

Pensieve header: A concise implementation of the FastKh algorithm.

```
<< KnotTheory`
Loading KnotTheory` version of February 5, 2013, 3:48:46.4762.
Read more at http://katlas.org/wiki/KnotTheory.

SetAttributes[{P, S}, Orderless];
dot /: dot[_]^{k_} := 0;
( $\sigma_S$ )[i_] :=  $\sigma$ [i] = First@Cases[ $\sigma$ , P[i, j_]  $\Rightarrow$  j];

ECP[ $\lambda\_List$ ] := Module[{ $\rho$ , ec}, (* "Equivalence Class Projection" *)
  ec = Fold[
    ( $\rho$  = First /@ Position[#1, #2];
     Append[Delete[#1, List /@  $\rho$ ], Union@@ (#1[[ $\rho$ ]])] &,
      $\lambda$ , Union @@  $\lambda$ 
    ] // SortBy[#, First] &;
  Union@@Replace[ec, c_  $\Rightarrow$  ((#  $\rightarrow$  First[c]) & /@ c), {1}];
ECP[ $\lambda\_S$ ] := ECP[Join[ $\lambda$ ] /. S | P  $\rightarrow$  List];
ECR[ $\lambda\_$ ] := Union[Last /@ ECP[ $\lambda$ ]] (* "Equiv. Class Representatives" *);

VCLaw[ $\beta_S, \mu_S, \tau_S$ ] := VCLaw[ $\beta, \mu, \tau$ ] = Module[
  {p, ins1, ins2, outs,  $\chi_S$ , h, law1, law2, dec},
  p = ECP[ $\beta, \mu, \tau$ ];
  ins1 = ECR[ $\beta, \mu$ ]; ins2 = ECR[ $\mu, \tau$ ]; outs = ECR[ $\beta, \tau$ ];
   $\chi_S$  =  $\frac{\text{Times} @@ (\text{h} /@ \text{Join}[\text{ins1}, \text{ins2}, \text{outs}] /. \text{p})}{\text{PowerExpand}[(\text{Times} @@ (\text{h} /@ (\text{Last} /@ \text{p})))]^{1/2}}$ ;
  dec =  $\chi_S /. \text{h}[i_]^{x_} \Rightarrow (2 \text{dot}[i])^{(2-x)/2}$ ;
  dec *= Times @@ MapThread[If[#1 == #2, 1, dot[#1] + dot[#2]] &,
    {outs, outs /. p}];
  law1 = dot /@ ins1; law1 = Thread[law1  $\rightarrow$  (law1 /. p)];
  law2 = dot /@ ins2; law2 = Thread[law2  $\rightarrow$  (law2 /. p)];
  {law1, law2, Expand[dec]}];

VC[Cob[ $\beta_S, \mu_S, \text{dots1}_$ ], Cob[ $\mu_S, \tau_S, \text{dots2}_$ ]] := Module[
  {law1, law2, dec},
  {law1, law2, dec} = VCLaw[ $\beta, \mu, \tau$ ];
  Expand[dec * (dots1 /. law1) (dots2 /. law2)];

m0[i_, j_][ $\sigma_S$ ] := Which[
   $\sigma$ [i]  $\neq$  j, Append[DeleteCases[ $\sigma$ , P[i, _] | P[_, j]], P[ $\sigma$ [i],  $\sigma$ [j]]],
   $\sigma$ [i] == j, DeleteCases[ $\sigma$ , P[i, j]];
m[i_, j_][ $\sigma_S$ ] := m0[i, j][ $\sigma$ ] * If[ $\sigma$ [i]  $\neq$  j, {1}, {q, q-1)];
m[i_, j_][q^{k_}  $\sigma_S$ ] := q^k m[i, j][ $\sigma$ ];
```

```

m[i_, j_][Cob[β_S, τ_S, dots_]] := Module[
  {nβ, nτ, p, ijdot, ndots, x},
  {nβ, nτ} = m0[i, j] /@ {β, τ};
  p = ECP[β, τ]; ijdot = dot[Min[i, j]];
  ndots = Which[
    β[i] ≠ j && τ[i] ≠ j, {{If[(i/.p) ≠ (j/.p), 1, dot[β[i]] + dot[τ[i]]]}},
    β[i] = j && τ[i] ≠ j, {{1, ijdot}},
    β[i] ≠ j && τ[i] = j, {{ijdot}, {1}},
    β[i] = j && τ[i] = j,  $\begin{pmatrix} ijdot & 0 \\ 1 & ijdot \end{pmatrix}$ ];
  ndots = Expand[dots * ndots] /. dot[k_] =>
    dot[k /. {i → β[i], j → β[j]} /. {i → τ[i], j → τ[j]} /. ECP[nβ, nτ]];
  If[β[i] = j && τ[i] = j, Coefficient[ndots /. ijdot → x, x], ndots];
Kom /: Kom[cs_, ds_] * Cob[qp1·β_, qp2·τ_, 1] := Module[{L, ρ, d, k},
  L = Length[cs]; ρk := ρk = Length[cs[[k]]]; ρ0 = ρL+1 = 0;
  Kom[
    MapThread[Join, List @@@ {
      Append[cs /. σ_S => qp1 Join[β, σ], {}],
      Prepend[cs /. σ_S => qp2 Join[τ, σ], {}] }],
    Table[
      If[(ρk + ρk-1) (ρk+1 + ρk) = 0, 0,
      d = Table[0, {ρk+1 + ρk}, {ρk + ρk-1}];
      If[k ≤ L && ρk ρk+1 ≠ 0, d[[1 ;; ρk+1, 1 ;; ρk]] = ds[[k]];
      If[k ≤ L && ρk ≠ 0, d[[ρk+1 + 1 ;; ρk+1 + ρk, 1 ;; ρk]] = (-1)k IdentityMatrix[ρk];
      If[k > 1 && ρk-1 ρk ≠ 0, d[[ρk+1 + 1 ;; ρk+1 + ρk, ρk + 1 ;; ρk + ρk-1]] = ds[[k - 1]];
      d
    ], {k, L} ] ] ]
m[i_, j_][Kom[cs_, ds_]] := Kom[
  Flatten /@ Map[m[i, j], cs, {2}],
  Table[
    If[Length[cs[[k]]] == 0 || Length[cs[[k+1]]] == 0, 0,
    (* else *) Table[
      m[i, j][Cob[cs[[k, b]] /. q → 1, cs[[k+1, a]] /. q → 1, ds[[k, a, b]]],
      {a, Length[cs[[k+1]]]}, {b, Length[cs[[k]]]}
    ] // ArrayFlatten ],
    {k, Length[ds]} ] ];

