

Contractions!

```

c_{x,y}_[w_wedge] := Module[{i, j},
  {i} = FirstPosition[w, x, {0}]; {j} = FirstPosition[w, y, {0}];
  {
    w (i == 0) ^ (j == 0)
    (-1)^{i+j+If{i>j,0,1}} Delete[w, {{i}, {j}}] (i > 0) ^ (j > 0)
  };
c_{x,y}_[e_] := e /. w_wedge -> c_{x,y}_[w]
WExp[a ^ b + 2 c ^ d]
c_{a,c}@WExp[a ^ b + 2 c ^ d]
Wedge[] + a ^ b + 2 c ^ d + 2 a ^ b ^ c ^ d
-Wedge[] - a ^ b

```

$\mathcal{A}[is, os, cs, w]$ is also a container for the values of the \mathcal{A} -invariant of a tangle. In it, is are the labels of the input strands, os are the labels of the output strands, cs is an assignment of colours (namely, variables) to all the ends $\{\xi_i\}_{i \in is} \sqcup \{\chi_j\}_{j \in os}$, and w is the "payload": an element of $\Lambda(\{\xi_i\}_{i \in is} \sqcup \{\chi_j\}_{j \in os})$.

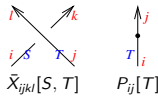


```

A[X_{i,j,k,l}[S_, T_]] := A[{L, i}, {j, k}, <{xi -> S, x_j -> T, x_k -> S, xi -> T}>,
  Expand[T^{-1/2} WExp[Expand[{xi, xi} . (1 1 - T; 0 T) . {x_j, x_k}] /. {xi -> x_b -> xi_a ^ x_b}]]];
A[X_{1,2,3,4}[u, v]]
A[{4, 1}, {2, 3}, <xi_1 -> u, x_2 -> v, x_3 -> u, xi_4 -> v>,
  Wedge[] - x_2 ^ xi_4 - sqrt(v) x_3 ^ xi_1 - x_3 ^ xi_4 + sqrt(v) x_3 ^ xi_4 + sqrt(v) x_2 ^ x_3 ^ xi_1 ^ xi_4];
A[X_{i,j,k,l}[tau_i, tau_j]];

```

The negative crossing and the "point":



```

A[X_{i,j,k,l}[S_, T_]] := A[{i, j}, {k, l}, <{xi -> S, xi_j -> T, x_k -> S, x_l -> T}>,
  Expand[T^{1/2} WExp[Expand[{xi, xi} . (T^{-1} 0; 1 - T^{-1} 1) . {x_k, x_l}] /. {xi_a -> xi_b}]]];
A[X_{i,j,k,l}[tau_i, tau_j]];
A[P_{i,j}[T_]] := A[{i}, {j}, <{xi -> T, x_j -> T}>, WExp[xi ^ x_j];
A[P_{i,j}[tau_i]];

```

The linear structure on \mathcal{A} 's:

```

A /: alpha_x A[is_, os_, cs_, w_] := A[is, os, cs, Expand[alpha w]]
A /: A[is1_, os1_, cs1_, w1_] + A[is2_, os2_, cs2_, w2_] /:
  (Sort@is1 == Sort@is2) ^ (Sort@os1 == Sort@os2) ^
  (Sort@Normal@cs1 == Sort@Normal@cs2) := A[is1, os1, cs1, w1 + w2]

```

Deciding if two \mathcal{A} 's are equal:

```

A /: A[is1_, os1_, _, w1_] == A[is2_, os2_, _, w2_] :=
  TrueQ[(Sort@is1 == Sort@is2) ^ (Sort@os1 == Sort@os2) ^
  PowerExpand[w1 == w2]]

```

The union operation on \mathcal{A} 's (implemented as "multiplication"):

```

A /: A[is1_, os1_, cs1_, w1_] * A[is2_, os2_, cs2_, w2_] :=
  A[is1 Union is2, os1 Union os2, Join[cs1, cs2], WP[w1, w2]]
Short[A[X_{2,4,3,1}[S, T]] * A[X_{3,4,6,5}]]

