Kashaev's Signature Conjecture

CMS Winter 2021 Meeting, December 4, 2021

Dror Bar-Natan with Sina Abbasi

Agenda. Show and tell with signatures.

Abstract. I will display side by side two nearly identical computer programs whose inputs are knots and whose outputs seem to always be the same. I'll then admit, very reluctantly, that I don't know how to prove that these outputs are always the same. One program I wrote mostly in Bedlewo, Poland, in the summer of 2003 and as of recently I understand why it computes the Levine-Tristram signature of a knot. The other is based on the 2018 preprint *On Symmetric Matrices Associated with Oriented Link Diagrams* by Rinat Kashaev (arXiv:1801.04632), where he conjectures that a certain simple algorithm also computes that same signature.

If you can, please turn your video on! (And mic, whenever needed).

These slides and all the code within are available at http://drorbn.net/cms21.

(I'll post the video there too)


```
Bed[K_, ω_] :=
  Module[{t, r, XingsByArmpits, bends, faces, p, A, is},
   t = 1 - ω; r = t + t*;
   XingsByArmpits =
    List @@ PD[K] /. x : X[i_, j_, k_, l_] :>
      If[PositiveQ[x], X+[-i, j, k, -l], X-[-j, k, l, -i]];
   bends = Times @@ XingsByArmpits /.
     _[X][a_, b_, c_, d_] :> p[a,-d] p[b,-a] p[c,-b] p[d,-c];
   faces = bends //. p[x__,y_] p[y_,z__] :> p[x,y,z];
   A = Table[0, Length@faces, Length@faces];
   Do[is = Position[faces, #][[1, 1]] & /@ List @@ x;
    A[[is, is]] += If[Head[x] === X+,
      ( -r    -t   2t    t*  )   (  r    -t   -2t*  t*  )
      ( -t*    0   t*    0   ) , ( -t*    0   t*    0   ) ],
      ( 2t*    t   -r   -t*  )   ( -2t    t    r   -t*  )
      (  t     0   -t    0   )   (  t     0   -t    0   )
    {x, XingsByArmpits}];
   MatrixSignature[A]];
```

```
Kas[K_, ω_] :=
  Module[{u, v, XingsByArmpits, bends, faces, p, A, is},
   u = Re[ω^(1/2)]; v = Re[ω];
   XingsByArmpits =
    List @@ PD[K] /. x : X[i_, j_, k_, l_] :>
      If[PositiveQ[x], X+[-i, j, k, -l], X-[-j, k, l, -i]];
   bends = Times @@ XingsByArmpits /.
     _[X][a_, b_, c_, d_] :> p[a,-d] p[b,-a] p[c,-b] p[d,-c];
   faces = bends //. p[x__,y_] p[y_,z__] :> p[x,y,z];
   A = Table[0, Length@faces, Length@faces];
   Do[is = Position[faces, #][[1, 1]] & /@ List @@ x;
    A[[is, is]] += If[Head[x] === X+,
      ( v  u  1  u )     ( v  u  1  u )
      ( u  1  u  1 ) , - ( u  1  u  1 ) ],
      ( 1  u  v  u )     ( 1  u  v  u )
      ( u  1  u  1 )     ( u  1  u  1 )
    {x, XingsByArmpits}];
   (MatrixSignature[A] - Writhe[K]) / 2];
```


Why am I showing you 💀code💀?

- I love code — it's fun!
- Believe it or not, it is more expressive than math-talk (though I'll do the math-talk as well, to confirm with prevailing norms).
- It is directly verifiable. Once it is up and running, you'll never ask yourself "did he misplace a sign somewhere"?

