Representation Theory of Graph Isomorphism

GCT Project on Graph Isomorphism: Work in Progress

Jacob Urisman

In collaboration with Joshua Grochow; Advisor: Cameron Musco

University of Massachussetts Amherst

jurisman@umass.edu

February 25, 2025

Brief Overview

1. First Part: Polynomials and WL
    - GI
    - Polynomials
    - WL
2. Second Part: GCT and Representation Theory
    - Representation Theory
    - Mini Theorem
    - Current Results
    - Future Directions

Graph Isomorphism

Definition (Graph Isomorphism Problem or GI)

A decision problem. Given two graphs $\mathcal{G}, \mathcal{H}$, is $\mathcal{G} \cong \mathcal{H}$?

Graph Isomorphism

Definition (Graph Isomorphism Problem or GI)

A decision problem. Given two graphs $\mathcal{G}, \mathcal{H}$, is $\mathcal{G} \cong \mathcal{H}$?

Why do we care?

Graph Isomorphism

Definition (Graph Isomorphism Problem or GI)

A decision problem. Given two graphs $\mathcal{G}, \mathcal{H}$, is $\mathcal{G} \cong \mathcal{H}$?

Why do we care? GI is in NP but is not known to be in P and has not been proven to be NP-complete.

Polynomials and Motivation

We can "represent" graphs with polynomials. Let M be the adjacency matrix of an arbitrary graph on n nodes. For each row i and column j of M, let x_{ij} be a variable which takes a value for a fixed graph.

Polynomials and Motivation

We can "represent" graphs with polynomials. Let M be the adjacency matrix of an arbitrary graph on n nodes. For each row i and column j of M, let x_{ij} be a variable which takes a value for a fixed graph.

We can encode graph properties via sets of polynomials vanishing.

Example

$\sum_i x_{ii} = 0 \iff \mathcal{G}$ is a loopless graph.

Example

$\{x_{ij} - x_{ji} = 0 \mid \forall i, j\} \iff \mathcal{G}$ is undirected.

WL and More Motivation

The Weisfeiler-Lehman (WL) algorithm is a GI algorithm that has many nice combinatorial and first order logical properties. The quantifier depth of a first order logical statement on two graphs is the number of rounds of WL required to verify that statement, and the number of variables used is the WL-dimension minus one.

WL and More Motivation

The Weisfeiler-Lehman (WL) algorithm is a GI algorithm that has many nice combinatorial and first order logical properties. The quantifier depth of a first order logical statement on two graphs is the number of rounds of WL required to verify that statement, and the number of variables used is the WL-dimension minus one.

[BG15] showed a connection between WL-dimension and degree of polynomials used in algebraic proof systems. This suggests some algebraic properties of WL that we wanted to explore via representation theory.

WL and More Motivation

The Weisfeiler-Lehman (WL) algorithm is a GI algorithm that has many nice combinatorial and first order logical properties. The quantifier depth of a first order logical statement on two graphs is the number of rounds of WL required to verify that statement, and the number of variables used is the WL-dimension minus one.

[BG15] showed a connection between WL-dimension and degree of polynomials used in algebraic proof systems. This suggests some algebraic properties of WL that we wanted to explore via representation theory.

Central to WL is easy access to node degrees. With that in mind, we chose our representation in such a way that polynomials representing node degree had degree 1. Note that degree will have two crucial but independent meanings going forward, so polynomial degree will be in standard text and node *degree* will be italicized.

Our Polynomials

Example (October-January)

$$V_d^* = \text{Span}\{\sum_j x_{ij} - d \mid \forall i\}$$

Each polynomial will vanish for a graph $\mathcal{G}$ if and only if the *degree* of the ith vertex of $\mathcal{G}$ is d. Thus, we have degree 1 polynomials that represent *degree*.

Back to WL: The Red Herring

Can we simulate quantified statements using products of polynomials?

Back to WL: The Red Herring

Can we simulate quantified statements using products of polynomials? Trying to build up inductively. Let f_d be a logical statement who's polynomial requires degree d and let p_f be a polynomial which simulates it.

