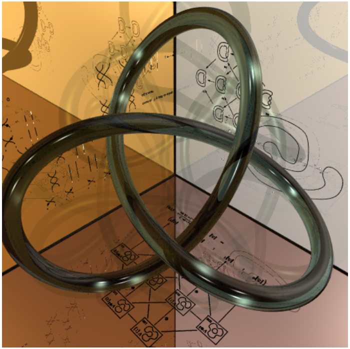
Heegaard Floer homology

Dylan Thurston

Joint with/work of Sarkar, Lipshitz, Manolescu, Ozsváth, Szabó

http://www.math.columbia.edu/~dpt/speaking



February 20, 2012, Tokyo Institute of Technology

Outline

▶ **Introduction**

Computing *HFK*

Variants of Heegaard Floer homology

Making Heegaard Floer homology more computable

Method 1: Nice diagrams

Method 2: Surgery

Method 3: Cut the 3-manifold

Further questions

4-dimensional invariants

There are multiple smooth structures on the same topological 4-manifold.

Several theories give 4-manifold invariants to detect this:[1]

Donaldson theory
  $\leftrightarrow$ (conj. Seiberg–Witten '94)
Monopole Floer homology (Seiberg-Witten)
  $\cong$ (Taubes '08)
Embedded contact homology
  $\cong$ (Cutluhan-Lee-Taubes, Colin-Ghiggini-Honda '10)
Heegaard Floer homology

We topologists only have one trick in 4 dimensions!

Monopoles: More computable version of Donaldson invariants
HF homology: More computable version of monopoles

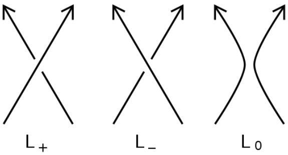
---

[1] Some theories/equivalences currently only work in 3 dimensions.

3-dimensional invariants

By contrast, there are many 3-manifold invariants.
Focus on a knot K in $\mathbb{R}^3$.

One of oldest: Alexander polynomial $\Delta(K)$.
Algebraic topology: Look at $H_1(\mathbb{R}^3 \setminus K)$ under deck transforms.
Skein theory: $\Delta(L_+) - \Delta(L_-) = (t^{1/2} - t^{-1/2})\Delta(L_0)$



More recent: Jones polynomial, HOMFLYPT polynomial, . . .

How are 3- and 4-dimensional theories related?

Knot homologies

Many knot invariants are one- or two-variable Laurent polynomials.

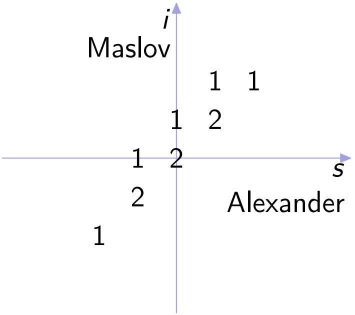
Can often find a doubly- or triply-graded homology theory whose Euler characteristic is the polynomial invariant.

Knot polynomial	Knot homology
Alexander (1928)	Heegaard Floer (2002) Instanton Floer (2008)
Jones (1983)	Khovanov (1999)
HOMFLYPT (1985)	Khovanov-Rozansky (2004)
Kauffman (1990)	Khovanov-Rozansky (2007) (conj)

Passage polynomial $\Rightarrow$ homology called *categorification*.

Heegaard Floer homology

$\dim(\widehat{HFK}_i(K;s))$:
$(K = 10_{132})$



Characteristics of $\widehat{HFK}$:

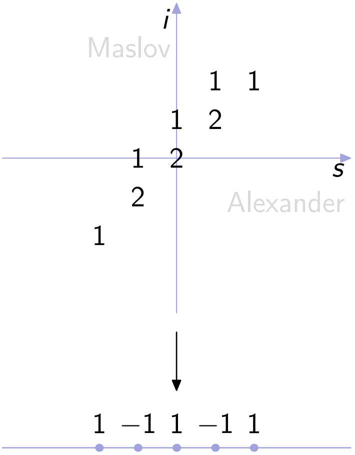
- Bigraded;
- Euler characteristic is Conway-Alexander polynomial Δ;
- Max grading is knot genus (so detects unknot); (Ozsváth-Szabó 2001)
- Determines knot fibration; (Ghiggini, Ni 2006)
- Defined via pseudo-holomorphic curves.

Today we will see a simple algorithm for computing HFK…

…and one of the world's simplest algorithms for detecting unknot!

Heegaard Floer homology

$\dim(\widehat{HFK}_i(K; s))$:
$(K = 10_{132})$



Characteristics of $\widehat{HFK}$:

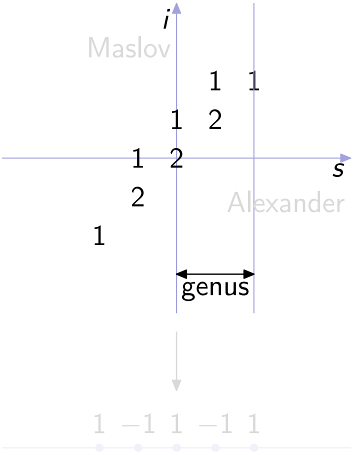
- Bigraded;
- Euler characteristic is Conway-Alexander polynomial Δ;
- Max grading is knot genus (so detects unknot); (Ozsváth-Szabó 2001)
- Determines knot fibration; (Ghiggini, Ni 2006)
- Defined via pseudo-holomorphic curves.

Today we will see a simple algorithm for computing *HFK*...

...and one of the world's simplest algorithms for detecting unknot!