```

```

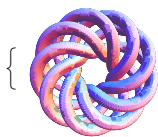
Contract[kom_Kom] := Module[{cs, ds, L, k, done, a, b,  $\phi$ ,  $\gamma\delta$ },
  {cs, ds} = List @@ kom; L = Length[ds];
  For[k = 1, k ≤ L, ++k,
    done = False; While[! done, done = True;
      For[a = 1, a ≤ Length[cs[[k+1]], ++a, For[b = 1, b ≤ Length[cs[[k]], ++b,
        If[NumberQ[ $\phi$  = ds[[k, a, b]]] &&  $\phi \neq 0$  && cs[[k+1, a]] = cs[[k, b]],
          done = False;
          If[Length[cs[[k]]] > 1 && Length[cs[[k+1]]] > 1,
             $\gamma\delta$  = Table[
              VC[Cob[cs[[k, d]], cs[[k+1, a]], ds[[k, a, d]]] /. q → 1,
              Cob[cs[[k, b]], cs[[k+1, c]], ds[[k, c, b]]] /. q → 1,
              {c, Length[cs[[k+1]]}], {d, Length[cs[[k]]] }];
            ds[[k]] = Expand[Drop[ds[[k]] -  $\phi^{-1} \gamma\delta$ , {a}, {b}],
              (* else *) ds[[k]] = 0];
            cs[[k]] = Drop[cs[[k]], {b}]; cs[[k+1]] = Drop[cs[[k+1]], {a}];
            If[k > 1, ds[[k-1]] = If[ds[[k-1]] === 0, 0, Drop[ds[[k-1]], {b}]]];
            If[k < L,
              ds[[k+1]] = If[ds[[k+1]] === 0, 0, Drop[ds[[k+1]], {}, {a}]]];
            If[a ≤ Length[cs[[k+1]], --a]; b = Length[cs[[k]]; ] ] ] ];
  Kom[cs, ds]];

CFKh[L_] := Module[
  {pd = PD[L], kom = Kom[{{S[]}}, {}], inside = {}, tp = 0, pos},
  While[Length[pd] > 0,
    pos = Last[Ordering[(Length[(List @@ #) ∩ inside]) & /@ pd]];
    kom = kom * (pd[[pos]] /. {
      X[i_, j_, k_, l_] /; (j - l == 1 || l - j > 1) =>
        Cob[q S[P[-i, j], P[k, -l]], q2 S[P[-i, -l], P[j, k]], 1],
      X[i_, j_, k_, l_] /; (l - j == 1 || j - l > 1) =>
        (-tp; Cob[q-2 S[P[-i, -j], P[k, l]], q-1 S[P[-i, l], P[-j, k]], 1])
    });
    (kom = Contract[kom // m[#, -#]) & /@ ((List @@ pd[[pos]]) ∩ inside);
    inside = inside ∪ (List @@ pd[[pos]); pd = Drop[pd, {pos}];
  Expand[ttp-1+Range[Length[First[kom]]].(List @@ Plus @@ First @ kom) /. S[] → 1] ]

```

```
K = TorusKnot[9, 5]; {TubePlot[K, ImageSize -> 80], CFKh[K]} // Timing
```

{1083.848148,



$$\left\{ \begin{aligned} & , q^{31} + q^{33} + q^{35} t^2 + q^{39} t^3 + q^{37} t^4 + q^{39} t^4 + q^{41} t^5 + q^{43} t^5 + q^{39} t^6 + q^{41} t^6 + q^{43} t^7 + \\ & q^{45} t^7 + q^{41} t^8 + 2 q^{43} t^8 + q^{45} t^9 + 2 q^{47} t^9 + 2 q^{45} t^{10} + 3 q^{49} t^{11} + 2 q^{47} t^{12} + 2 q^{49} t^{12} + q^{53} t^{12} + \\ & 3 q^{51} t^{13} + 2 q^{53} t^{13} + q^{49} t^{14} + 2 q^{51} t^{14} + q^{55} t^{14} + 2 q^{53} t^{15} + 3 q^{55} t^{15} + 2 q^{53} t^{16} + q^{57} t^{16} + \\ & q^{59} t^{16} + 3 q^{57} t^{17} + q^{55} t^{18} + q^{57} t^{18} + q^{61} t^{18} + 2 q^{59} t^{19} + q^{61} t^{19} + q^{59} t^{20} + q^{63} t^{20} + q^{63} t^{21} \end{aligned} \right\}$$