, t^5, t^5,$
$t^5, t^5, t^5, t^5, t^5, t^5, t^5, t^6, t^5, t^5, t^6, t^5, t^6, t^5, t^5, t^6, t^5, t^5, t^5, t^6, t^4,$
$t^5, t^6, t^5, t^5, t^6, t^5, t^5, t^5, t^5, t^6, t^5, t^6, t^6, t^6, t^6, t^5, t^5, t^5, t^6, t^5, t^6, t^6\}$