Heegaard Floer homology

$\dim(\widehat{HFK}_i(K;s))$:
($K = 10_{132}$)



Characteristics of $\widehat{HFK}$:

- Bigraded;
- Euler characteristic is Conway-Alexander polynomial Δ;
- Max grading is knot genus (so detects unknot); (Ozsváth-Szabó 2001)
- Determines knot fibration; (Ghiggini, Ni 2006)
- Defined via pseudo-holomorphic curves.

Today we will see a simple algorithm for computing HFK...

...and one of the world's simplest algorithms for detecting unknot!

Knot genus

Definition

A *Seifert surface* for a knot is an oriented surface embedded in space whose boundary is the knot.
The *genus* of a knot is the minimal genus of any Seifert surface.

Seifert surfaces always exist.
The genus of a knot is 0 iff it is the unknot.

Theorem (Neuwirth 1960)

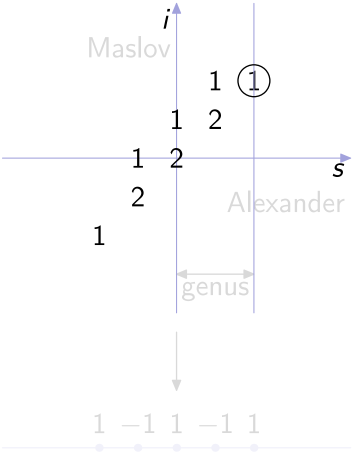
Genus of $K \geq$ degree of Alexander-Conway polynomial

Theorem (Ozsváth-Szabó 2001)

Genus of $K = \max s$ so that $\widehat{HFK}_*(K; s) \neq 0$

Heegaard Floer homology

$\dim(\widehat{HFK}_i(K; s))$:
$(K = 10_{132})$



Characteristics of $\widehat{HFK}$:

- Bigraded;
- Euler characteristic is Conway-Alexander polynomial Δ;
- Max grading is knot genus (so detects unknot); (Ozsváth-Szabó 2001)
- Determines knot fibration; (Ghiggini, Ni 2006)
- Defined via pseudo-holomorphic curves.

Today we will see a simple algorithm for computing HFK...

...and one of the world's simplest algorithms for detecting unknot!

Fibered knots

Definition

A knot is *fibered* if complement is a fiber bundle of a surface over the circle.
Or: Seifert surface can be swept around to cover complement.

Theorem (Neuwirth 1960)

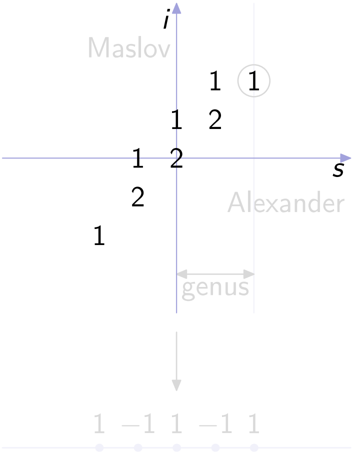
K is fibered $\Rightarrow$ Alexander-Conway polynomial is monic

Theorem (Ghiggini-Ni 2006)

K is fibered $\Leftrightarrow \widehat{HFK}_i(K; s)$ is monic w.r.t. s
$\Leftrightarrow$ for max s so that $\widehat{HFK}_*(K; s) \neq 0$, $\dim(\widehat{HFK}_*(K; s)) = 1$

Heegaard Floer homology

$\dim(\widehat{HFK}_i(K; s))$:
$(K = 10_{132})$



Characteristics of $\widehat{HFK}$:

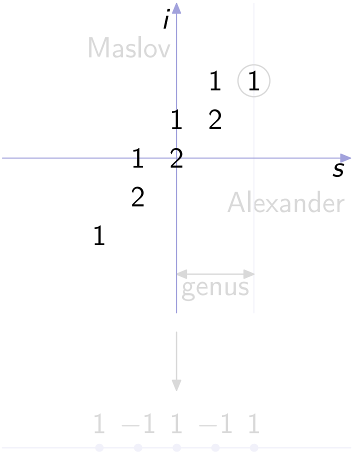
- Bigraded;
- Euler characteristic is Conway-Alexander polynomial Δ;
- Max grading is knot genus (so detects unknot); (Ozsváth-Szabó 2001)
- Determines knot fibration; (Ghiggini, Ni 2006)
- Defined via pseudo-holomorphic curves.

Today we will see a simple algorithm for computing HFK...

...and one of the world's simplest algorithms for detecting unknot!

Heegaard Floer homology

$\dim(\widehat{HFK}_i(K;s))$:
$(K = 10_{132})$



Characteristics of $\widehat{HFK}$:

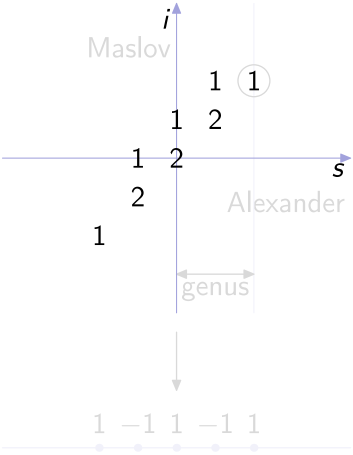
- Bigraded;
- Euler characteristic is Conway-Alexander polynomial Δ;
- Max grading is knot genus (so detects unknot); (Ozsváth-Szabó 2001)
- Determines knot fibration; (Ghiggini, Ni 2006)
- Defined via pseudo-holomorphic curves.

Today we will see a simple algorithm for computing HFK...

...and one of the world's simplest algorithms for detecting unknot!