FOL Term	Polynomial	Degree
$E(i,j)$	x_{ij}	1
$\neg f_d$	$1^d - p_f$	d
$f_d \wedge g_e$	$p_f p_g$	$d+e$
$f_d \vee g_e \equiv \neg(\neg f_d \wedge \neg g_e)$	$1^{d+e} - (1^d - p_f)(1^e - p_g)$	$d+e$
$\forall i : f(i)_{d_i}$	$\prod_i p_f(i)$	$\sum_i d_i$
$\exists i : f(i)_{d_i}$	$1^{\sum_i d_i} - \prod_i (1^{d_i} - p_f(i))$	$\sum_i d_i$

Note that variables of the polynomials do not correspond one-to-one with variables of the logic of graphs. In some sense, the x_{ij}'s correspond to pairs of variables in FO logic, i.e. the universe can be thought of as $\mathcal{V} \times \mathcal{V}$.

Negative Result?

It turns out, even ignoring the degree blowup for quantifiers, it seems that there is no correspondence between the polynomial degrees and WL-dimension.

The degree was completely dependent on the number of calls to the edge predicate in the logical formula.

The Takeaway

In polynomials with variables corresponding to edges, the degrees of polynomials relate to how many times we look at edges.

Shifting Gears

We are going to introduce some new math: representation theory, and some more motivation.

Shifting Gears

We are going to introduce some new math: representation theory, and some more motivation.

Note that in all of our examples so far, the sets of polynomials we were looking at were isomorphism invariant

Proposition

If the vanishing of a polynomial is invariant under the action of S_n then the polynomial vanishes on all isomorphic copies of a graph. The converse does not hold.

$$\mathcal{Z}(V) = \mathcal{Z}(S_n \curvearrowright V) \implies \forall p \in V : (p|_{\mathcal{G}} \sim p|_{\mathcal{H}} \iff \mathcal{G} \cong \mathcal{H})$$

We can use representation theory to organize polynomially defined properties.

GCT

Geometric Complexity Theory (GCT) is an area of complexity theory that I have been interested in for the past few years. The most basic idea is to use tools from representation theory and algebraic geometry on open problems in complexity theory.

Geometric Complexity Theory (GCT) is an area of complexity theory that I have been interested in for the past few years. The most basic idea is to use tools from representation theory and algebraic geometry on open problems in complexity theory.

Representation Theory

Represent objects with polynomials.

Geometric Complexity Theory (GCT) is an area of complexity theory that I have been interested in for the past few years. The most basic idea is to use tools from representation theory and algebraic geometry on open problems in complexity theory.

Representation Theory

Represent objects with polynomials.

Algebraic Geometry

See when those polynomials vanish.

GCT

Geometric Complexity Theory (GCT) is an area of complexity theory that I have been interested in for the past few years. The most basic idea is to use tools from representation theory and algebraic geometry on open problems in complexity theory.

Representation Theory

Represent objects with polynomials.

Algebraic Geometry

See when those polynomials vanish.

GCT was developed in the early 00's and has already led to many interesting results.

Representation Theory

Definition (representation)

Given a group G and a vectorspace V over a field F, a *representation* is a homomorphism $\rho : G \to GL(V)$ such that $\rho(g_1 g_2) = \rho(g_1)\rho(g_2), \forall g_1, g_2 \in G$. If $\dim_F V = k < \infty$, we can use the alternate notation $\rho : G \to GL_k(F)$.

We sometimes refer to the vector space V as the representation if the homomorphism is obvious. Note: $G \curvearrowright V$.

Definition (subrepresentation)

Given a representation (ρ, V), a *subrepresentation* is a subspace W of V that is invariant under the action of G. Specifically, $\rho(g)W \subseteq W, \forall g \in G$. Another way to denote this is $\rho|_W(g) = \rho(g)|_W$.

What we really mean is that if we instead took a homomorphism of group elements into just the subspace W it would be equivalent to how the homomorphism of elements into V acts on W.

More Representation Theory

Definition (irreducible representation)

An *irreducible representation* or irrep is a representation that has no nonzero proper subrepresentation.

