Computing TVBW Mapping Class Group Representations

Paul Gustafson
Texas A&M University

The Property F conjecture

Conjecture (Rowell)

Let $\mathcal{C}$ be a braided fusion category and let X be a simple object in $\mathcal{C}$. The braid group representations $\mathcal{B}_n$ on $\mathrm{End}(X^{\otimes n})$ have finite image for all $n > 0$ if and only if X is weakly integral (i.e. $\mathrm{FPdim}(X)^2 \in \mathbf{Z}$).

Modified Property F conjecture

Conjecture

The Turaev-Viro-Barrett-Westbury (TVBW) mapping class group representation associated to a compact surface Σ and spherical fusion category $\mathcal{A}$ has finite image iff $\mathcal{A}$ is weakly integral.

Test cases:

- $\mathcal{A} = \mathrm{Vect}_G^{\omega}$
- $\mathcal{A} = \mathcal{TY}(\mathbb{Z}_N, \cdot, \cdot)$

Related Work (Vect_G^ω case)

Theorem (Ng–Schauenberg)

Every modular representation associated to a modular category has finite image.

Theorem (Etingof–Rowell–Witherspoon)

The braid group representation associated to the modular category $\mathrm{Mod}(D^\omega(G))$ *has finite image.*

Theorem (Fjelstad–Fuchs)

Every mapping class group representation of a closed surface with at most one marked point associated to $\mathrm{Mod}(D(G))$ *has finite image.*

First result

Theorem (G.)

The image of any Vect_G^ω *TVBW representation* ρ *of a mapping class group of an orientable, compact surface* Σ *with boundary is finite.*

Idea of proof:
Show that the representation of each Birman generator lies in a quotient of a finite group of monomial matrices.

The TVBW space associated to a 2-manifold

- Using Kirillov's definitions, the representation space we consider is

$$H := \frac{\mathcal{A}\text{-colored graphs in } \Sigma}{\text{local relations}}$$

- The vector space H is canonically isomorphic to the usual (triangulation-based) TVBW state sum vector space associated to Σ.

Colored graphs in Σ

- Let $\Gamma \subset \Sigma$ be an undirected finite graph embedded in Σ.

Colored graphs in Σ

- Let $\Gamma \subset \Sigma$ be an undirected finite graph embedded in Σ.
- Define E^{or} to be the set of orientation edges of Γ, i.e. pairs $\mathbf{e} = (e, \text{orientation of } e)$; for such an oriented edge $\mathbf{e}$, we denote by $\bar{\mathbf{e}}$ the edge with opposite orientation.

Colored graphs in Σ

- Let $\Gamma \subset \Sigma$ be an undirected finite graph embedded in Σ.
- Define E^{or} to be the set of orientation edges of Γ, i.e. pairs $\mathbf{e} = (e, \text{orientation of } e)$; for such an oriented edge $\mathbf{e}$, we denote by $\bar{\mathbf{e}}$ the edge with opposite orientation.
- A *coloring* of Γ is the following data:
    - Choice of an object $V(\mathbf{e}) \in \operatorname{Obj} \mathcal{A}$ for every oriented edge $\mathbf{e} \in E^{or}$ so that $V(\bar{\mathbf{e}}) = V(\mathbf{e})^*$.

Colored graphs in Σ

- Let $\Gamma \subset \Sigma$ be an undirected finite graph embedded in Σ.
- Define E^{or} to be the set of orientation edges of Γ, i.e. pairs $\mathbf{e} = (e, \text{orientation of } e)$; for such an oriented edge $\mathbf{e}$, we denote by $\bar{\mathbf{e}}$ the edge with opposite orientation.
- A *coloring* of Γ is the following data:
  - Choice of an object $V(\mathbf{e}) \in \operatorname{Obj} \mathcal{A}$ for every oriented edge $\mathbf{e} \in E^{or}$ so that $V(\bar{\mathbf{e}}) = V(\mathbf{e})^*$.
  - Choice of a vector $\varphi(v) \in \operatorname{Hom}_{\mathcal{A}}(1, V_1 \otimes \cdots \otimes V_n)$ for every interior vertex v, where $\mathbf{e}_1, \ldots, \mathbf{e}_n$ are edges incident to v, taken in counterclockwise order and with outward orientation.