Heegaard Floer homology

$\dim(\widehat{HFK}_i(K;s))$:
$(K = 10_{132})$

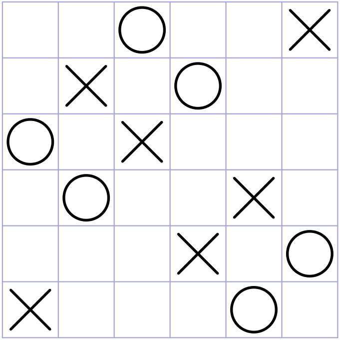


Characteristics of $\widehat{HFK}$:

- Bigraded;
- Euler characteristic is Conway-Alexander polynomial Δ;
- Max grading is knot genus (so detects unknot); (Ozsváth-Szabó 2001)
- Determines knot fibration; (Ghiggini, Ni 2006)
- Defined via pseudo-holomorphic curves.

Today we will see a simple algorithm for computing HFK...
...and one of the world's simplest algorithms for detecting unknot!

Setting: Grid diagrams



Grid diagram: square diagram with one X and one O per row and column.

Turn it into a knot: connect

X to O in each column;

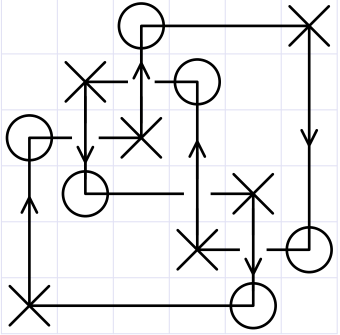
O to X in each row.

Cross vertical strands over horizontal.

Grid diagrams exist: take any diagram, rotate crossings so vertical crosses over horizontal.

The knot is unchanged under *cyclic rotations*:
Move top segment to bottom.

Setting: Grid diagrams



Grid diagram: square diagram with one X and one O per row and column.

Turn it into a knot: connect

X to O in each column;

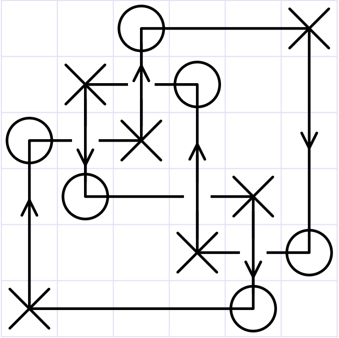
O to X in each row.

Cross vertical strands over horizontal.

Grid diagrams exist: take any diagram, rotate crossings so vertical crosses over horizontal.

The knot is unchanged under *cyclic rotations*:
Move top segment to bottom.

Setting: Grid diagrams



Grid diagram: square diagram with one X and one O per row and column.

Turn it into a knot: connect

X to O in each column;

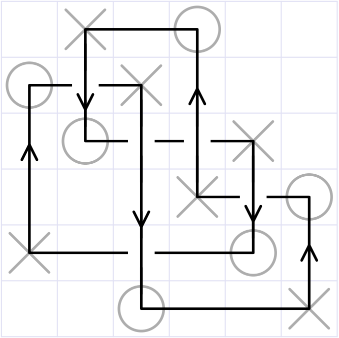
O to X in each row.

Cross vertical strands over horizontal.

Grid diagrams exist: take any diagram, rotate crossings so vertical crosses over horizontal.

The knot is unchanged under *cyclic rotations*:
Move top segment to bottom.

Setting: Grid diagrams



Grid diagram: square diagram with one X and one O per row and column.

Turn it into a knot: connect

X to O in each column;

O to X in each row.

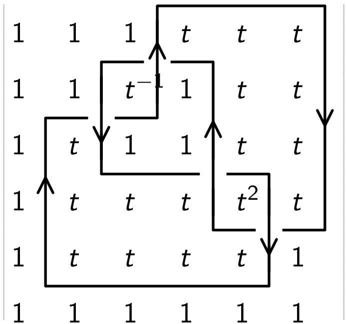
Cross vertical strands over horizontal.

Grid diagrams exist: take any diagram, rotate crossings so vertical crosses over horizontal.

The knot is unchanged under *cyclic rotations*:
Move top segment to bottom.

Computing the Alexander polynomial

We categorify the following formula:

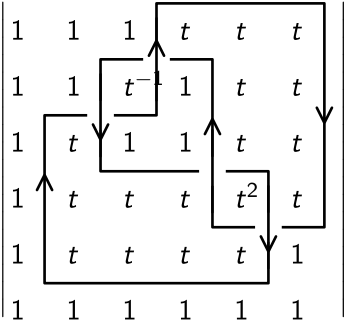


$$\begin{vmatrix} 1 & 1 & 1 & t & t & t \\ 1 & 1 & t^{-1} & 1 & t & t \\ 1 & t & 1 & 1 & t & t \\ 1 & t & t & t & t^2 & t \\ 1 & t & t & t & t & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{vmatrix} = \pm t^*(1-t)^{n-1}\Delta(K;t)$$

- Make matrix of $t^{-\text{winding } \#}$ (with extra row/column of 1's);
- det determines the Conway-Alexander polynomial Δ ($n =$ size of diagram; here 6)

Computing the Alexander polynomial

We categorify the following formula:



$$\begin{vmatrix} 1 & 1 & 1 & t & t & t \\ 1 & 1 & t^{-1} & 1 & t & t \\ 1 & t & 1 & 1 & t & t \\ 1 & t & t & t & t^2 & t \\ 1 & t & t & t & t & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{vmatrix} = \pm t^*(1-t)^{n-1}\Delta(K;t)$$

