Counting Partitions of Graphs

Pavol Hell [1]   Miki Hermann [2]   Mayssam Mohammadi Nevisi [1]

[1]Simon Fraser University, Burnaby, Canada

[2]École Polytechnique, Palaiseau, France

June 11, 2013

CanaDAM 2013

Memorial University of Newfoundland

Outline

1. Introduction
   - Graph Homomorphisms and Counting
   - Matrix Partition Problem for Graphs
2. The Dichotomy
3. A Warm-up: Two by two matrices
4. Polynomial Algorithms
   - Sparse-Dense Partitions
   - Homogeneous Sets and Modular Decomposition
   - Counting by Components
   - One Special Case
5. Hardness Results
6. Future Directions

Counting Homomorphisms of Graphs

Let H be a fixed graph. The problem of *counting homomorphisms* to H, denoted $\#\mathrm{H}$, asks for the number of homomorphisms of an input graph G to graph H.

Counting Homomorphisms of Graphs

Let H be a fixed graph. The problem of *counting homomorphisms* to H, denoted $\#\mathrm{H}$, asks for the number of homomorphisms of an input graph G to graph H.

Theorem — M. Dyer, C. Greenhill (1999)

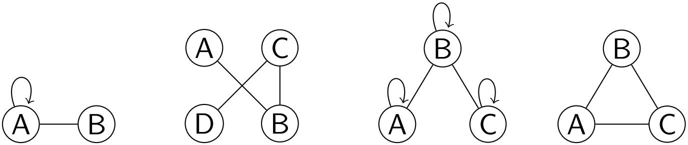
The problem $\#\mathrm{H}$ is $\#\mathrm{P}$-complete if H has a connected component which is not a complete reflexive graph or a complete bipartite irreflexive graph. Otherwise, the problem is in FP.

Counting Homomorphisms of Graphs

Equivalently, $\#\mathrm{H}$ is in *FP* when *H* does not contain any of the followings as an induced subgraph, and is $\#\mathrm{P}$-complete otherwise.

- A looped vertex adjacent to a non-looped vertex
- An irreflexive P_4
- A reflexive P_3
- Triangle

Counting Homomorphisms of Graphs



$$\begin{pmatrix} 1 & 1 \\ & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 0 & 0 \\ & 0 & 1 & 0 \\ & & 0 & 1 \\ & & & 0 \end{pmatrix} \quad \begin{pmatrix} 1 & 1 & 0 \\ & 1 & 1 \\ & & 1 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 1 \\ & 0 & 1 \\ & & 0 \end{pmatrix}$$

Counting Homomorphisms of Graphs

Theorem — M. Dyer, C. Greenhill (1999)

Let H be a graph such that $\#\mathrm{H}(H)$ is $\#\mathrm{P}$-complete. Then there exists a constant Δ such that $\#\mathrm{H}(H)$ remains $\#\mathrm{P}$-complete when restricted to instances with maximum degree at most Δ.

Counting List Homomorphisms of Graphs

Let H be a fixed graph. The problem of *counting list homomorphisms* to H, denoted $\#\mathrm{LHOM}$, asks for the number of list homomorphisms of an input graph G with respect to the lists $L(v) \subset V(H)$, to H.

Counting List Homomorphisms of Graphs

Let H be a fixed graph. The problem of *counting list homomorphisms* to H, denoted $\#\mathrm{LHOM}$, asks for the number of list homomorphisms of an input graph G with respect to the lists $L(v) \subset V(H)$, to H.

Theorem — **Pavol H., J. Nešetřil (2001)**

$\#\mathrm{LHOM}(H)$ is polynomial time solvable when each component of H is either a reflexive complete graph or an irreflexive complete bipartite graph, and is $\#\mathrm{P}$-complete otherwise.

Graph M-Partition

Let M be a fixed symmetric matrix of size k over $\{0, 1, *\}$.

An *M-partition* of a graph G is a partition $P_1, P_2, \cdots, P_k$ of $V(G)$ such that for any two distinct vertices u, v in (not necessarily different) parts P_i, P_j are adjacent when $M_{i,j} = 1$ and are not adjacent when $M_{i,j} = 0$. $M_{i,j} = *$ does not enforce any restriction.

The *graph partition problem* for matrix M (or simply the *M-partition problem*), asks whether an input graph G admits an M-partition.

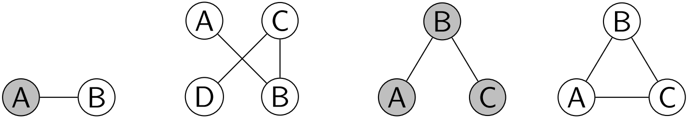
Graph partition problems generalize both graph colouring and graph homomorphisms.