Local relations

- Isotopy of the graph embedding

Local relations

- Isotopy of the graph embedding
- Linearity in the vertex colorings

Local relations

- Isotopy of the graph embedding
- Linearity in the vertex colorings

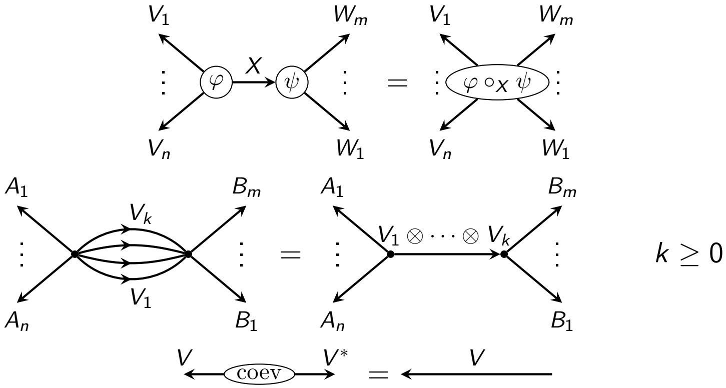

Figure: The remaining local relations.

Consequences of the local relations

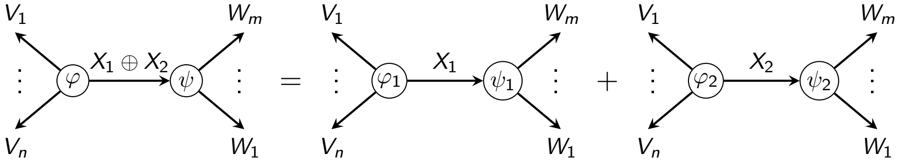

Figure: Additivity in edge colorings. Here φ_1, φ_2 are compositions of φ with projector $X_1 \oplus X_2 \to X_1$ (respectively, $X_1 \oplus X_2 \to X_2$), and similarly for ψ_1, ψ_2.

- Additivity in edge colorings

Consequences of the local relations

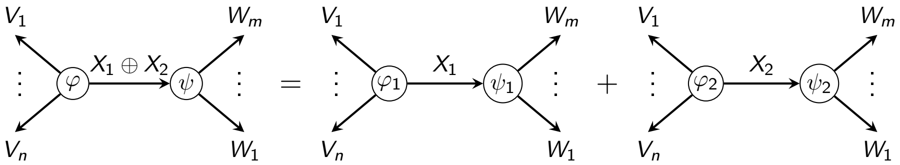

Figure: Additivity in edge colorings. Here φ_1, φ_2 are compositions of φ with projector $X_1 \oplus X_2 \to X_1$ (respectively, $X_1 \oplus X_2 \to X_2$), and similarly for ψ_1, ψ_2.

- Additivity in edge colorings
- A colored graph may be evaluated on any disk $D \subset S$, giving an equivalent colored graph Γ' such that Γ' is identical to Γ outside of D, has the same colored edges crossing ∂D, and contains at most one colored vertex within D.

First Dehn twist

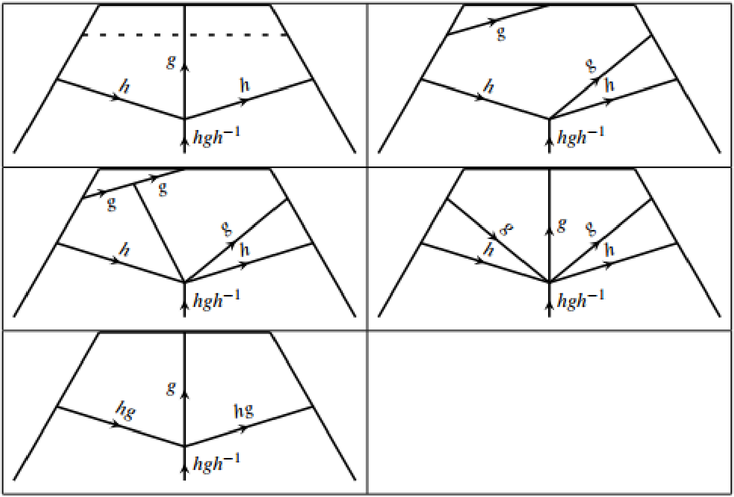

TABLE 1. First type of Dehn twist. Unlabeled interior edges are colored by the group identity element.