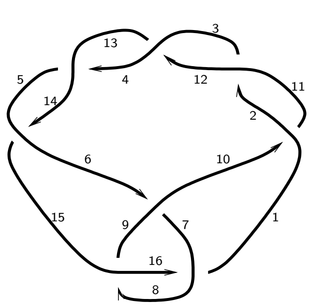

```
Bed[K_, ω_] :=
  Module[{t, r, XingsByArmpits, bends, faces, p, A, is},
   t = 1 - ω; r = t + t*;
   XingsByArmpits =
    List @@ PD[K] /. x : X[i_, j_, k_, l_] :>
      If[PositiveQ[x], X+[-i, j, k, -l], X-[-j, k, l, -i]];
   bends = Times @@ XingsByArmpits /.
     _[X][a_, b_, c_, d_] :> p[a,-d] p[b,-a] p[c,-b] p[d,-c];
   faces = bends //. p[x__,y_] p[y_,z__] :> p[x,y,z];
   A = Table[0, Length@faces, Length@faces];
   Do[is = Position[faces, #][[1, 1]] & /@ List @@ x;
    A[[is, is]] += If[Head[x] === X+,
      ( -r    -t   2t    t*  )   (  r    -t   -2t*  t*  )
      ( -t*    0   t*    0   ) , ( -t*    0   t*    0   ) ],
      ( 2t*    t   -r   -t*  )   ( -2t    t    r   -t*  )
      (  t     0   -t    0   )   (  t     0   -t    0   )
    {x, XingsByArmpits}];
   MatrixSignature[A]];
```

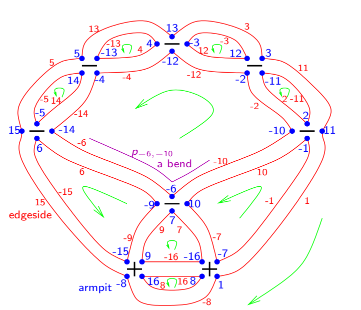
```
Kas[K_, ω_] :=
  Module[{u, v, XingsByArmpits, bends, faces, p, A, is},
   u = Re[ω^(1/2)]; v = Re[ω];
   XingsByArmpits =
    List @@ PD[K] /. x : X[i_, j_, k_, l_] :>
      If[PositiveQ[x], X+[-i, j, k, -l], X-[-j, k, l, -i]];
   bends = Times @@ XingsByArmpits /.
     _[X][a_, b_, c_, d_] :> p[a,-d] p[b,-a] p[c,-b] p[d,-c];
   faces = bends //. p[x__,y_] p[y_,z__] :> p[x,y,z];
   A = Table[0, Length@faces, Length@faces];
   Do[is = Position[faces, #][[1, 1]] & /@ List @@ x;
    A[[is, is]] += If[Head[x] === X+,
      ( v  u  1  u )     ( v  u  1  u )
      ( u  1  u  1 ) , - ( u  1  u  1 ) ],
      ( 1  u  v  u )     ( 1  u  v  u )
      ( u  1  u  1 )     ( u  1  u  1 )
    {x, XingsByArmpits}];
   (MatrixSignature[A] - Writhe[K]) / 2];
```


Verification.

```
Once[<< KnotTheory`]
Loading KnotTheory` version of February 2, 2020, 10:53:45.2097.
Read more at http://katlas.org/wiki/KnotTheory.
MatrixSignature[A_] :=
  Total[Sign[Select[Eigenvalues[A], Abs[#] > 10^-12 &]]];
Writhe[K_] := Sum[If[PositiveQ[x], 1, -1], {x, List @@ PD@K}];
Sum[ω = e^(i RandomReal[{0,2π}]); Bed[K, ω] == Kas[K, ω], {10},
  {K, AllKnots[{3, 10}]}]
KnotTheory: Loading precomputed data in PD4Knots`.
2490 True
```


Label everything!

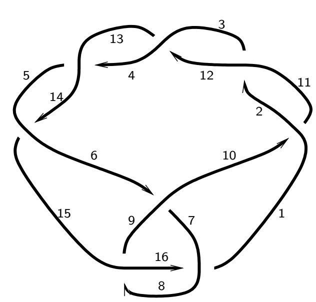




$PD[X[10,1,11,2], X[2,11,3,12], \ldots]$ $\{X_-[-1,11,2,-10], X_-[-11,3,12,-2], \ldots\}$

Lets run our code line by line. . .

```
PD[8_2] = PD[X[10, 1, 11, 2],
    X[2, 11, 3, 12], X[12, 3, 13, 4],
    X[4, 13, 5, 14], X[14, 5, 15, 6],
    X[8, 16, 9, 15], X[16, 8, 1, 7],
    X[6, 9, 7, 10]];
K = 8_2;
```



Video and more at http://www.math.toronto.edu/~drorbn/Talks/CMS-2112/