Graph M-Partition

- The diagonal values define the constraints on the parts:
  - P_i must induce an independent set whenever $M_{i,i} = 0$
  - P_i must induce a clique whenever $M_{i,i} = 1$
- The off-diagonal values define the constraints among different parts:
  - $M_{i,j} = 1$ means every vertex in P_i must be adjacent to every vertex in P_j in G in any M-partition
  - $M_{i,j} = 0$ means no vertex in P_i might be adjacent to a vertex in P_j in G in any M-partition

Graph M-Partition

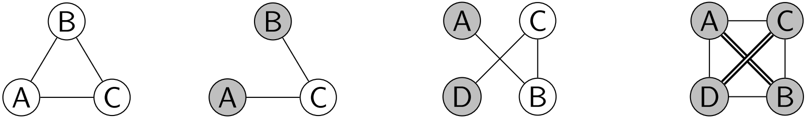
$$\begin{pmatrix} \ast & \ast \\ & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & \ast & 0 & 0 \\ & 0 & \ast & 0 \\ & & 0 & \ast \\ & & & 0 \end{pmatrix} \quad \begin{pmatrix} \ast & \ast & 0 \\ & \ast & \ast \\ & & \ast \end{pmatrix} \quad \begin{pmatrix} 0 & \ast & \ast \\ & 0 & \ast \\ & & 0 \end{pmatrix}$$



Graph M-Partition

The matrices for 3-colouring, stable cutset, stable cutset pair, and $2K_2$ partition.

$$\begin{pmatrix} 0 & * & * \\ & 0 & * \\ & & 0 \end{pmatrix} \quad \begin{pmatrix} * & 0 & * \\ & * & * \\ & & 0 \end{pmatrix} \quad \begin{pmatrix} * & * & 0 & 0 \\ & 0 & * & 0 \\ & & 0 & * \\ & & & * \end{pmatrix} \quad \begin{pmatrix} * & 1 & * & * \\ & * & * & * \\ & & * & 1 \\ & & & * \end{pmatrix}$$



Graph M-Partition

- When there is an $*$ on the diagonal, then the graph M-partition problem is trivial.

Graph M-Partition

- When there is an $*$ on the diagonal, then the graph M-partition problem is trivial.

Theorem (Feder et al. (2003))

Suppose $k = 4$. The M-partition problem is NP-complete when M contains the matrix of 3-colouring or its complement, and is polynomial time solvable otherwise.

Counting Partitions of Graphs

Theorem

Suppose M is an m by m matrix with $m < 4$, and assume that M contains both a 0 and a 1.

If M contains, as a principal submatrix, the matrix for independent sets, or the matrix for cliques, then the counting problem for M-partitions is $\#\mathrm{P}$-complete.

Otherwise, counting M-partitions is polynomial.

Warm-Up: Two by Two Matrices

Theorem — J.S. Provan and M.O. Ball (1983)

The problem of counting number of independent sets in graphs, denoted $\#\mathrm{I}(G)$, is $\#\mathrm{P}$-complete.

Warm-Up: Two by Two Matrices

Theorem J.S. Provan and M.O. Ball (1983)

The problem of counting number of independent sets in graphs, denoted $\#\mathrm{I}(G)$, is $\#\mathrm{P}$-complete.

Corollary

The problem of counting number of cliques in graphs, denoted $\#\mathrm{K}(G)$, is $\#\mathrm{P}$-complete.

Warm-Up: Two by Two Matrices

Theorem J.S. Provan and M.O. Ball (1983)

The problem of counting number of independent sets in graphs, denoted $\#\mathrm{I}(G)$, is $\#\mathrm{P}$-complete.

Corollary

The problem of counting number of cliques in graphs, denoted $\#\mathrm{K}(G)$, is $\#\mathrm{P}$-complete.

In all other cases, counting is trivial or easy.

Warm-Up: Two by Two Matrices

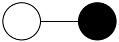
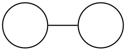
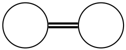
Theorem — J.S. Provan and M.O. Ball (1983)

The problem of counting number of independent sets in graphs, denoted $\#\mathrm{I}(G)$, is $\#\mathrm{P}$-complete.

Corollary

The problem of counting number of cliques in graphs, denoted $\#\mathrm{K}(G)$, is $\#\mathrm{P}$-complete.

In all other cases, counting is trivial or easy.

$\begin{pmatrix}0 & * \\ & 1\end{pmatrix}$  $\begin{pmatrix}0 & * \\ & 0\end{pmatrix}$  $\begin{pmatrix}0 & 1 \\ & 0\end{pmatrix}$ 

Polynomial Algorithms

- Algorithms based on sparse-dense partitions
- Algorithms based on modular decomposition
- Algorithms based on connected components
- Pre-colouring extension and one special case

Sparse-Dense Partitions

Sparse-Dense Partitions

If $\mathcal{S}$ and $\mathcal{D}$ are two families of subsets of $V(G)$, and if there exists a constant t such that all intersections $S \cap D$ with $S \in \mathcal{S}, D \in \mathcal{D}$, have at most t vertices, then G with n vertices has at most n^{2t} sparse-dense partitions $V(G) = S \cup D$, with $S \in \mathcal{S}, D \in \mathcal{D}$, and they can be generated in polynomial time.

