

P475 – Special Topics in Quantum Mechanics

Semester Project Report

Jyotirmaya Shivottam

1711069

jyotirmaya.shivottam@niser.ac.in

Abstract

In this term paper project, we have looked into Rudinger et al's paper [main], on the application of non-interacting multiparticle continuous-time quantum random walks (CTQWs) to the Graph Isomorphism (GI) problem for strongly regular graphs (SRGs). We have implemented their algorithm and tried to replicate their results and contrasted them with a classical GI test. We have also explored the empirical run-time and space complexity of their algorithm.

I. Introduction

Before proceeding into the overview of the work done by the authors, we require the definitions of a few terms.

I. Key terms

Definition 1. Two graphs G and H are said to be isomorphic, if there exists a bijection $\phi : V(G) \rightarrow V(H)$ such that for every edge, $e \in E(G)$, there exists an edge, $f \in E(H)$, such that $\phi(e) = f$. Here, V and E denote the vertex and edge sets respectively.

Definition 2. The adjacency matrix of a graph G is a $N \times N$ matrix, A , such that

$$A_{ij} = \begin{cases} 1, & \text{if } (i, j) \in E(G). \\ 0, & \text{otherwise,} \end{cases}$$

where, N is the number of vertices in G . The adjacency matrix of a graph G is symmetric and has $2m$ non-zero entries, where m is the number of edges in G .

In terms of the adjacency matrices, A_G & A_H , of two graphs, the isomorphism can be defined as the existence of a permutation matrix, P , such that $PA_G P^T = A_H$. Here, the permutation matrix represents a relabeling of the vertices of G to the vertices of H . As the name suggests, the GI problem is to ascertain whether two graphs are isomorphic or not. Fig. ?? presents a simple example of the GI problem. The GI problem is in

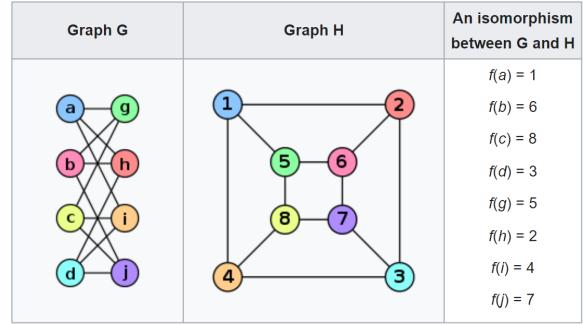


Figure 1: A simple example of the Graph Isomorphism problem, highlighting the relabeling of vertices [wiki2]

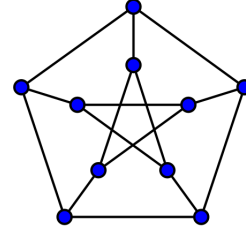


Figure 2: The Petersen graph, $\text{srg}(10, 3, 0, 1)$ [wiki]

the NP complexity class and is purported to be NP-Intermediate (if $P \neq NP$).

Definition 3. A strongly regular graph (SRG), denoted as $\text{srg}(v, k, \lambda, \mu)$, is a distance-regular graph with the following properties:

1. Every two adjacent vertices have λ common neighbors.
2. Every two non-adjacent vertices have μ common neighbors.

Usually, the notation, $\text{srg}(v, k, \lambda, \mu)$, denotes a family of graphs. Fig. ?? shows an example of a SRG is the Petersen graph, parameterized as $\text{srg}(v = 10, k = 3, \lambda = 0, \mu = 1)$. The adjacency matrix of a SRG satisfies certain useful identities

that have been listed below:

$$\begin{aligned} A^2 &= (k - \mu)I + \mu J + (\lambda - \mu)A, \\ JA &= AJ = kA, \\ J^2 &= NJ, \end{aligned}$$

where, I is the identity matrix, J is the all-ones matrix, and N is number of vertices. Using these, we find that $\{I, J, A\}$ forms a commutative 3-dimensional algebra, which leads us to:

$$A^n = \alpha_n I + \beta_n J + \gamma_n A,$$

where, $\alpha_n, \beta_n, \gamma_n$ are constants dependent on the SRG under consideration, hence also called SRG family parameters.