- Make matrix of $t^{-\text{winding } \#}$ (with extra row/column of 1's);
- det determines the Conway-Alexander polynomial Δ ($n =$ size of diagram; here 6)

Outline

Introduction

▶ **Computing** *HFK*

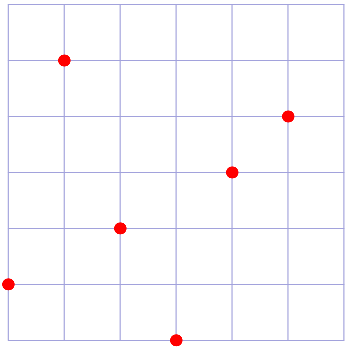
Variants of Heegaard Floer homology

Making Heegaard Floer homology more computable

- Method 1: Nice diagrams
- Method 2: Surgery
- Method 3: Cut the 3-manifold

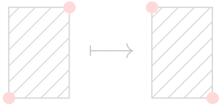
Further questions

Computing *HFK*: Chain complex $\widetilde{CK}$



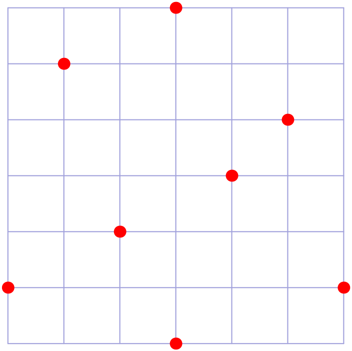
Define a chain complex $\widetilde{CK}$ over $\mathbb{F}_2$.

- $n!$ generators: matchings between horizontal and vertical gridcircles (as counted in det for Alexander).
- Boundary ∂ switches corners on *empty rectangles*:



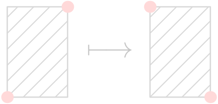
Sum over all ways to switch SW-NE corners of an empty rectangle to NW-SE corners. (*Empty* means: no X's, O's, or other points in generator.)

Computing HFK: Chain complex $\widetilde{CK}$



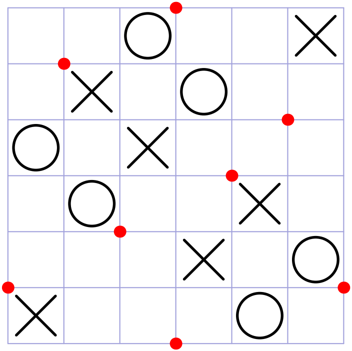
Define a chain complex $\widetilde{CK}$ over $\mathbb{F}_2$.

- $n!$ generators: matchings between horizontal and vertical gridcircles (as counted in det for Alexander).
- Boundary ∂ switches corners on *empty rectangles*:



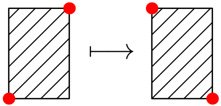
Sum over all ways to switch SW-NE corners of an empty rectangle to NW-SE corners. (*Empty* means: no X's, O's, or other points in generator.)

Computing HFK: Chain complex $\widetilde{CK}$



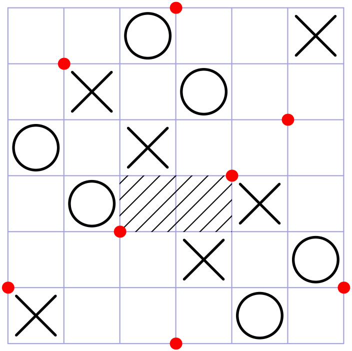
Define a chain complex $\widetilde{CK}$ over $\mathbb{F}_2$.

- $n!$ generators: matchings between horizontal and vertical gridcircles (as counted in det for Alexander).
- Boundary ∂ switches corners on *empty rectangles*:



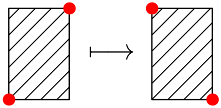
Sum over all ways to switch SW-NE corners of an empty rectangle to NW-SE corners. (*Empty* means: no X's, O's, or other points in generator.)

Computing HFK: Chain complex $\widetilde{CK}$



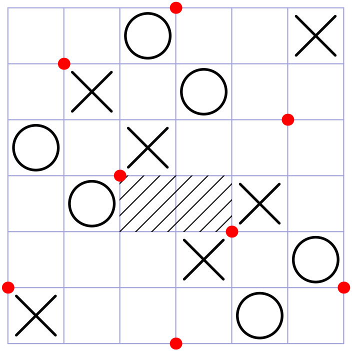
Define a chain complex $\widetilde{CK}$ over $\mathbb{F}_2$.

- $n!$ generators: matchings between horizontal and vertical gridcircles (as counted in det for Alexander).
- Boundary ∂ switches corners on *empty rectangles*:



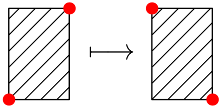
Sum over all ways to switch SW-NE corners of an empty rectangle to NW-SE corners. (*Empty* means: no X's, O's, or other points in generator.)

Computing HFK: Chain complex $\widetilde{CK}$



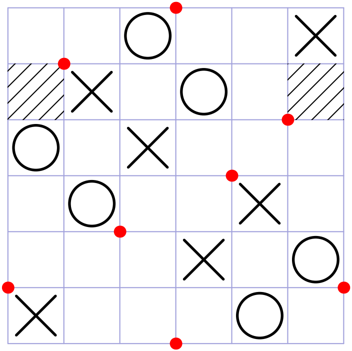
Define a chain complex $\widetilde{CK}$ over $\mathbb{F}_2$.

- $n!$ generators: matchings between horizontal and vertical gridcircles (as counted in det for Alexander).
- Boundary ∂ switches corners on *empty rectangles*:



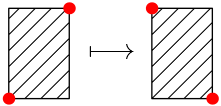
Sum over all ways to switch SW-NE corners of an empty rectangle to NW-SE corners. (*Empty* means: no X's, O's, or other points in generator.)

