

Article

Not peer-reviewed version

Number of Hamiltonian Cycles in Certain Bipartite Graphs of Size $4k+2$

Ishan Banerjee *

Posted Date: 9 July 2024

doi: 10.20944/preprints202407.0686.v1

Keywords: bipartite graphs; hamiltonian cycles; cycle cover



Preprints.org is a free multidiscipline platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Article

Number of Hamiltonian Cycles in Certain Bipartite Graphs of Size $4k+2$

Ishan Banerjee

Independent Researcher, India; ishanb3.141@gmail.com

Abstract: We will try to use the approach of labeled cycle cover sum to count the number of hamiltonian cycles in a special type of bipartite graphs, with $4k + 2$ vertices which can also be considered as an alternative method to check hamiltonicity in such bipartite graphs. This graph is such that, there exists at least 1 vertex which is not a part of any cycle of size $4m + 2$ where $m < k$, in $O^*(1.414^n)$ time. The algorithm is inspired by Bjorklund's approach where instead of working on the field of $GF(2^k)$, we will be working on $\mathbb{C}$.

Keywords: bipartite graphs; hamiltonian cycles; cycle cover

1. Introduction

The problem of Hamiltonicity is identified as an *NP* hard problem and the first super polynomial development was an improvement from $O(n^2 2^n)$ to $O^*(1.657^n)$ and $O^*(1.414^n)$ for bipartite graphs. Counting the number of hamiltonian cycles is an associated problem which proceeds a step further. Earlier results give an upper-bound with the condition of bounded degree of the vertices. Apart from that, there exists algorithm using dynamic programming and automorphism group of the graph for bipartite graph. We will use a method motivated from the approach in [1] of labeled cycle cover sum.

2. The Idea

We will use a slight modification of the approach in [1] to check hamiltonicity using labeled cycle covers to count the number of Hamiltonian cycles. We will modify the labeled sum on a graph with odd number of vertices. In order to define the special vertex as mentioned in [1], we are considering a graph where there is at least one vertex which is not a part of any odd length cycle except the hamiltonian cycle. We will then cancel the cycle covers, keeping only the hamiltonian covers. Later we represent the sum as a polynomial which sums over all hamiltonian covers and evaluate the polynomial at the point where all the variables are equal to 1, giving us the number of hamiltonian cycles.

3. Preliminaries

We have the identity for Labeled cycle cover sum as

$$\Lambda(G, L, f) = \sum_{C \in cc(G)} \sum_{g: L \rightarrow C} \prod_{a \in C} f(a, g^{-1}(a))$$

Here $G = (V, E)$ is the graph, L is a labeled set, g is the labeling function (we consider it to be surjective) and $f: E \times 2^L \setminus \{\emptyset\} \rightarrow R$. We will be defining the sum as

$$\Lambda'(G, L, f) = - \sum_{C \in cc(G)} \text{sign}(C) \sum_{g: L \rightarrow C} \prod_{a \in C} f(a, g^{-1}(a))$$

3.1. Cancellation of Non Hamiltonian Cycle Covers

In [1] the author defines the function f , fixing a certain vertex s , in such a way that $f(uv, Z) = f(vu, Z)$ where $u, v \neq s$. We will modify the function to form f' , fixing the vertex s as one through which no odd length cycle passes except the Hamiltonian. (There exists such a vertex by our assumptions) and define the sum Λ' , such that

$$\Lambda'(G, L, f') = - \sum_{C \in cc(G)} \text{sign}(C) \sum_{g: L \rightarrow C} \prod_{a \in C} f'(a, g^{-1}(a))$$

where,

$$f'(uv, Z) = -f'(vu, Z)$$

where $u, v \neq s$

We claim that, the sum will contain only the Hamiltonian covers and the non Hamiltonian covers get canceled. (As a side note, the choice of required vertex will be by going through all the vertices and checking for the non zero result of the algorithm. This does not impact the complexity by a great margin)

Proof:

We note that even cycles don't contribute opposite signs on flipping the orientation of one cycle. For graphs with odd number of vertices, for each cycle cover, there must exist at least one odd cycle.

Consider the following set of cycles as a cycle cover

$$\{C_1, C_2, \dots, C_{n-1}, C_{n*}\}$$

Here C_{n*} is the cycle which passes through s . By our assumption, this is not an odd length cycle.