II. Issues with distinguishing SRGs

In principle, the GI problem can be retooled as obtaining a canonical labeling via the automorphism group, which is also what many classical approaches exploit. However, there are several issues with this approach, some of which are listed below:

1. The automorphism group of a SRG family is non-trivial. Calculating it is computationally intractable at the moment.
2. Graphs in the same SRG family are co-spectral.
3. Each SRG signature defining a family can contain several graphs, for instance, $\text{srg}(36, 15, 6, 6)$ has 32,548 graphs, which makes calculating the automorphism group of the entire SRG family intractable.

III. Classical GI tests and limitations

There are several classical tests for GI, such as the *Weisfeiler-Lehman (WL) test*, which is an iterative procedure that uses heuristic-based color-refinement to produce canonical vertex-label sets. This test gives a necessary but not sufficient condition for GI. In particular, the test fails in the presence of graph automorphisms, i.e., mappings from the graph to itself that leave the graph unchanged. And, regular graphs usually have a large number of automorphisms. The WL-test fails for SRGs, like the Petersen graph. Another classical test, which is also the state-of-the-art, is the *VF2++ test [vf2]* which is a recursive algorithm that uses *partial edge-matching* for subgraph isomorphism. While this test can be used to test SRG GI, the memory requirements grow exponentially with the number of vertices. Currently, the best classical algorithm has a worst-case time

complexity of $O(c^{\sqrt{N} \log N})$ [main], where N is the number of vertices.

In the paper, the authors have investigated the dynamics of non-interacting multiparticle continuous-time quantum random walks (CTQWs) and their applicability to the GI problem for SRGs. In the next section, we discuss their approach.

II. Paper's approach

The paper defines continuous-time quantum walks (CTQW) on graphs, using the Hubbard model, without the short-range interaction term, where each site corresponds to a vertex.

$$H = - \sum A_{ij} c_i^\dagger c_j$$

Here, c_i^\dagger is the creation operator for a particle at site i , and A_{ij} is the adjacency matrix of the graph. For bosons, since $[c_i, c_j^\dagger] = \delta_{ij}$ and $[c_i, c_j] = [c_i^\dagger, c_j^\dagger] = 0$, we have symmetrized basis states, with multi-occupancy, while for fermions, $\{c_i, c_j^\dagger\} = \delta_{ij}$ and $\{c_i, c_j\} = \{c_i^\dagger, c_j^\dagger\} = 0$ implies anti-symmetrized basis states, with single-occupancy. This yields the p -boson or p -fermion Hamiltonian as:

$$\begin{aligned} {}_B \langle i_1, \dots, i_p | H_{p,B} | j_1, \dots, j_p \rangle_B &= \\ - {}_B \langle i_1, \dots, i_p | A^{\oplus p} | j_1, \dots, j_p \rangle_B \\ {}_F \langle i_1, \dots, i_p | H_{p,B} | j_1, \dots, j_p \rangle_F &= \\ {}_F \langle i_1, \dots, i_p | A^{\oplus p} | j_1, \dots, j_p \rangle_F \end{aligned}$$

where, $A^{\oplus p} = A \otimes I \otimes \dots + \dots + I \otimes \dots A$. From here, the evolution operator is defined as usual, in the particles-on-vertices basis:

$$U(t) = e^{-iHt}$$

The elements of the resulting evolution matrix are also termed as *Green's functions*, in the *correlator* sense. These Green's functions form a unique certificate or signature for the graph. This forms the theoretical basis of the paper's approach. Using this, the algorithm for using CTQW to distinguish the non-isomorphic SRGs is as follows:

1. For each graph in the SRG family:
 - (a) Begin with the (complex) evolution matrix U .
 - (b) Take the magnitude of each element.
 - (c) Write all the (real) entries in a list, X_A .
 - (d) Sort the list.

