

Pensieve header: Cheap CF optimization for the NOE1 program (V6).

Initialization

```
SetDirectory["C:\\drorbn\\AcademicPensieve\\2016-12"];
Once[<< KnotTheory`];
Once[<< "../Projects/Profile/Profile.m"]
```

Loading KnotTheory` version of September 6, 2014, 13:37:37.2841.

Read more at <http://katlas.org/wiki/KnotTheory>.

This is Profile.m, Nov 2016 mods of July 1994 version

Rotational Virtual Knots

```
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] := Module[{n, xs, x, rots, front, k},
  n = Length[pd];
  xs = List@@pd /. x_X => If[PositiveQ[x], Xp[x[[4]], x[[1]], Xm[x[[2]], x[[1]]];
  rots = Table[0, {2 n});
  front = {0};
  For[k = 0, k < 2 n, ++k,
  If[k == 0 ∨ FreeQ[front, -k],
  front = Flatten[front /. k → Catch[xs /. {
    Xp[k + 1, L_] | Xm[L_, k + 1] => Throw[{L, k + 1, 1 - L]},
    Xp[L_, k + 1] | Xm[k + 1, L_] => ({++rots[[L]]; Throw[{1 - L, k + 1, L]})
  }]],
  If[MatchQ[front, {___, k, ___, -k, ___}], --rots[[k + 1]]
  ]
  ];
  RVK[xs, rots]
  ];
RVK[K_] := RVK[PD[K]];
```

NOE-It

1Gens

```
Ri_,j_+ := E[1, Log[ti cj, vi wj, vi ci wj + ci cj + vi2 wj2 / 4];
Ri_,j_- := E[1, -Log[ti cj, -ti-1 vi wj, -ci cj + ti-1 vi cj wj - ti-2 vi2 wj2 / 4];
(uri_ := E[ti-1/2, 0, 0, ci ti2]; nri_ := E[ti1/2, 0, 0, -ci ti2];)
```

1DP

```
DPx_→D_α,y_→D_β[P_][f_] := (* means P[∂α, ∂β][f] *)
PPDP@Total[CoefficientRules[P, {x, y}] /. ({m_, n_} → c_) => c D[f, {α, m}, {β, n}]]
```

1Util

```
CF[E[ω_, L_, Q_, P_]] :=
  PPCF@E[Expand@Together@ω, Expand@Together@L, Expand@Together@Q, Expand@PPTogether4P@Together@P];
```

1Util

$$\mathbb{E} /: \mathbb{E}[\omega_1, L_1, Q_1, P_1] \mathbb{E}[\omega_2, L_2, Q_2, P_2] := \text{CF} @ \mathbb{E}[\omega_1 \omega_2, L_1 + L_2, \omega_2 Q_1 + \omega_1 Q_2, \omega_2^4 P_1 + \omega_1^4 P_2];$$

Logos

$$\Delta[k_] := \left((t_k - 1) (2(\alpha\beta + \delta\mu)^2 - \alpha^2\beta^2) - 4v_k c_k w_k \delta^2 \mu^2 - \delta(1 + \mu)(w_k^2 \alpha^2 + v_k^2 \beta^2) - v_k^2 w_k^2 \delta^3 (1 + 3\mu) - 2(\alpha\beta + 2\delta\mu + v_k w_k \delta^2 (1 + 2\mu) + 2c_k \delta \mu^2)(w_k \alpha + v_k \beta) - 4(c_k \mu^2 + v_k w_k \delta(1 + \mu))(\alpha\beta + \delta\mu)(1 + t_k) \right) / 4;$$

1NOuw

$$\begin{aligned} N_{w_i, v_j \rightarrow k}[\mathbb{E}[\omega, L, Q, P]] &:= \text{PP}_{Nwv} @ \text{With}[\{q = ((1 - t_k) \alpha\beta + \beta v_k + \delta v_k w_k + \alpha w_k) / \mu\}, \\ &\mathbb{E}[\mu \omega, L, \mu \omega q + \mu(Q / w_i | v_j \rightarrow \theta), \mu^4 (\text{DP}_{w_i \rightarrow D_\alpha, v_j \rightarrow D_\beta}[P][e^q] / e \rightarrow 1) + \omega^4 \Delta[k]] // \text{CF} // \\ &\text{ReplaceAll}[\mu \rightarrow 1 + (t_k - 1) \delta] // \text{CF} // \\ &\text{ReplaceAll}[\{\alpha \rightarrow \omega^{-1} (\partial_{w_i} Q / v_j \rightarrow \theta), \beta \rightarrow \omega^{-1} (\partial_{v_j} Q / w_i \rightarrow \theta), \delta \rightarrow \omega^{-1} \partial_{w_i, v_j} Q\}]]]; \end{aligned}$$