Definition (Polynomial Representation)

Given a representation V with dimension $k < \infty$ over a field F, a *polynomial representation* is a polynomial ring $V^* \subseteq F[X]$ for $X = \{x_{ij} \mid 1 \leq i, j \leq k\}$ and k^2 maps $f_{ij} : GL(V) \to F \coloneqq x \mapsto x_{ij}$.

Vanishing Polynomials and Irreps

The vanishing of irreps is a huge tool in representation theory.

Vanishing Polynomials and Irreps

The vanishing of irreps is a huge tool in representation theory.

Theorem (Maschke)

If F has characteristic 0, every representation is a direct sum of irreducible representations.

Vanishing Polynomials and Irreps

The vanishing of irreps is a huge tool in representation theory.

Theorem (Maschke)

If F has characteristic 0, every representation is a direct sum of irreducible representations.

Corollary

For any group G, the set of possible irreps of any representation of G is determined by G.

Vanishing Polynomials and Irreps

The vanishing of irreps is a huge tool in representation theory.

Theorem (Maschke)

If F has characteristic 0, every representation is a direct sum of irreducible representations.

Corollary

For any group G, the set of possible irreps of any representation of G is determined by G. For S_n, we know the irreps very well.

Vanishing Polynomials and Irreps

The vanishing of irreps is a huge tool in representation theory.

Theorem (Maschke)

If F has characteristic 0, every representation is a direct sum of irreducible representations.

Corollary

For any group G, the set of possible irreps of any representation of G is determined by G. For S_n, we know the irreps very well.

Remember that representations are vectorspaces, meaning they have bases. Polynomial representations have sets of polynomials as bases. Any polynomial in a polynomial representation is a linear combination of irrep basis polynomials.

Vanishing Polynomials and Irreps

The vanishing of irreps is a huge tool in representation theory.

Theorem (Maschke)

If F has characteristic 0, every representation is a direct sum of irreducible representations.

Corollary

For any group G, the set of possible irreps of any representation of G is determined by G. For S_n, we know the irreps very well.

Remember that representations are vectorspaces, meaning they have bases. Polynomial representations have sets of polynomials as bases. Any polynomial in a polynomial representation is a linear combination of irrep basis polynomials.

Proposition

A polynomial vanishes if and only if every irrep it belongs to vanishes.

We have a strategy to analyze the polynomials! We decide on a representation and find the irreps, which representation theory gives us tools for.

Definition (GI)

Given two n-vertex graphs $\mathcal{G}, \mathcal{H}, \mathcal{G} \cong \mathcal{H} \iff \exists \sigma \in S_n : \mathcal{G} = \sigma(\mathcal{H})$.

Example

Let $G = S_n, F = \mathbb{C}, V$ be the space of adjacency matrices of graphs with n vertices. For generality, let the entries of these adjacency matrices be $\{0, \lambda\}$ for $\lambda \in \mathbb{C}$.

Example

$$V_d^* = \text{Span}\{\sum_j x_{ij} - dz \mid \forall i\}$$

Where, z is a variable that takes value λ and d is the *degree* we are interested in.

ρ maps permutations in S_n to permutation matrices. $S_n \curvearrowright V$ in the same way as in the definition of GI on this slide (via conjugation). f_{ij} sends entries of graph adjacency matrices to variables x_{ij}.

The Irreps

Proposition

For the degree 1 polynomials, there are 3 copies of the trivial representation, 3 copies of the reflection representation (also called the standard representation), and one copy each of the representations of shape $(n-2,2)$ and $(n-2,1,1)$.

The Irreps

Proposition

For the degree 1 polynomials, there are 3 copies of the trivial representation, 3 copies of the reflection representation (also called the standard representation), and one copy each of the representations of shape $(n-2, 2)$ and $(n-2, 1, 1)$.

Corollary

We can immediately distinguish graphs on the following properties:

The Irreps

Proposition

For the degree 1 polynomials, there are 3 copies of the trivial representation, 3 copies of the reflection representation (also called the standard representation), and one copy each of the representations of shape $(n-2,2)$ and $(n-2,1,1)$.