Computing HFK: Chain complex $\widetilde{CK}$



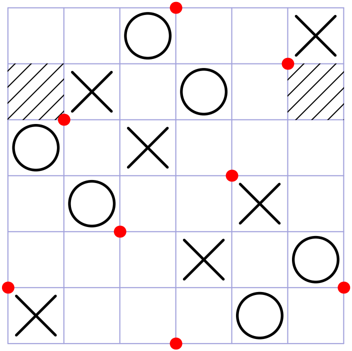
Define a chain complex $\widetilde{CK}$ over $\mathbb{F}_2$.

- $n!$ generators: matchings between horizontal and vertical gridcircles (as counted in det for Alexander).
- Boundary ∂ switches corners on *empty rectangles*:



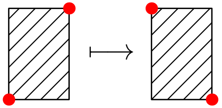
Sum over all ways to switch SW-NE corners of an empty rectangle to NW-SE corners. (*Empty* means: no X's, O's, or other points in generator.)

Computing *HFK*: Chain complex $\widetilde{CK}$



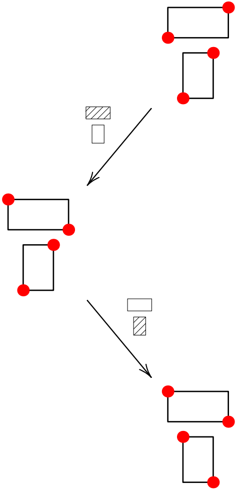
Define a chain complex $\widetilde{CK}$ over $\mathbb{F}_2$.

- $n!$ generators: matchings between horizontal and vertical gridcircles (as counted in det for Alexander).
- Boundary ∂ switches corners on *empty rectangles*:



Sum over all ways to switch SW-NE corners of an empty rectangle to NW-SE corners. (*Empty* means: no X's, O's, or other points in generator.)

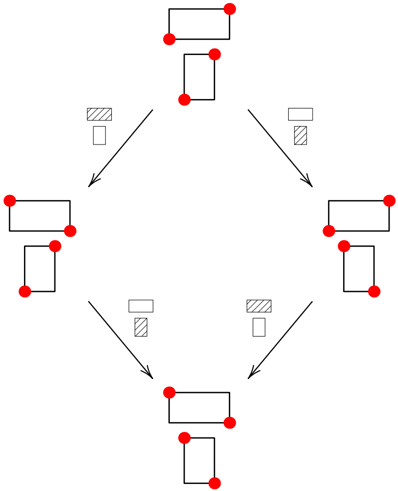
Computing HFK: $\partial^2 = 0$



Each term in ∂^2 must have a mate:

- If rectangles are disjoint, take rectangles in either order.
- If rectangles share a corner, decompose the union in another way.

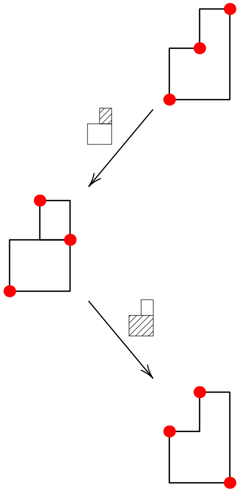
Computing HFK: $\partial^2 = 0$



Each term in ∂^2 must have a mate:

- If rectangles are disjoint, take rectangles in either order.
- If rectangles share a corner, decompose the union in another way.

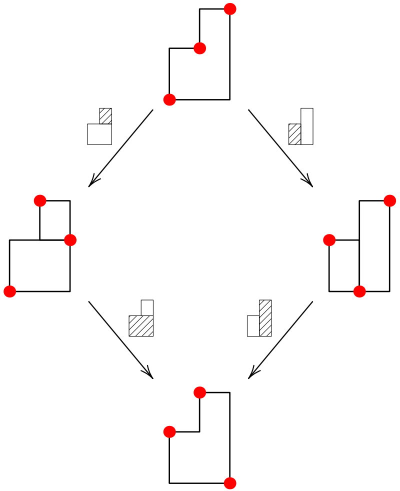
Computing *HFK*: $\partial^2 = 0$



Each term in ∂^2 must have a mate:

- If rectangles are disjoint, take rectangles in either order.
- If rectangles share a corner, decompose the union in another way.

Computing *HFK*: $\partial^2 = 0$

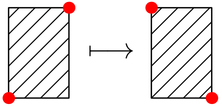


Each term in ∂^2 must have a mate:

- If rectangles are disjoint, take rectangles in either order.
- If rectangles share a corner, decompose the union in another way.

Computing *HFK*: Gradings on $\widetilde{CK}$

In the plane,



removes one *inversion*.

For $A, B, C \subset \mathbb{R}^2$,

$$\mathcal{I}(A, B) := \#\{\, {}_a\square^b \mid a \in A, b \in B \,\}$$
$$\mathcal{I}(A - B, C) := \mathcal{I}(A, C) - \mathcal{I}(B, C)$$

For $\mathbf{x}$ a generator, $\mathbb{X}$ = set of X's, $\mathbb{O}$ = set of of O's, gradings are:

- **Maslov:** $M(\mathbf{x}) := \mathcal{I}(\mathbf{x} - \mathbb{O}, \mathbf{x} - \mathbb{O}) + 1.$
- **Alexander:** Sum of winding numbers around generator pts, or $A(\mathbf{x}) := \frac{1}{2}\big(\mathcal{I}(\mathbf{x} - \mathbb{O}, \mathbf{x} - \mathbb{O}) - \mathcal{I}(\mathbf{x} - \mathbb{X}, \mathbf{x} - \mathbb{X}) - (n-1)\big).$

Computing HFK: The answer

Theorem (Manolescu-Ozsváth-Sarkar '06)

For G a grid diagram for K,

$$H_*(\widetilde{CK}(G)) \simeq \widehat{HFK}(K) \otimes V^{\otimes n-1}$$

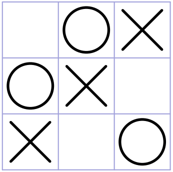
where $V := (\mathbb{F}_2)_{0,0} \oplus (\mathbb{F}_2)_{-1,-1}$.