1NOc

$$\begin{aligned} N_{c_j} (x: v | w)_i \rightarrow k [\mathbb{E}[\omega, L, Q, P]] &:= \text{PP}_{Ncx} @ \text{With}[\{q = e^\gamma \beta x_k + \gamma c_k\}, \text{CF} [\\ &\mathbb{E}[\omega, \gamma c_k + (L / c_j \rightarrow \theta), \omega e^\gamma \beta x_k + (Q / x_i \rightarrow \theta), e^{-q} \text{DP}_{c_j \rightarrow D_\gamma, x_i \rightarrow D_\beta}[P][e^q]] / \{\gamma \rightarrow \partial_{c_j} L, \beta \rightarrow \omega^{-1} \partial_{x_i} Q\}]]; \end{aligned}$$

1m

$$\begin{aligned} m_{i, j \rightarrow k} [Z_{\mathbb{E}}] &:= \text{PP}_m @ \text{Module}[\{x, z\}, \\ &\text{CF}[(Z // N_{w_i, v_j \rightarrow x} // N_{c_i, v_x \rightarrow x} // N_{w_x, c_j \rightarrow x}) / z_{-i | j | x} \rightarrow z_k]] \end{aligned}$$

Z

```
ul_ = nl_ = rot[_ , 0] = E[1, 0, 0, 0];
rot[i_ , 1] := ur_i;
rot[i_ , n_Integer] /; n > 1 := Module[{y}, rot[i, n - 1] rot[y, 1] // m_{i, y \to i};
rot[i_ , -1] := nr_i;
rot[i_ , n_Integer] /; n < -1 := Module[{y}, rot[i, n + 1] rot[y, -1] // m_{i, y \to i};
```

```

t_ = t;
Z[K_] := Z[RVK@K];
Z[rvk_RVK] := PPz@Module[{todo, n, rots, ζ, done, st, x, ζ1, i, j, k, k1, k2, k3},
  {todo, rots} = List@@rvk;
  AppendTo[rots, 0];
  n = Length[todo];
  ζ = E[1, 0, 0, 0];
  done = {0};
  st = Range[0, 2 n + 1];
  While[todo != {},
    {x} = MaximalBy[todo, Length[done ∩ {#[[1]], #[[2]], #[[1]] - 1, #[[2]] - 1}] &, 1];
    Z$todo = todo; Z$x = x;
    {i, j} = List@@x;
    ζ1 = Switch[Head[x],
      Xp, mj,k→j[R+i,j (R-k3,k nrk1 ulk2 // mk,k1→k // mk,k2→k // mk,k3→k)],
      Xm, mj,k→j[R-i,j (R+k,k3 nrk1 ulk2 // mk,k1→k // mk,k2→k // mk,k3→k)];
    ];
    ζ1 = rot[k, rots[[i]] ζ1 // mk,i→i; rots[[i]] = 0;
    ζ1 = ζ1 rot[k, rots[[i + 1]] // mi,k→i; rots[[i + 1]] = 0;
    ζ1 = rot[k, rots[[j]] ζ1 // mk,j→j; rots[[j]] = 0;
    ζ1 = ζ1 rot[k, rots[[j + 1]] // mj,k→j; rots[[j + 1]] = 0;
    ζ *= ζ1;
    If[MemberQ[done, i], ζ = ζ // mi,i+1→i; st = st /. st[[i + 2]] → st[[i + 1]];
    If[MemberQ[done, i - 1], ζ = ζ // mst[[i],i→st[[i]]; st = st /. st[[i + 1]] → st[[i]];
    If[MemberQ[done, j], ζ = ζ // mj,j+1→j; st = st /. st[[j + 2]] → st[[j + 1]];
    If[MemberQ[done, j - 1], ζ = ζ // mst[[j],j→st[[j]]; st = st /. st[[j + 1]] → st[[j]];
    done = done ∪ {i - 1, i, j - 1, j};
    todo = DeleteCases[todo, x];
  ];
  ζ /. {V0 → V, C0 → C, W0 → W}
]

```

Timing[Z[Knot[3, 1]]]

KnotTheory: Loading precomputed data in PD4Knots`.

$$\left\{ 2.8125, \mathbb{E} \left[-1 + \frac{1}{t} + t, 0, 0, -16 - \frac{2}{t^4} + \frac{2c}{t^4} + \frac{7}{t^3} - \frac{6c}{t^3} - \frac{14}{t^2} + \frac{10c}{t^2} + \frac{18}{t} - \frac{8c}{t} + 10t + 8ct - 4t^2 - 10ct^2 + t^3 + 6ct^3 - 2ct^4 + 2vw - \frac{2vw}{t^4} + \frac{4vw}{t^3} - \frac{6vw}{t^2} + \frac{2vw}{t} - 6tvw + 4t^2vw - 2t^3vw \right] \right\}$$

Testing I₁₀₀...

```

BeginProfile[];
Timing[Z[Knot[10, 100]]]
EndProfile[];

PrintProfile[];

```

Testing T_{9,5}...

```

BeginProfile[];
Timing[Z[TorusKnot[9, 5]]]
EndProfile[];

PrintProfile[];

```