Second Dehn twist

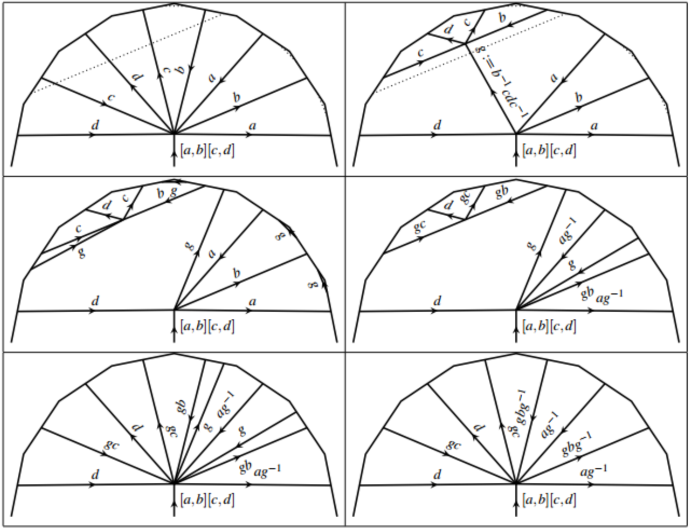

TABLE 2. Second type of Dehn twist.

Braid generator

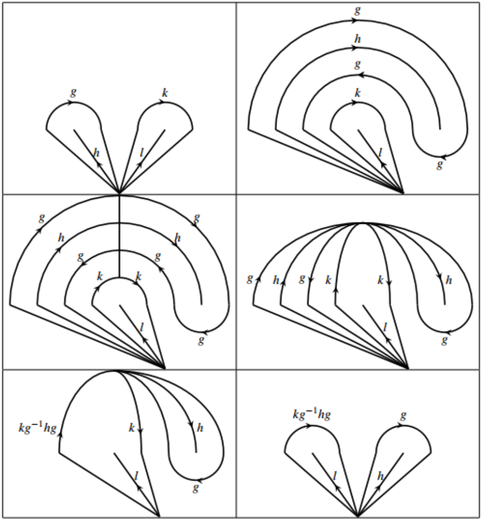

TABLE 3. A braid generator. Unlabeled interior edges are colored by the group identity element.

Dragging a point

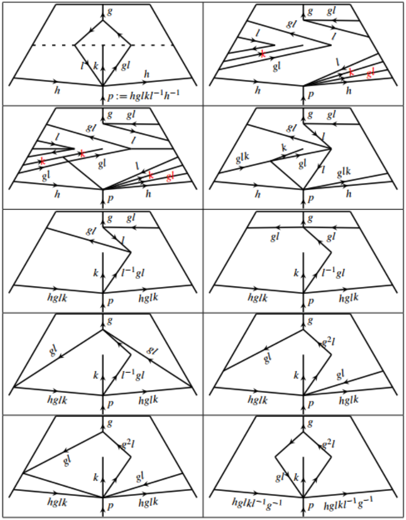

TABLE 4. Pulling a boundary component along a generator for the fundamental group of the corresponding closed surface. Unlabeled interior edges are colored by the group identity element.

Next step: Tambara-Yamagami categories

Let A be a finite abelian group, χ a bicharacter on A, and $\nu \in \{\pm 1\}$. The **Tamabara-Yamagami category** $\mathcal{TY}(A, \chi, \nu)$ is the skeletal spherical category with simple objects $\{a : a \in A\} \cup \{m\}$, fusion rules given by

$$a \otimes b = ab \text{ for } a, b \in A \qquad a \otimes m = m \qquad m \otimes m = \bigoplus_{a \in A} a,$$

and the following nontrivial structural morphisms

$$\alpha_{a,m,b} = \chi(a,b)\,\mathrm{id}_m \qquad \alpha_{m,a,m} = \bigoplus_{b \in A} \chi(a,b)\,\mathrm{id}_b$$

$$\alpha_{m,m,m} = (\nu|A|^{-1/2}\chi^{-1}(a,b)\,\mathrm{id}_m)_{a,b \in A},$$

$$j_m = \nu\,\mathrm{id}_m \qquad ev_m = \nu|A|^{1/2}\pi_1$$

Related Work ($\mathcal{TY}$ case)

