Computations in Tangle Floer Homology

Ayeong(Amy)Lee

Columbia University

August 1, 2018

Movitation: Why compute Tangle Floer Homology Groups?

- Powerful Link Invariant: Detects genus, fiberedness, Alexander polynomial, and an enhanced version called HFK- which contains a concordance invariant.
- To compute grid homology, for many knots (e.g., knots with large grid number n > 12) the number of generators (n!) is too large for a computer to handle.
  - Divide our knot into pieces called tangles and compute the tangle invariants(D, A, and DA modules).
  - Take "Box tensored product" of the tangle invariants and recover the grid homology of the knot.

Goals

1. Basic Structures
   - Free Module, Element, Generators, DGAlgebra, Tensor, Tangle, Strand Algebra
2. Tangle Module Structures
   - D, A, DA Structures
3. Chain Complex and Box Tensor Product
   - $\widetilde{CT}(\mathbb{T}_i) \boxtimes_{A^-(\partial^R \mathcal{T})} \widetilde{CT}(\mathbb{T}_j) \cong \widetilde{CT}(\mathbb{T}_i \circ \mathbb{T}_j)$
4. Tangle Floer Homology Groups

Basic Structure of the Floer Package

- Define Abstract Classes and Interfaces in Python for basic algebraic structures such as Free Module, DG Algebra, Tensor DGAlgebra.
- Given a knot, create a **Tangle object** and objects of related algebraic structures such as **Strand Algebra** and **Strand Diagram**.
- After computing a chain complex, apply cancellation lemma to reduce the structure.
- Code an algorithm that computes box tensor product between modules $\widetilde{CT}(\mathbb{T}_i)$ and get the chain complex for the entire tangle $\mathbb{T}$.

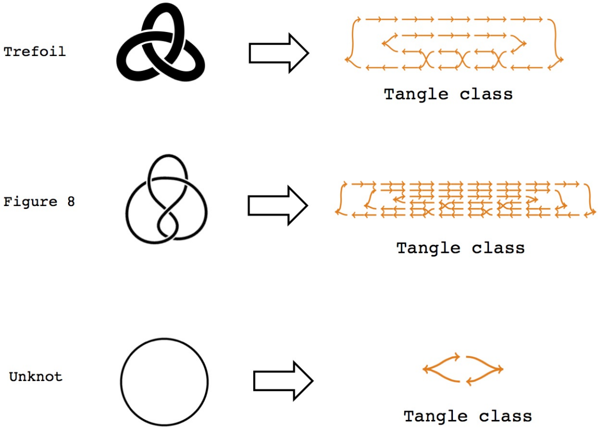
Main Data Structure Used

- 'Dictionary' Object is used for implementing elementary tangle pairs, free module elements, differentials, and chain complex arrows.
- Append is O(1), Get O(1), Get Length O(1), Delete O(n), Copy O(n)

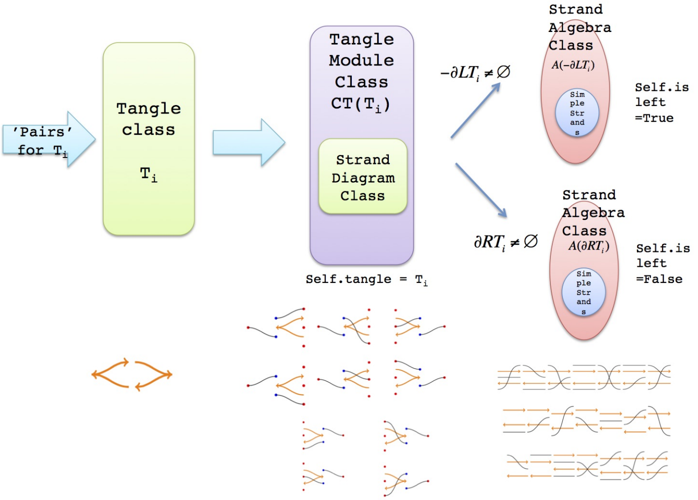
Dictionary Code

```
dic = {key1 : value1, key2 : value2, key3 : value 3...}
```