(Remember the factor of $(1-t)^{n-1}$ in determinant formula for Δ.)

Gillam and Baldwin used this to compute $\widehat{HFK}$ for all knots with ≤ 11 crossings, including new values of knot genus.

Exercises

- Find a grid diagram for the trefoil.
- Compute $\widehat{HFK}$ of the unknot from the diagram below.

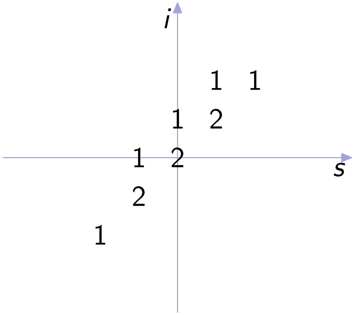


- Show that the figure 8 knot is fibered by computing $\widehat{HFK}$ of the figure 8 knot in the highest Alexander grading.
- Show that the Alexander polynomial is invariant under some moves that preserve the knot.

Outline

The Heegaard Floer homology package

$\dim \widehat{HFK}_i(K;s)$:



HF comes in many variants. So far, we have seen the simplest, $\widehat{HF}$. Also HF^+, HF^-, and HF^∞.

Thinking about variants will allow removing factors of $V^{\otimes n-1}$.

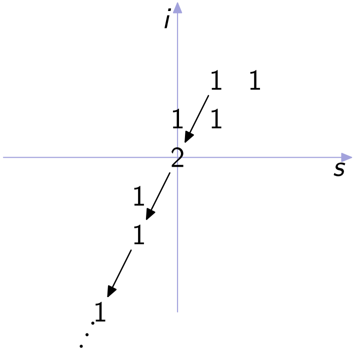
Also allows computing 4-manifold invariants.

Relation:

- CFK^- is a complex over $\mathbb{F}_2[U]$
- U has degree $(-1, -2)$
- Related to $\widehat{HFK}$ by Universal Coefficient Theorem (set U to 0 on chains).

The Heegaard Floer homology package

dim $HFK^-_i(K;s)$:



HF comes in many variants. So far, we have seen the simplest, $\widehat{HF}$. Also HF^+, HF^-, and HF^∞.

Thinking about variants will allow removing factors of $V^{\otimes n-1}$.

Also allows computing 4-manifold invariants.

Relation:

- CFK^- is a complex over $\mathbb{F}_2[U]$
- U has degree $(-1,-2)$
- Related to $\widehat{HFK}$ by Universal Coefficient Theorem (set U to 0 on chains).

Computing HFK^-

To compute HFK^- for grid diagrams:

- Add one U_i for each O.
- Complex $CK^-(G)$ generated by same generators over $\mathbb{F}_2[U_1, \dots, U_n]$
- ∂ counts rects. that contain only O's, weighted by corresponding U_i.

Theorem (Manolescu-Ozsváth-Sarkar)

$H_*(CK^-(G)) \simeq HFK^-(K)$.
Each U_i acts by U on the homology.

Other variants are similar, but treat O's in different ways.
E.g., CFK^∞ is similar, but over $\mathbb{F}_2[U, U^{-1}]$ instead of $\mathbb{F}_2[U]$.

Further variants

Can also:

- Allow rectangles to cross X's as well to get a filtered complex with trivial total homology.
- Add signs (in essentially unique way) to work over $\mathbb{Z}[U]$.

Cross O's? (count with U_i variables) \ Cross X's?	No	Yes
No	$\widehat{HFK} \otimes V^{n-1}$	Filtered version
All but 1	$\widehat{HFK}$	"
Yes	HFK^-	"

Exercises

- Check that ∂^2 is still 0 in the version where rectangles can cross O's.
- Compute HFK^- of the unknot from the same diagram as before.

Outline

Rest of the theory?

Grid diagrams can compute Heegaard Floer homology for knots.

Problem

Inefficient: basis of chain complex grows like $n!$

But what about 3- and 4-manifolds? How to compute for those?

Heegaard diagrams

The grid diagram technique can be generalized.

Definition

A *Heegaard diagram* $\mathcal{H}$ is a surface Σ, with two sets of k curves $\{\alpha_i\}$ and $\{\beta_i\}$, with each set of curves not intersecting itself and homologically independent. Heegaard diagrams represent 3-manifolds.

Heegaard Floer homology is defined by counting pseudo-holomorphic curves related to $\mathcal{H}$. Either:

- Count holomorphic disks in $\mathrm{Sym}^k(\Sigma)$ or
- Count holomorphic curves in $\Sigma \times [0,1] \times \mathbb{R}$.

Boundary conditions given by the $\{\alpha_i\}$ and $\{\beta_i\}$.

Method 1: Nice diagrams

Definition

A *nice Heegaard diagram* is one in which all regions, except for one, have 2 or 4 sides.

Theorem (Sarkar-Wang '06)

Nice Heegaard diagrams exist for any 3-manifold.

In nice diagrams, holomorphic curve counts are easy: Count empty bigons, rectangles.

