Implementation

August-26-08
9:18 PM

1. Implement a general "W" for wedge products.

   * wReduce
   * Wedge multiply.

   variable equivalences of the form Equiv[$v_1, v_2, \ldots$] Equiv[...]...

2. "Alexander Half Density"

   AHD[{list of in legs}, W[list of out legs], equivs, $\underbrace{\text{linear comb of W[ ]}}_{\text{the payload}}$]

   "Wedge"

Reducing an AHD:
* Sort legs of both kinds.
* Reduce the W's.
* Transfer a sign from the out legs to the payload
* Apply variable equivalences

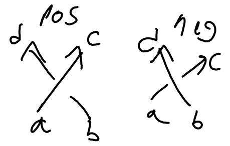
3. Composing AHD's:

   * Merge the variable equivalences.
   * Determine glued legs.
   * Wedge the payloads and the out legs.
   * Inner multiply by the glued legs
   * Recompute the in legs
   * Transfer signs.

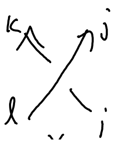
4. Define A on X_p, X_m, P, Chord, Arrow

Jana's conventions:

Pos[a_, b_, c_, d_] :> H[{a, b}, 1] + H[{a, c}, t[b] - 1] + H[{a, d}, -t[a]] + H[{b, c}, 1] + H[{c, d}, t[a]],
Neg[a_, b_, c_, d_] :> H[{a, b}, t[b]] + H[{a, d}, -t[b]] + H[{b, c}, 1] + H[{b, d}, -1 + t[a]] + H[{c, d}, 1],

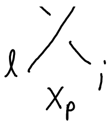
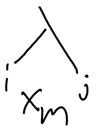


Dror's conventions:



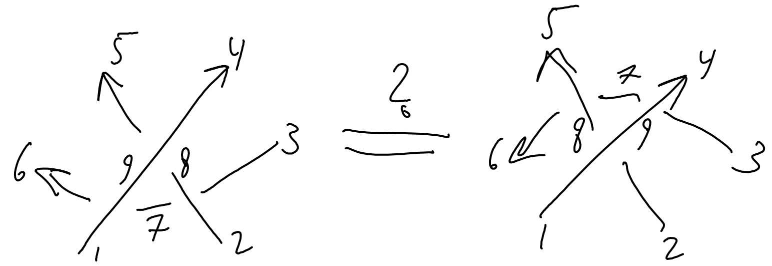
$\to$ $l \wedge i + (t_i - 1)\, l \wedge j - t_l\, l \wedge k + i \wedge j + t_l\, j \wedge k$

$t_j\, i \wedge j - t_j\, i \wedge l + j \wedge k + (t_i - 1)\, j \wedge l + k \wedge l$

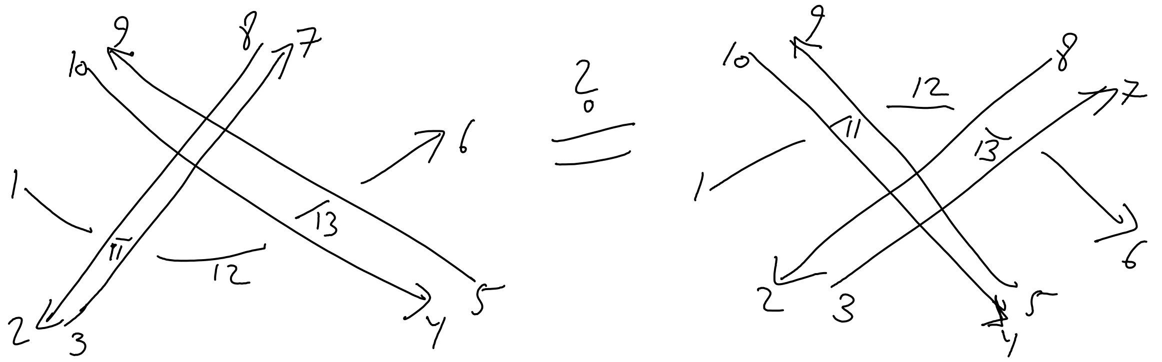
 $\to -t_l\, l\wedge k + i\wedge j + t_l\, j\wedge k$  $+ j\wedge k + (t_i - 1)\, j\wedge l + k\wedge l$

5. Compute the overall invariant.

R3:



Virtual Double Delta



Double Delta