Corollary

We can immediately distinguish graphs on the following properties:

- *existence of self-loops*

The Irreps

Proposition

For the degree 1 polynomials, there are 3 copies of the trivial representation, 3 copies of the reflection representation (also called the standard representation), and one copy each of the representations of shape $(n-2,2)$ and $(n-2,1,1)$.

Corollary

We can immediately distinguish graphs on the following properties:

- *existence of self-loops*
- *existence of edges*

The Irreps

Proposition

For the degree 1 polynomials, there are 3 copies of the trivial representation, 3 copies of the reflection representation (also called the standard representation), and one copy each of the representations of shape $(n-2, 2)$ and $(n-2, 1, 1)$.

Corollary

We can immediately distinguish graphs on the following properties:

- *existence of self-loops*
- *existence of edges*
- *different levels of regularity*

The Irreps

Proposition

For the degree 1 polynomials, there are 3 copies of the trivial representation, 3 copies of the reflection representation (also called the standard representation), and one copy each of the representations of shape $(n-2,2)$ and $(n-2,1,1)$.

Corollary

We can immediately distinguish graphs on the following properties:

- *existence of self-loops*
- *existence of edges*
- *different levels of regularity*
  - *distinguish all or no self-loops from some self loops*
  - *every node has the same indegree*
  - *every node has the same outdegree*
  - *combinations of the above*

Combinations

We can also look at what different multiplicities of the same irreducible representation vanishing means. This is not interesting for the trivial representations. For the reflection representations, not only does the number of vanishing reflection representations contain information, but which ones specifically vanish contains information too.

Proposition

For each vertex i of the graph, consider a 3-tuple with entries equalling the number of self-loops of i, the *indegree* of i, and the *outdegree* of i. For the whole graph, we get a point cloud in $\mathbb{R}^3$. If just one reflection representation vanishes, the point cloud lies on a plane. If two vanish, the point cloud lies on a line. If all three vanish, the point cloud is just a single point. In this case, the graph is also regular.

More on GCT

A key pillar of GCT is the notion of "characterization by symmetries." You can think of this as how much information in a representation is encoded by the orbits of the group action.

More on GCT

A key pillar of GCT is the notion of “characterization by symmetries.” You can think of this as how much information in a representation is encoded by the orbits of the group action.

Graphs with a lot of symmetry are very hard for GI algorithms.

More on GCT

A key pillar of GCT is the notion of "characterization by symmetries." You can think of this as how much information in a representation is encoded by the orbits of the group action.

Graphs with a lot of symmetry are very hard for GI algorithms.

For representations of S_n, irrep multiplicities encode symmetry. The number of orbits of X under the action of S_n is equal to the multiplicity of the trivial representation in the polynomial repesentation of X.

Characterization by Symmetries

So which graphs are characterized by their symmetries?

Characterization by Symmetries

So which graphs are characterized by their symmetries? We are looking for graphs where the group of their symmetries (as a permutation group or explicit group and action) uniquely determines the graph.

Characterization by Symmetries

So which graphs are characterized by their symmetries? We are looking for graphs where the group of their symmetries (as a permutation group or explicit group and action) uniquely determines the graph.

We can use automorphism groups for this!

Definition (Automorphism Group)

Given a graph $\mathcal{G}$, the automorphism group G of $\mathcal{G}$ is a permutation group where $\forall \sigma \in G : \sigma(\mathcal{G}) \cong \mathcal{G}$.

Characterization by Symmetries

So which graphs are characterized by their symmetries? We are looking for graphs where the group of their symmetries (as a permutation group or explicit group and action) uniquely determines the graph.

We can use automorphism groups for this!

Definition (Automorphism Group)

Given a graph $\mathcal{G}$, the automorphism group G of $\mathcal{G}$ is a permutation group where $\forall \sigma \in G : \sigma(\mathcal{G}) \cong \mathcal{G}$.

Representation Theory Under the Hood

Irrep multiplicities for an automorphism group $G \leq S_n$ are

$$\langle \mathrm{Ind}_G^{S_n} \chi_{triv}, \chi \rangle \overset{Frobenius}{=} \langle \chi_{triv}, \mathrm{Res}_G^{S_n} \chi \rangle_G = \frac{1}{|G|} \sum_{g \in G} \chi(g)$$

for χ of irrep of S_n.