Problem

Nice diagrams are huge! Cannot handle 4-manifolds.

Method 2: Surgery approach

Every 3-manifold is *surgery* on a link: cut out a neighborhood of the link and reglue another way.

Manolescu-Ozsváth, Manolescu-Ozsváth-T: the effect of surgery can be computed using grid diagram techniques. Also compute 4-manifold invariants this way.

Problem

Complicated, inefficient.

Method 3: Bordered Floer homology

Another approach: Cut 3-manifold into simpler pieces along surfaces. (Lipshitz-Ozsváth-T, ongoing)

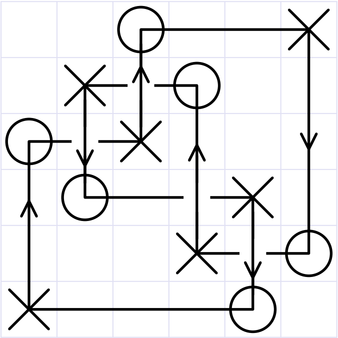
Want: Extend HF as a TQFT down a dimension

Geometry	Algebra
Closed 4-manifold W^4	Invariant $HF(W, \mathfrak{s})$
3-manifold Y^3	Homology $HF(Y, \mathfrak{s})$
4-manifold w/ $\partial W^4 = Y$	$HF(W) \in HF(Y)$
Surface F	Algebra A(F)
3-manifold w/ $\partial Y = F$	Module $CF(Y)$ over $\mathcal{A}(F)$
$Y = Y_1 \cup_F Y_2$	$CF(Y) \simeq CF(Y_1) \otimes_{\mathcal{A}(F)} CF(Y_2)$

Benefits:

- Computability (theoretical)
- Computability (practical)
- Axioms

Toy model: Planar diagrams



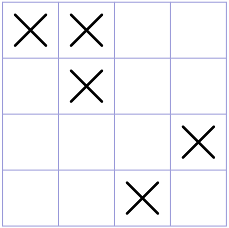
A *grid diagram* represents a knot.

A *planar diagram* P is a square grid with blocks. Do not identify sides.

Chain complex $CF(P)$:

- Generators given by permutations
- Differential counts empty rectangles (*not* wrapping)
- Not an invariant of anything

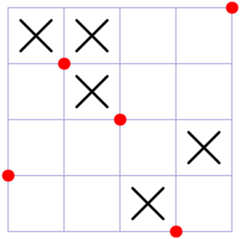
Toy model: Planar diagrams



A *grid diagram* represents a knot.
A *planar diagram* P is a square grid with blocks. Do not identify sides.
Chain complex $CF(P)$:

- Generators given by permutations
- Differential counts empty rectangles (*not* wrapping)
- Not an invariant of anything

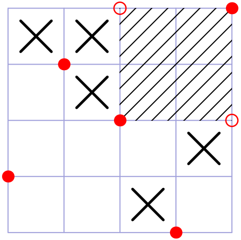
Toy model: Planar diagrams



A *grid diagram* represents a knot.
A *planar diagram* P is a square grid with blocks. Do not identify sides.
Chain complex $CF(P)$:

- Generators given by permutations
- Differential counts empty rectangles (*not* wrapping)
- Not an invariant of anything

Toy model: Planar diagrams



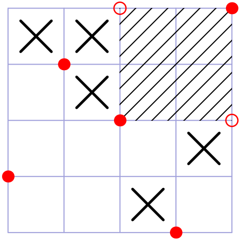
A *grid diagram* represents a knot.

A *planar diagram* P is a square grid with blocks. Do not identify sides.

Chain complex $CF(P)$:

- Generators given by permutations
- Differential counts empty rectangles (*not* wrapping)
- Not an invariant of anything

Toy model: Planar diagrams



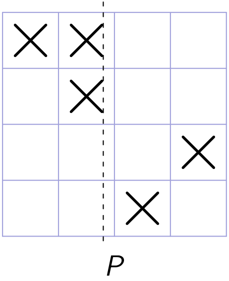
A *grid diagram* represents a knot.

A *planar diagram* P is a square grid with blocks. Do not identify sides.

Chain complex $CF(P)$:

- Generators given by permutations
- Differential counts empty rectangles (*not* wrapping)
- Not an invariant of anything

Splitting planar diagrams



Want to split planar diagrams in two: $P = P_1 \cup P_2$.

Associate modules $CPA(P_1)$, $CPD(P_2)$

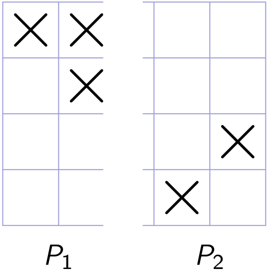
$$CP(P) = CPA(P_1) \otimes CPD(P_2)$$

$CPA(P_1)$ is a right differential module. Interactions on boundary encoded in algebra action.

$CPD(P_2)$ is a left, projective module. Interactions on boundary encoded in differential.

Tensor product is nice since $CPD(P_2)$ is projective.

Splitting planar diagrams



Want to split planar diagrams in two: $P = P_1 \cup P_2$.

Associate modules $CPA(P_1)$, $CPD(P_2)$

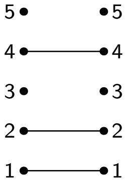
$$CP(P) = CPA(P_1) \otimes CPD(P_2)$$

$CPA(P_1)$ is a right differential module. Interactions on boundary encoded in algebra action.

$CPD(P_2)$ is a left, projective module. Interactions on boundary encoded in differential.

Tensor product is nice since $CPD(P_2)$ is projective.