```



```

A[{1, 2, 3, 4}, {3, 4, 5, 6},
  <xi_2 -> S, x_4 -> T, x_3 -> S, xi_1 -> T, xi_3 -> tau_3, xi_4 -> tau_4, x_6 -> tau_3, x_5 -> tau_4>,
  sqrt(tau_4) Wedge[] -
  sqrt(tau_4) x_3 ^ xi_1 + sqrt(tau_4) x_3 ^ xi_1 - sqrt(tau_4) x_3 ^ xi_2 - sqrt(tau_4) x_4 ^ xi_1 - sqrt(tau_4) x_5 ^ xi_4 -
  x_6 ^ xi_3 + <<40>> + sqrt(tau_4) x_3 ^ x_5 ^ x_6 ^ xi_1 ^ xi_3 ^ xi_4 - sqrt(tau_4) x_3 ^ x_5 ^ x_6 ^ xi_2 ^ xi_3 ^ xi_4 -
  x_4 ^ x_5 ^ x_6 ^ xi_1 ^ xi_3 ^ xi_4 + sqrt(tau_4) x_3 ^ x_4 ^ x_5 ^ x_6 ^ xi_1 ^ xi_2 ^ xi_3 ^ xi_4];

```

Contractions of \mathcal{A} -objects:

```

c_{h,t}_@A[is_, os_, cs_, w_] := A[
  DeleteCases[is, t], DeleteCases[os, h], KeyDrop[cs, {x_h, xi_t}], c_{h,xi_t}[w]
] /. If[MatchQ[cs[{xi_t}, tau_], cs[{xi_t} -> cs[x_h], cs[x_h] -> cs[{xi_t}]]];
c_{4,4}[A[X_{2,4,3,1}[S, T]] * A[X_{3,4,6,5}]]
A[{1, 2, 3}, {3, 5, 6}, <xi_2 -> S, x_3 -> S, xi_1 -> T, xi_3 -> tau_3, x_6 -> tau_3, x_5 -> T>,
  Wedge[] - x_3 ^ xi_1 + T x_3 ^ xi_1 - T x_3 ^ xi_2 - x_5 ^ xi_1 - x_6 ^ xi_1 + x_6 ^ xi_1 - x_6 ^ xi_3 +
  T x_3 ^ x_5 ^ xi_1 ^ xi_2 - x_3 ^ x_6 ^ xi_1 ^ xi_2 + T x_3 ^ x_6 ^ xi_1 ^ xi_2 + x_3 ^ x_6 ^ xi_1 ^ xi_3 -
  x_3 ^ x_6 ^ xi_1 ^ xi_3 - x_3 ^ x_6 ^ xi_2 ^ xi_3 - x_5 ^ x_6 ^ xi_1 ^ xi_3 - x_3 ^ x_5 ^ x_6 ^ xi_1 ^ xi_2 ^ xi_3];

```

4. Skein relations and evaluations for \mathcal{A}

Automatic and intelligent multiple contractions:

```

c@A[is_, os_, cs_, w_] := Fold[c_{h,t}_@A[is_, os_, cs_, w_], A[is, os, cs, w], is os]
A[{A_}]:= c[A];
A[{A1_}, A2_] := Module[{A2},
  A2 = First@MaximalBy[{A2}, Length[A1[[1]] # #[[2]] + Length[A1[[2]] # #[[1]]] &];
  A[Join[{c[A1 A2]}, DeleteCases[{A2}, A2]]];
A[os_List] := A[os / os]
c[A[X_{2,4,3,1}[S, T]] * A[X_{3,4,6,5}]]
A[{1, 2}, {5, 6}, <xi_2 -> S, xi_1 -> T, x_6 -> S, x_5 -> T>,
  Wedge[] - x_5 ^ xi_1 - x_6 ^ xi_2 - x_5 ^ x_6 ^ xi_1 ^ xi_2];
A[{A[X_{2,4,3,1}[S, T]], A[X_{3,4,6,5}]}]
A[{1, 2}, {5, 6}, <xi_2 -> S, xi_1 -> T, x_6 -> S, x_5 -> T>,
  Wedge[] - x_5 ^ xi_1 - x_6 ^ xi_2 - x_5 ^ x_6 ^ xi_1 ^ xi_2];

```



```

A@{X_{4,1,6,3}[v, u], X_{3,2,5,4}}
A[{1, 2}, {5, 6}, <xi_2 -> v, x_5 -> u, xi_1 -> u, x_6 -> v>,
  sqrt(u) sqrt(v) Wedge[] - sqrt(u) x_5 ^ xi_1 + sqrt(u) x_5 ^ xi_2 - sqrt(u) sqrt(v) x_5 ^ xi_2 + sqrt(v) x_6 ^ xi_1 - sqrt(u) sqrt(v) x_6 ^ xi_1 -
  sqrt(v) x_6 ^ xi_2 - sqrt(u) x_5 ^ x_6 ^ xi_1 ^ xi_2 - sqrt(v) x_5 ^ x_6 ^ xi_1 ^ xi_2 + sqrt(u) sqrt(v) x_5 ^ x_6 ^ xi_1 ^ xi_2];

```