Motwani et. al (2003)

Sparse-Dense Partitions

Sparse-Dense Partitions

If $\mathcal{S}$ and $\mathcal{D}$ are two families of subsets of $V(G)$, and if there exists a constant t such that all intersections $S \cap D$ with $S \in \mathcal{S}, D \in \mathcal{D}$, have at most t vertices, then G with n vertices has at most n^{2t} sparse-dense partitions $V(G) = S \cup D$, with $S \in \mathcal{S}, D \in \mathcal{D}$, and they can be generated in polynomial time.

Motwani et. al (2003)

- For split partitions, take $\mathcal{S}$ to be all independent sets, $\mathcal{D}$ all cliques ($t = 1$)

Sparse-Dense Partitions

Let $M = \begin{pmatrix} a & d & e \\ & b & f \\ & & c \end{pmatrix}$.

Theorem

If a, b, c are not all the same and none is $*$, then the number of M-partitions can be counted in polynomial time.

Sparse-Dense Partitions

Let $M = \begin{pmatrix} a & d & e \\ & b & f \\ & & c \end{pmatrix}$.

Theorem

If a, b, c are not all the same and none is $*$, then the number of M-partitions can be counted in polynomial time.

- By symmetry and complementarity, we may assume that $a = b = 0$ and $c = 1$
- Take for $\mathcal{S}$ all bipartite induced subgraphs, and for $\mathcal{D}$ all cliques ($t = 2$)

Modular Decomposition

Definition

A *module* (or a *homogeneous set*) in a graph G is a set $S \subseteq V(G)$ such that every vertex not in S is either adjacent to all vertices of S or to none of them.

Trivial examples include:

- The empty set
- The entire vertex set of G
- Every singleton vertex

Gallai's Theorem

Theorem — Gallai (1967)

For any graph G one of the following three cases must occur:

1. G is disconnected, with components $G_1, G_2, \dots G_k$.
   Each union of the sets $V(G_i)$ is a module of G, and the other modules of G are precisely all the modules of individual components G_i.
2. The $\overline{G}$ is disconnected, with components $H_1, H_2, \dots, H_\ell$.
   Each union of the sets $V(H_j)$ is a module of G, and the other modules of G are precisely all the modules of individual subgraphs $\overline{H_j}$.
3. Both G and $\overline{G}$ connected. There is a partition $S_1, S_2, \dots, S_r$ of $V(G)$ (computable in linear time), such that all the modules of G are precisely all the modules of individual subgraphs induced by the sets S_t, $t = 1, \dots, r$, plus the module $V(G)$.

Modular Decomposition

Theorem

There is a polynomial time algorithm to count the number of modules satisfying any combination of restrictions where the module itself, its set of neighbours, and its set of non-neighbours are an independent set, a clique, or unrestricted.

Modular Decomposition

Theorem

There is a polynomial time algorithm to count the number of modules satisfying any combination of restrictions where the module itself, its set of neighbours, and its set of non-neighbours are an independent set, a clique, or unrestricted.

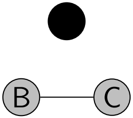
Theorem

If d, e, f are all different, and M does not contain, as a principal submatrix, the matrix for independent sets or cliques, then the number of M-partitions can be counted in polynomial time.

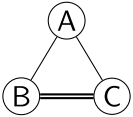
Counting via Components

Theorem

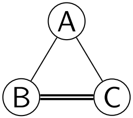
Assume that two of d, e, f are 0 and M does not contain as principal submatrix the matrix for independent sets or cliques. Then counting the number of M-partitions is polynomial.



One Special Case



One Special Case



When the input graph G is not bipartite:

- If G is triangle-free, then the answer is 0.
- A triangle in G must use all three colours, in six possible ways. And any of these six partial colourings can be extended uniquely until an induced bipartite sub-graph of G remains unassigned.

One Special Case

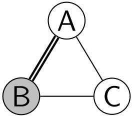
And if $G = (X, Y)$ is bipartite, then there is a recursive polynomial-time algorithm:

One Special Case

And if $G = (X, Y)$ is bipartite, then there is a recursive polynomial-time algorithm:

- If G is $2K_2$ free, then it has a (bipartite) universal vertex (J.P. Liu and H.S. Zhou, 1997).
- Partial proper colourings of a $2K_2$ can be extended nicely (until all remaining vertices are disjoint from partial set A and fully adjacent to either B or C)
- There are exactly two ways to colour a $2K_2$, and the sub-graphs remained after extending each of them are disjoint.
- Partial proper colourings of a universal vertex u can be extended nicely (It will either enforce the colour of the opposite part completely, or nothing)

Hardness Proof Technique



Let G^* be the graph obtained from G by adding a universal vertex u to G. We have:

$$\#\mathrm{M}(G^*) = \#\mathrm{I}(G) + \#\mathrm{M}(G) + \#\mathrm{AB}(G)$$

$$(u \in A) \quad (u \in B) \quad (u \in C)$$

Open Questions

- Is there a dichotomy when $k \geq 4$?
- Are there a finite set of forbidden structures (principal sub-matrices) that make the problem hard?

Thank you