Without loss of generality, assuming C_1 is an odd cycle, the cycle cover where the orientation of C_1 is opposite, contributes opposite sign to the sum, hence, they together get canceled. Here we note that, since none of the odd length cycles pass through s , we need not worry about the case when the only odd length cycle passes through s . Therefore we have the sum as

$$\Lambda'(G, L, f') = - \sum_{H \in hc(G)} \text{sign}(C) \sum_{g: L \rightarrow C} \prod_{a \in H} f'(a, g^{-1}(a))$$

3.2. Relation between Determinant and Cycle Cover Sum

For solving the labeled sum, Bjorklund had given a method which based on the representation of permanent. Determinant coincided with the permanent in that case as the field had characteristic 2. For this case where $R = \mathbb{C}$, the representation of determinant we will be using is (for an $n \times n$ matrix)

$$A_{ij} = \begin{cases} w_{ij} & : ij \in E \\ 0 & : \text{otherwise} \end{cases}$$

$$\det(A) = \sum_{c \in cc(G)} \text{sign}(C) \prod_{i=1}^n w_{ij}$$

We will now define the matrix

$$M_f(Z)_{ij} = \begin{cases} f(ij, Z) & : ij \in E, Z \neq \emptyset \\ 0 & : \text{otherwise} \end{cases}$$

Next we define the polynomial

$$p(f, r) = \sum_{Y \subseteq L} (-1)^{|Y|} \det \left(\sum_{Z \subseteq Y} r^{|Z|} M_f(Z) \right)$$

Lemma 1 : $-\Lambda'(G, L, f)$, as the coefficient of $r^{|L|}$ in $p(f, r)$.

Proof : Expanding using the definition of determinant we get,

$$\begin{aligned}
p(f, r) &= \sum_{Y \subseteq L} (-1)^{|Y|} \sum_{C \in cc(G)} \text{sign}(C) \sum_{q: C \rightarrow 2^{Y \setminus \{\phi\}}} \prod_{a \in C} r^{|q(a)|} f(a, q(a)) \\
&= \sum_{C \in cc(G)} \text{sign}(C) \sum_{q: C \rightarrow 2^L \setminus \{\phi\}} \sum_{\substack{a \in C \\ q(a) \subseteq Y, Y \subseteq L}} (-1)^{|Y|} \prod_{a \in C} r^{|q(a)|} f(a, q(a))
\end{aligned}$$

For the qs which do not cover the entire L , we will find the same product for all the subsets $Y \subseteq L \setminus \bigcup_{a \in C} q(a)$. As there will be same number of subsets with even and odd cardinalities, the net sum becomes 0 and we are left with the sum,

$$p(f, r) = \sum_{C \in cc(G)} \text{sign}(C) \sum_{q: C \rightarrow 2^L \setminus \{\phi\}, \bigcup_{a \in C} q(a) = L} \prod_{a \in C} r^{|q(a)|} f(a, q(a))$$

which further leads to

$$p(f, r) = \sum_{C \in cc(G)} \text{sign}(C) \sum_{q: C \rightarrow 2^L \setminus \{\phi\}, \bigcup_{a \in C} q(a) = L} r^{\sum_{a \in C} |q(a)|} \prod_{a \in C} f(a, q(a))$$

The coefficient of $r^{|L|}$ is thus

$$\sum_{C \in cc(G)} \text{sign}(C) \sum_{q: C \rightarrow 2^L \setminus \{\phi\}, \bigcup_{a \in C} q(a) = L} \prod_{a \in C} f(a, q(a))$$

where $\sum_{a \in C} |q(a)| = |L|$ which happens when for all $a \neq b$, $q(a) \cap q(b) = \phi$. On inverting the function q , we get $-\Lambda'(G, L, f)$, as the coefficient of $r^{|L|}$.

Lemma 1. The Labeled cycle cover sum, with function f , over the domain $GF(2^k)$, takes $O(|L|^2 n + |L| n^{1+\omega} 2^{|L|} + |L|^2 n^2)$ computations with k such that $2^k > |L|n$, n as the number of vertices and ω being the constant of matrix multiplication.