Forming the evolution operators $\mathbf{U} = e^{-it\mathbf{H}}$, we have

$$\mathbf{U}_A = \begin{pmatrix} \left(\frac{\cos(\sqrt{2}t)}{2} + \frac{1}{2}\right) & -\frac{i \sin(\sqrt{2}t)}{\sqrt{2}} & \left(\frac{\cos(\sqrt{2}t)}{2} - \frac{1}{2}\right) \\ -\frac{i \sin(\sqrt{2}t)}{\sqrt{2}} & \cos(\sqrt{2}t) & -\frac{i \sin(\sqrt{2}t)}{\sqrt{2}} \\ \left(\frac{\cos(\sqrt{2}t)}{2} - \frac{1}{2}\right) & -\frac{i \sin(\sqrt{2}t)}{\sqrt{2}} & \left(\frac{\cos(\sqrt{2}t)}{2} + \frac{1}{2}\right) \end{pmatrix} \quad (28)$$

and

$$\mathbf{U}_B = \begin{pmatrix} \cos(\sqrt{2}t) & -\frac{i \sin(\sqrt{2}t)}{\sqrt{2}} & \frac{i \sin(\sqrt{2}t)}{\sqrt{2}} \\ -\frac{i \sin(\sqrt{2}t)}{\sqrt{2}} & \left(\frac{\cos(\sqrt{2}t)}{2} + \frac{1}{2}\right) & \left(\frac{1}{2} - \frac{\cos(\sqrt{2}t)}{2}\right) \\ \frac{i \sin(\sqrt{2}t)}{\sqrt{2}} & \left(\frac{1}{2} - \frac{\cos(\sqrt{2}t)}{2}\right) & \left(\frac{\cos(\sqrt{2}t)}{2} + \frac{1}{2}\right) \end{pmatrix}. \quad (29)$$

Figure 3: Sign-ambiguities in U (a potential limitation) [main2]

2. Compare the lists pairwise for all graphs using $\Delta = \sum_v |X_A[v] - X_B[v]|$.

If $\Delta \neq 0$, then A and B are non-isomorphic or distinguished. The converse is however not true, because non-isomorphic non-distinguished graphs (false negatives) can also have $\Delta = 0$. As an example, for the 3-particle walk, U can be decomposed as: $U_{3B} = U_1^{\otimes 3}$ or $U_{3F} = \bar{U}_1^{\otimes 3}$, where $U_1 = e^{iAt}$ and $\bar{U}_1 = e^{-iAt}$. Recalling the 3-algebra, we can decompose $U_1 = \alpha_n I + \beta_n J + \gamma_n A$, where the constants depend only on SRG family parameters and t . Therefore, we have the following intuition as to why this algorithm would work:

1. All possible values of Green's functions are determined by the family parameters ($p = 3$).
2. Distinguishing power of the walks comes from the existence of at least one Green's function with different multiplicities for nonisomorphic graphs in the same family.

Also, note that the reason for using the magnitude of Green's functions is that depending on the choice of basis, sign ambiguities can arise in U (e.g. see Fig. ??). Taking a magnitude removes this ambiguity.

The authors have tested their approach on the SRG families listed in Fig. ?? and Fig. ??. Based on these results, the authors have shown that 3-particle walks have significant, but not universal, distinguishing power on SRGs, something that 1 and 2-particle walks do not have [main2]. The authors have also shown that 4-particle walks have remarkably higher distinguishing power, while both bosonic and fermionic walks have similar performance, despite the fermionic walks having a smaller state space, owing to the Pauli exclusion principle.

These calculations are quite computationally intensive. So, we have only been able to replicate a

SRG family (N, k, λ, μ)	No. of graphs	Comparisons	Boson failures	Fermion failures
(16,6,2,2)	2	1	0	0
(16,9,4,6)	2	1	0	0
(25,12,5,6)	15	105	0	0
(26,10,3,4)	10	45	1	1
(28,12,6,4)	4	6	0	0
(29,14,6,7)	41	820	0	0
(35,18,9,9)	227	25 651	38	38
(36,14,4,6)	180	16 110	89	89
(40,12,2,4)	28	378	8	8
(45,12,3,3)	78	3 003	7	7
(49,18,7,6)	147	10 731	21	21
(64,18,2,6)	167	13 861	92	92

Figure 4: SRG families tested for 3-particle walk by the authors [main]

Family (N, k, λ, μ)	3-particle failures	4-fermion failures
(26,10,3,4)	1	0
(35,18,9,9)	38	0
(36,14,4,6)	89	1
(40,12,2,4)	8	0

Figure 5: SRG families tested for 4-particle walk by the authors [main]

portion of their work. We present our results & a short discussion in the next section.

III. Results & Discussion

We have implemented the algorithm as in the paper. The VF2++ algorithm was used for classical GI tests. Due to space constraints in this report, we have moved the details of the data sources, complete code & design decision and extended results to our GitHub repository for this project.