Theorem (Rowell–Wenzl)

The images of the braid group representations on $\mathrm{End}_{SO(N)_2}(S^{\otimes n})$ *for* N *odd are isomorphic to images of braid groups in Gaussian representations; in particular, they are finite groups.*

Motivation

- Why are Tambara-Yamagami categories interesting?
  - Multi-fusion channels ($m \otimes m = \bigoplus_{a \in A} a$)
  - Gauging (with respect to group inversion action on $\mathbf{Z}_N$)
- Problem: Want to calculate actual matrices

Calculations = Hard

Easiest example (first Dehn twist, Vect_G^{ω}):

$$
\begin{aligned}
&\frac{\omega(h,g,h^{-1})\omega(h,gh^{-1},hg^{-1}h^{-1})\omega(g,h^{-1},hg^{-1}h^{-1})\omega(g,g^{-1}h^{-1},h)}{\omega(g^{-1},g^{-1},g)\omega(g^{-1},g^{-1}h^{-1},h)\omega(g^{-1},h^{-1},hg^{-1}h^{-1})\omega(g,g^{-2}h^{-1},hg)}\cdot \\
&\frac{\omega(g,g^{-1},h^{-1})\omega(g,g^{-1}h^{-1},h)\omega(g^{-1}h^{-1},h,g)\omega(g^{-2}h^{-1},h,g)}{\omega(g,g^{-1}h^{-1},hg^{-1}h^{-1})\omega(hg,h^{-1},hg^{-1}h^{-1})\omega(hg,g,g^{-1}h^{-1})} \\
&= \frac{\omega(h,g,g^{-1}h^{-1})\omega(g,g^{-1}h^{-1},h)}{\omega(g^{-1},g^{-1},g)\omega(g^{-1},g^{-1}h^{-1},h)\omega(g^{-1},h^{-1},hg^{-1}h^{-1})\omega(g,g^{-2}h^{-1},hg)}\cdot \\
&\frac{\omega(g,g^{-1},h^{-1})\omega(g,g^{-1}h^{-1},h)\omega(g^{-1}h^{-1},h,g)\omega(g^{-2}h^{-1},h,g)}{\omega(g,g^{-1}h^{-1},hg^{-1}h^{-1})\omega(hg,g,g^{-1}h^{-1})} \\
&= \frac{\omega(h,g,g^{-1}h^{-1})\omega(g,g^{-1}h^{-1},h)}{\omega(g^{-1},g^{-1},g)\omega(g^{-1},g^{-1}h^{-1},h)\omega(g,g^{-2}h^{-1},hg)}\cdot \\
&\frac{\omega^2(g,g^{-1},h^{-1})\omega(g,g^{-1}h^{-1},h)\omega(g^{-1}h^{-1},h,g)\omega(g^{-2}h^{-1},h,g)}{\omega(g,g^{-1},g^{-1}h^{-1})\omega(hg,g,g^{-1}h^{-1})} \\
&= \frac{\omega(h,g,g^{-1}h^{-1})\omega^2(g,g^{-1}h^{-1},h)\omega^2(g,g^{-1},h^{-1})\omega^2(g^{-1}h^{-1},h,g)}{\omega(g,g^{-2}h^{-1},hg)\omega(g,g^{-1},g^{-1}h^{-1})\omega(hg,g,g^{-1}h^{-1})\omega(g^{-1},g^{-1}h^{-1},hg)} \\
&= \frac{\omega(h,g,g^{-1}h^{-1})\omega^2(g,g^{-1}h^{-1},h)\omega^2(g,g^{-1},h^{-1})\omega^2(g^{-1}h^{-1},h,g)}{\omega(hg,g,g^{-1}h^{-1})} \\
&= \frac{\omega(h,g,g^{-1}h^{-1})\omega^2(g^{-1}h^{-1},h,g)}{\omega^2(g^{-1},h^{-1},h)\omega(hg,g,g^{-1}h^{-1})}
\end{aligned}
$$

Why is it hard?