Mini Theorem

Proposition (Grochow-U.)

Graphs with conjugate automorphism groups cannot be distinguished by multiplicity information.

Definition (Cycle Index)

The *cycle index* of a permutation group is the multiset of its cycle types.

Mini Theorem

Proposition (Grochow-U.)

Graphs with conjugate automorphism groups cannot be distinguished by multiplicity information.

Definition (Cycle Index)

The *cycle index* of a permutation group is the multiset of its cycle types.

Theorem (Grochow-U.)

Two graphs are indistinguishable by irrep multiplicities if and only if their automorphism groups have the same cycle index.

Mini Theorem

Proposition (Grochow-U.)

Graphs with conjugate automorphism groups cannot be distinguished by multiplicity information.

Definition (Cycle Index)

The *cycle index* of a permutation group is the multiset of its cycle types.

Theorem (Grochow-U.)

Two graphs are indistinguishable by irrep multiplicities if and only if their automorphism groups have the same cycle index.

This is cool! This is a combinatorial result for a representation theoretic property. Makes checking irrep multiplicities extremely fast with code.

Code

```
def duplicate_two_closed(n): # Function takes n and returns the (order, group) of any 2-closed subgroup of S_n
whose cycle index is indistinguishable from another
    G=SymmetricGroup(n)
    subgroups=G.conjugacy_classes_subgroups()
    print("Filtering 2-closed subgroups")
    two_closed_subgroups=[sg for sg in subgroups if is_two_closed(sg)]
    orders=[sg.order() for sg in two_closed_subgroups]
    print("Checking for duplicate orders")
    duplicate_orders=[order for order in orders if orders.count(order)>1]
    duplicate_order_subgroups=[sg for sg in two_closed_subgroups if sg.order() in duplicate_orders]
    cycle_indices=[]
    total=len(duplicate_order_subgroups)
    print("There are "+str(total)+" duplicate order subgroups that are 2-closed")
    for sg in duplicate_order_subgroups:
        cycle_indices.append(sorted([g.cycle_type() for g in sg.list()]))
    indices=[i for i, elem in enumerate(cycle_indices) if cycle_indices.count(elem)>1]
    all_duplicate_subgroups=[(duplicate_orders[i],duplicate_order_subgroups[i]) for i in indices]
    return all_duplicate_subgroups

Tabnine | Edit | Test | Fix | Explain | Document
def is_two_closed(G):
    try:
        return str(gap.IsTransitive(G))=="true" and gap.Order(gap.TwoClosure(G))==G.order()
    except:
        print(G,'produced an error')
```

Current Results

We have checked all subgroups of the symmetric group for $n \leq 13$. We have found a single pair of automorphism groups up to conjugacy that are indistinguishable by irrep multiplicities

Current Results

We have checked all subgroups of the symmetric group for $n \leq 13$. We have found a single pair of automorphism groups up to conjugacy that are indistinguishable by irrep multiplicities ($\mathbb{Z}_4 \circ D_4$ and $\mathbb{Z}_2^2 \rtimes \mathbb{Z}_4$).

Current Results

We have checked all subgroups of the symmetric group for $n \leq 13$. We have found a single pair of automorphism groups up to conjugacy that are indistinguishable by irrep multiplicities ($\mathbb{Z}_4 \circ D_4$ and $\mathbb{Z}_2^2 \rtimes \mathbb{Z}_4$).

We also know there exists a pair which are subgroups of S_{27}.

Future Directions

- We hope to explore higher n (memory bottleneck with current methods).
- We want to better understand the graphs which have these groups as automorphism groups.
- We ideally hope to classify when pairs of indistinguishable automorphism groups arise.

Open questions:

- Can we develop an efficient GI algorithm for highly regular graphs?
- Prove unconditionally that WL and the approach from part 1 are incomparable.

Thank You!

Berkholz, Christoph and Martin Grohe. *Limitations of Algebraic Approaches to Graph Isomorphism Testing.* 2015. arXiv: 1502.05912 [cs.CC]. URL: https://arxiv.org/abs/1502.05912.