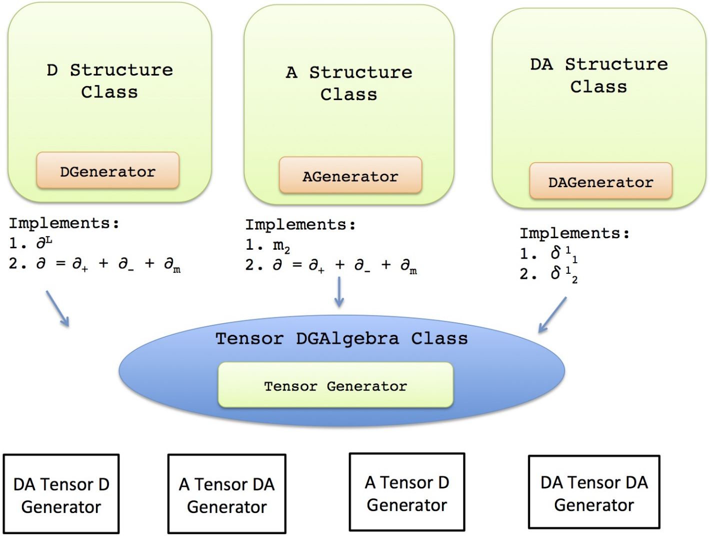
Basic Structure of the Floer Package



Basic Structure of the Floer Package

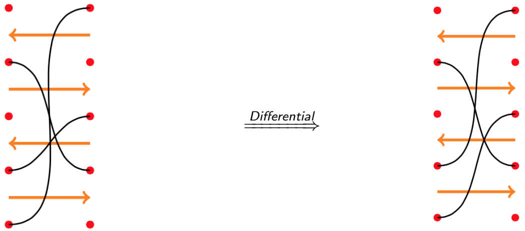


Basic Structure of the Floer Package



Algebra Differential

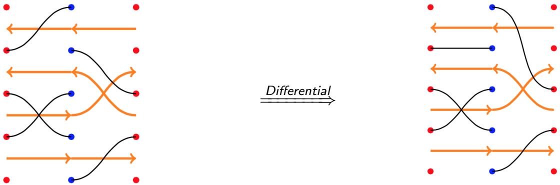
1. returns list = [((s1,s2), <class 'StrandDiagram'>))...]



Differential for A and D Structures

$\partial(\mathbf{x}) := (\partial_+ + \partial_- + \partial_m)(\mathbf{x})$

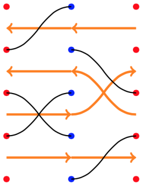
1. returns list = [((s1,s2), <class 'StrandDiagram'>))...]



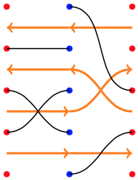
D Structure Class

D Structure

1. $\partial^L(\mathbf{x}) = \widetilde{CT}(\mathbb{T}i) \to A(-\partial^L\mathcal{T}) \otimes \widetilde{CT}(\mathbb{T}i)$
   - returns list = [((s1,s2), <class 'SimpleStrand'>,<class 'StrandDiagram'>))...]
2. $\partial(\mathbf{x}) = (\partial_+ + \partial_- + \partial_m)(\mathbf{x})$
   - returns list = [((s1,s2), <class 'StrandDiagram'>))...]



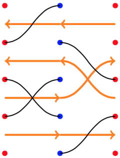
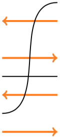
$\Rightarrow \partial^L(\mathbf{x}) =$



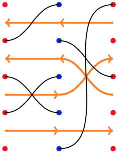
A Structure Class

A Structure

1. $m_2 : \widetilde{CT}(\mathbb{T}) \otimes A(-\partial^R \mathcal{T}) \to \widetilde{CT}(\mathbb{T})$
   - returns <class 'StrandDiagram'>
2. $\partial(\mathbf{x}) := (\partial_+ + \partial_- + \partial_m)(\mathbf{x})$

 $\otimes$ 

$\Rightarrow \mathrm{m}_2(\mathbf{x}, a) =$



DA Structure Class

DA Structure

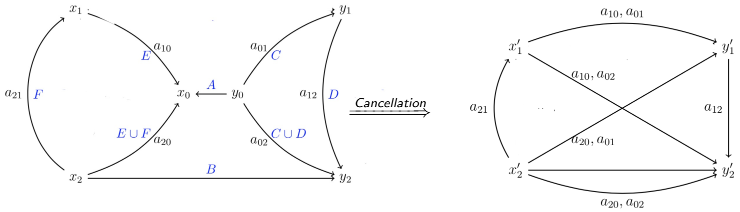
1. $\delta_1^1(\mathbf{x}) : \widetilde{CT}(\mathbb{T}) \to A(-\partial^L \mathcal{T}) \otimes \widetilde{CT}(\mathbb{T})$

   $:= e_L^D(x) \otimes d\mathbf{x} + \partial^L \mathbf{x}$
2. $\delta_2^1(\mathbf{x}) : \widetilde{CT}(\mathbb{T}) \otimes A(\partial^R \mathcal{T}) \to A(-\partial^L \mathcal{T}) \otimes \widetilde{CT}(\mathbb{T})$

   $:= e_L^D(x) \otimes m_2(\mathbf{x}, a)$
3. Both return <class 'TensorGenerator'>)

Code Result

Cancellation Lemma



Future Work

- Finish algorithm for Box Tensor Product between DA, D, and A modules.
- Modify the mod relations for differentials such that double points are allowed.
- Implement a computer program to compute the tangle Floer invariants working over the polynomial ring $\mathbb{F}_2[U]$, not just $\mathbb{F}_2$.
  - This recovers another version of grid homology (the minus version) that is more powerful as a knot invariant (than the tilde version).

The End