- Computationally intensive
- High level of abstraction

Solution: Haskell

```
data ColoredGraph = ColoredGraph
                  { vertices      :: [InteriorVertex]
                  , edges         :: [Edge]
                  , disks         :: [Disk]
                  , perimeter     :: Disk -> [Edge]

                  -- image under contractions
                  , imageVertex    :: Vertex -> Vertex

                  , edgeTree      :: Vertex -> Tree Edge
                  , morphismLabel :: InteriorVertex
                                      -> Morphism
                  , objectLabel   :: Edge -> Object
                  }
```

Objects

```
data Object
  = OVar InitialEdge
  | One
  | Star Object
  | TensorO Object Object
```

Morphisms

```
data Morphism
  = Phi
  | Id Object
  | Lambda Object
  | LambdaI Object
  | Rho Object
  | RhoI Object
  | Alpha Object Object Object
  | AlphaI Object Object Object
  | Coev Object
  | Ev Object
  | TensorM Morphism Morphism
  | PivotalJ Object
  | PivotalJI Object
  | Compose Morphism Morphism
```

Local Moves

```
tensor :: Disk -> State Stringnet ()
contract :: Edge -> State Stringnet InteriorVertex
connect :: Edge -> Edge -> Disk -> State Stringnet Edge
addCoev :: Edge
  -> State Stringnet (InteriorVertex, Edge, Edge)
```

Vertex Hom-Space Moves

```
associateL ::
  InteriorVertex -> Tree Edge -> State Stringnet (Tree Edge)
associateR ::
  InteriorVertex -> Tree Edge -> State Stringnet (Tree Edge)
isolateR :: InteriorVertex -> State Stringnet ()
isolateL :: InteriorVertex -> State Stringnet ()
zMorphism :: Object -> Object -> Morphism -> Morphism
zRotate :: InteriorVertex -> State Stringnet ()
isolate2 :: Edge -> Edge -> InteriorVertex
  -> State Stringnet ()
```

Two-Complex Datatypes (partially specific to braid move)

```
data Puncture = LeftPuncture | RightPuncture
data InteriorVertex = Main | Midpoint Edge | Contraction Edge
data Vertex = Punc Puncture | IV InteriorVertex
data InitialEdge = LeftLoop | RightLoop | LeftLeg | RightLeg
data Edge
  = IE InitialEdge
  | FirstHalf Edge
  | SecondHalf Edge
  | Connector Edge Edge Disk
  | TensorE Edge Edge
  | Reverse Edge
data Disk = Outside | LeftDisk | RightDisk | Cut Edge
```

Braid move

```
(_,l1,r1) <- addCoev $ IE LeftLoop
(_,l2,r2) <- addCoev $ IE LeftLeg
(_,r13,l3) <- addCoev r1
(_,_,r4) <- addCoev $ IE RightLoop
e1 <- connect (rev l1) r2 LeftDisk
e2 <- connect (rev l2) (rev r13) (Cut $ e1)
e3 <- connect l3 r4 Outside
contract e1
contract e2
contract e3
tensor (Cut $ rev e1)
tensor (Cut $ rev e2)
tensor (Cut $ rev e3)
v <- contract r4
```

+ some reassociating

TambaraYamagami types ($A = \mathbb{Z}_N$)

```
newtype AElement = AElement Int

newtype RootOfUnity = RootOfUnity AElement

data Scalar =  Scalar
  { coeff :: [Int]
  , tauExp :: Sum Int
  }
```

A scalar is represented as $\tau^k \sum_{i=0}^{N-1} a_i \zeta_N^i$

TambaraYamagami types

```
data SimpleObject =
  -- Group-element-indexed simple objects
  AE !AElement

  -- non-group simple object
  | M

newtype Object = Object
  { multiplicity_ :: [Int]
  }
```

TambaraYamagami types

```
data Morphism = Morphism
  { domain   :: Object
  , codomain :: Object
  , subMatrix_ :: [M.Matrix Scalar]
  }
```

TambaraYamagami types

```
data BasisElement = BasisElement
  { initialLabel :: S.InitialEdge -> SimpleObject
  , oneIndex :: Int
  }
```

Next steps

- Verify braid relations
- Compare with Ising R-matrices
- Optimize composition to be local wrt tensor products
- Other mapping class group generators
- Other categories

Thanks

Thanks for listening!