Proof. Considering the degree of the polynomial $p(f, r)$ in r has the maximum degree $|L|n$, we will have to choose $|L|n$ many points to evaluate at and through interpolation, we will be able to recover the coefficient of $r^{|L|}$. Considering $|L|n$ different points, we can evaluate the polynomial at $a_0, a_1, a_2, \dots, a_{|L|n-1}$. Thus we can use interpolation. The algorithm, for a fixed r , tabulates $T(Y) = \sum_{Z \subseteq Y} r^{|Z|} M_f(Z)$ using Yates' fast Zeta transform which uses $O(|L|2^{|L|})$ field operations. The evaluation of $p(f, r) = \sum_{Y \subseteq Z} \det(T(Y))$ can be done with $O(n^\omega 2^{|L|})$ operations using Bunch and Hopcroft's determinant algorithms, taking $\omega = 2.376$ (Coppersmith-Winograd square matrix multiplication). Summing over all the required operations for the $|L|n$ values of r , we get the time $O(|L|^2 n + |L| n^{1+\omega} 2^{|L|})$. Further considering the time for Lagrange's interpolation, we will further add $|L|^2 n^2$. $\square$

4. Reduction to Bipartite Graphs

We will proceed in a way, very similar to the approach of Bjorklund's paper. For a graph $G = (V_1, V_2, E)$, for a fixed vertex s , Bjorklund had defined variables satisfying $x_{uv} = x_{vu}$ where $u, v \neq s$. In this case we will define $x_{uv} = ix_{vu}$, with $u, v \neq s$

We further have $|V_1| = |V_2| = 2k + 1$ for a certain k .

For every vertex pair $u, v \in V_1$, the neighborhood set is defined as $N(u, v) = \{w : uw \in E, vw \in E\}$. Apart from this, we will have $D = (V_1, F)$, where F contains arcs in both direction for every pair of vertices in V_1 . We will keep the definition of f' intact, which is $f'(uv, \{w\}) = x_{uw}x_{wv}$, with $w \in N(u, v)$. Therefore, we will have $f'(uv, \{w\}) = -f'(vu, \{w\})$.

Lemma 2 : $\Lambda'(D, V_2, f') = \sum_{H \in hc(G)} \prod_{uv \in H} x_{uv}$

Proof : By definition, we have

$$\Lambda'(D, V_2, f') = - \sum_{C \in cc(G)} \text{sign}(C) \sum_{g: V_2 \rightarrow C} \prod_{a \in C} f'(a, g^{-1}(a))$$

By the property of f' , using Lemma 1, we will have

$$\Lambda'(D, V_2, f') = - \sum_{H \in hc(D)} \text{sign}(H) \sum_{g: V_2 \rightarrow C} \prod_{a \in H} f'(a, g^{-1}(a))$$

Further due to the non zero result of $f(a, Z)$ only when X is singleton, we will have

$$\Lambda'(D, V_2, f') = - \sum_{H \in hc(D)} \text{sign}(H) \sum_{q: H \rightarrow V_2} \prod_{a \in H} f'(a, q(a))$$

Plugging in the definition of f' ,

$$\Lambda'(D, V_2, f') = - \sum_{H \in hc(D)} \text{sign}(H) \sum_{g: V_2 \rightarrow C} \prod_{a \in H} x_{uq(uv)} x_{q(uv)v}$$

Here for every $uv \in V_2$, $q(uv) \in N(u, v)$. Further we note that q is one-to-one. Therefore every vertex in V_2 is mapped to by exactly one arc from V_1 . Therefore, we will have

$$\Lambda'(D, V_2, f') = - \sum_{H \in hc(G)} \text{sign}(H) \prod_{uv \in H} x_{uv}$$

Finally, to count the number of hamiltonian cycles, we will plug 1 into each variable and divide by 2.

We note that the cardinality of the label set and the number of vertices in the graph where we are doing the computations is $n/2$. Therefore from Lemma 1, the time taken for computing the sum becomes $O^*(2^{n/2})$.

5. Conclusion

The result in particular might not cover the entire section of computing a $\#P$ complete problem, but this might motivate to consider the importance of cycle covers in the problem in particular. Our method covers a very specific type of bipartite graph where we have a vertex which is not a part of any cycle whose length is of the form $4m + 2$ for some $m < k$, considering the size of the graph is $4k + 2$. It can be ensured that if the algorithm outputs a non-negative result, it is indeed correct as the only problem occurs when the specific type of cycle as mentioned passes through every vertex

References

1. Bjorklund, Andreas. "Determinant sums for undirected hamiltonicity." SIAM Journal on Computing 43, no. 1 (2014): 280-299.
2. Björklund, Andreas. "Exact covers via determinants." arXiv preprint arXiv:0910.0460 (2009).
3. Yates, F., 1938. The design and analysis of factorial experiments.
4. Coppersmith, Don, and Shmuel Winograd. "Matrix multiplication via arithmetic progressions." In Proceedings of the nineteenth annual ACM symposium on Theory of computing, pp. 1-6. 1987.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.