In essence, we have verified the results of the paper for certain SRG families. We have also compared this work with a prior work, where 1- and 2- particle walks were discussed [main2]. The result tables can be found in the GitHub repository as well as in Appendix - A at the end of this document. We list our findings, which overlap with the paper, below:

- 1 & 2-particle walks are unable to distinguish between graphs from the same SRG family since the elements of U are only dependent on the family parameters and are therefore all identical. *Analytical proof of non-distinguishability can be found in [proof].*
- In contrast, $p = 3$ (& $p = 4$) non-interacting CTQW has sufficient power to distinguish

between non-isomorphic graphs from the same SRG family. We have replicated this result for $\text{srg}(16, 6, 2, 2)$ with $p = 3$ $p = 4$, and $\text{srg}(26, 10, 3, 4)$ with only $p = 3$. CTQW with $p = 4$ led to an OutOfMemory error.

As can be perused from the outputs on GitHub, the memory and time consumption of both the classical test as well as the CTQW-based test, are quite high. In the next section, we have discussed some potential extensions to this work.

IV. Potential Extensions

There are several potential optimizations to the algorithm that can be made to this work. We have mentioned these in the GitHub repository at this link, in order to keep this report concise. Below, we list some extensions that could be made to this work. Some of these have been published since the original work came out.

- The authors had purported that including the phase information might improve the algorithm's results, especially for larger p or larger N . However, Mahasinghe et al [maha] have since shown that phase-modified CTQW is still unable to distinguish strongly regular graphs; since Green's functions are still not unique.
- The authors studied the dynamics of CTQW. DTQW or discrete-time quantum walks are also worth considering, since they evolve in a large state space, allowing them to possess more distinguishing power [main3].
- Wang et al [wang] have presented an optimizing heuristic that omits the sorting step.
- Tamascelli et al [tama] have presented a QW-inspired adiabatic algorithm to reduce the search space and translate the problem to 2-SAT.

All the references mentioned here can be found in Appendix - B, near the end of this document.

Appendix A - Result Tables

SRG Label	SRG family	N(Graphs)
A	(16, 6, 2, 2)	2
B	(26, 10, 3, 4)	10
C	(36, 14, 4, 6)	180
D	(40, 12, 2, 4)	45

Table 1: SRG families used in our testing. Labels mentioned here are used in the tables below.

SRG Label	Number of Comparisons	Boson failures	Fermion failures	Peak Mem Usage (in GB)	Avg. CPU Time (in s)
A	1	1	1	0.009	4.8
B	45	45	45	1.39	21.4
C	16,110	16,110	16,110	1.8	976
D	378	378	378	1.53	90.1

Table 2: Results for $p = 1$ or 1-particle walk.

SRG Label	Number of Comparisons	Boson failures	Fermion failures	Peak Mem Usage (in GB)	Avg. CPU Time (in s)
A	1	1	1	0.009	5.1
B	45	45	45	1.92	35.4
C	16,110	NA	NA	3.36	DNF (> 2 hrs)
D	378	378	378	2.73	930

Table 3: Results for $p = 2$ or 2-particle walk. (NA = Not Applicable. DNF = Did Not Finish.)

SRG Label	Number of Comparisons	Boson failures	Fermion failures	Peak Mem Usage (in GB)	Avg. CPU Time (in s)
A	1	0	0	0.5	9.1
B	45	1	1	112	2547.5
C	16,110	NA	NA	> 528	DNF (OoM)
D	378	NA	NA	> 528	DNF (OoM)

Table 4: Results for $p = 3$ or 3-particle walk. (NA = Not Applicable. DNF = Did Not Finish. OoM = Out of Memory)

SRG Label	Number of Comparisons	Boson failures	Fermion failures	Peak Mem Usage (in GB)	Avg. CPU Time (in s)
A	1	0	0	156	525.1
B	45	NA	NA	> 528	DNF (OoM)
C	16,110	NA	NA	> 528	DNF (OoM)
D	378	NA	NA	> 528	DNF (OoM)

Table 5: Results for $p = 4$ or 4-particle walk. (NA = Not Applicable. DNF = Did Not Finish. OoM = Out of Memory)

Appendix B