Naive algebra $\widetilde{\mathcal{A}}(n, k)$

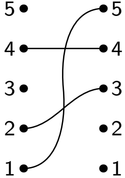


Idempotent
$\{1, 2, 4\} \subset \{1, \ldots, 5\}$

Defining a (naive) version of strands algebra $\widetilde{\mathcal{A}}(n, k)$ (really a category).

- Idempotents (objects): k-element subsets $S \subset \{1, \ldots, n\}$
- Elements (morphisms): $\mathrm{Mor}(S, T)$ spanned by $\phi : S \xrightarrow{\sim} T$, $\phi(i) \geq i$
- Product: composition
- Differential: sum over smoothings of crossings

Naive algebra $\widetilde{\mathcal{A}}(n,k)$



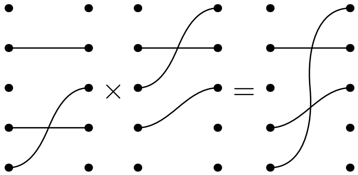
Element

$1 \mapsto 5,\ 2 \mapsto 3,\ 4 \mapsto 4$

Defining a (naive) version of strands algebra $\widetilde{\mathcal{A}}(n,k)$ (really a category).

- Idempotents (objects): k-element subsets $S \subset \{1,\ldots,n\}$
- Elements (morphisms): $\mathrm{Mor}(S,T)$ spanned by $\phi : S \xrightarrow{\sim} T$, $\phi(i) \geq i$
- Product: composition
- Differential: sum over smoothings of crossings

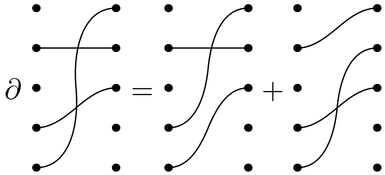
Naive algebra $\widetilde{\mathcal{A}}(n, k)$



Defining a (naive) version of strands algebra $\widetilde{\mathcal{A}}(n, k)$ (really a category).

- Idempotents (objects): k-element subsets $S \subset \{1, \ldots, n\}$
- Elements (morphisms): $\mathrm{Mor}(S, T)$ spanned by $\phi : S \xrightarrow{\sim} T$, $\phi(i) \geq i$
- Product: composition
- Differential: sum over smoothings of crossings

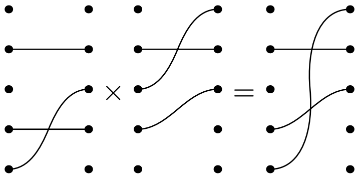
Naive algebra $\widetilde{\mathcal{A}}(n, k)$



Defining a (naive) version of strands algebra $\widetilde{\mathcal{A}}(n, k)$ (really a category).

- Idempotents (objects): k-element subsets $S \subset \{1, \ldots, n\}$
- Elements (morphisms): $\mathrm{Mor}(S, T)$ spanned by $\phi : S \xrightarrow{\sim} T$, $\phi(i) \geq i$
- Product: composition
- Differential: sum over smoothings of crossings

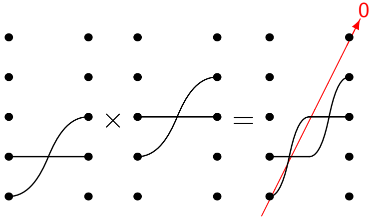
Strands algebra $\mathcal{A}(n, k)$



$\widetilde{\mathcal{A}}(n, k)$ has a filtration by number of crossings.
$\mathcal{A}(n, k)$ is the associated graded algebra.

- Product is 0 if it introduces a double-crossing.
- Differential: likewise.

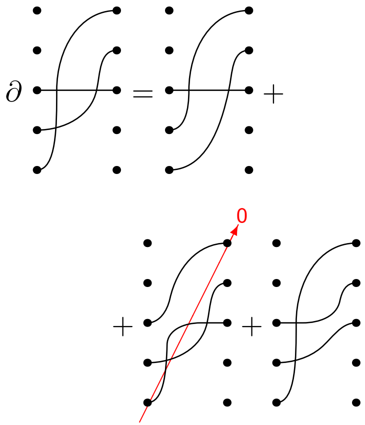
Strands algebra $\mathcal{A}(n, k)$



$\widetilde{\mathcal{A}}(n, k)$ has a filtration by number of crossings.
$\mathcal{A}(n, k)$ is the associated graded algebra.

- Product is 0 if it introduces a double-crossing.
- Differential: likewise.

Strands algebra $\mathcal{A}(n, k)$



$\widetilde{\mathcal{A}}(n, k)$ has a filtration by number of crossings.
$\mathcal{A}(n, k)$ is the associated graded algebra.

- Product is 0 if it introduces a double-crossing.
- Differential: likewise.

Outline

Introduction

Computing *HFK*

Variants of Heegaard Floer homology

Making Heegaard Floer homology more computable

- Method 1: Nice diagrams
- Method 2: Surgery
- Method 3: Cut the 3-manifold

▶ **Further questions**

Further questions

There are many questions left!

- Combinatorial proofs of genus equality and other nice properties?
- Construct a topological space with this homology?
- Manifold version of other link homologies?
- Make entire theory computable?
- Do computations!

Appendix: Crossing number vs. Grid number

Knots are usually ordered by *crossing number*:
Minimum number of crossings in a planar diagram.

For grid diagrams, natural to consider *grid number* (or *arc index*):
Minimum size of a grid diagram.

Theorem (Bae–Park, Morton–Beltrami)

Grid number of an alternating knot is equal to crossing number $+2$.
For non-alternating knots, grid number strictly less.