Available online at www.sciencedirect.com
SCIENCE DIRECT®
Theoretical Computer Science 348 (2005) 148–186
Theoretical Computer Science
www.elsevier.com/locate/tcs

The complexity of partition functions

Andrei Bulatov[a], Martin Grohe[b,*]

[a] *School of Computing Science, Simon Fraser University, Burnaby, Canada*
[b] *Institut für Informatik, Humboldt-Universität, Unter den Linden 6, 10099 Berlin, Germany*

Abstract

We give a complexity theoretic classification of the counting versions of so-called H-colouring problems for graphs H that may have multiple edges between the same pair of vertices. More generally, we study the problem of computing a weighted sum of homomorphisms to a weighted graph H.

The problem has two interesting alternative formulations: first, it is equivalent to computing the partition function of a spin system as studied in statistical physics. And second, it is equivalent to counting the solutions to a constraint satisfaction problem whose constraint language consists of two equivalence relations.

In a nutshell, our result says that the problem is in polynomial time if the adjacency matrix of H has row rank 1, and #P-hard otherwise.

Keywords: Counting complexity; Partition function; Graph homomorphism; Constraint satisfaction

1. Introduction

This paper has two different motivations: the first is concerned with constraint satisfaction problems, the second with "spin-systems" as studied in statistical physics. A known link between the two are so-called H-colouring problems. Our main result is a complete complexity theoretic classification of the problem of counting the number of solutions of an H-colouring problem for an undirected graph H which may have multiple edges, and actually of a natural generalisation of this problem to weighted graphs H. Translated to the world of constraint satisfaction problems, this yields a classification of the problem of counting the solutions to constraint satisfaction problems for two equivalence relations. Translated to the world of statistical physics, it gives a classification of the problem of computing the partition function of a spin system.

Let us describe our result from each of the different perspectives: Let H be a graph, possibly with multiple edges between the same pair of vertices, e.g. a *multi-graph*. An *H-colouring* of a graph G is a homomorphism from G to H. Both the decision problem, asking whether a given graph has an H-colouring, and the problem of counting the H-colourings of a given graph, have received considerable attention [5,6,9,11,12]. Here we are interested in the counting problem. Dyer and Greenhill [5] gave a complete complexity theoretic classification of the counting problem for undirected graphs H without multiple edges; they showed that the problem is in polynomial time if each connected component of

* Corresponding author.
E-mail addresses: abulatov@cs.sfu.ca (A. Bulatov), grohe@informatik.hu-berlin.de (M. Grohe).

0304-3975/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.tcs.2005.09.011

H is complete bipartite without any loops or is complete with all loops present, and #P-hard otherwise. Here we are interested in counting H-colourings for multi-graphs H. Note that, as opposed to the decision problem, multiple edges do make a difference for the counting problem. Let H be a multi-graph with vertex set $\{1, \ldots, k\}$. H is best described in terms of its adjacency matrix $A = (A_{ij})$, where A_{ij} is the number of edges between vertices i and j. Given a graph $G = (V, E)$, we want to compute the number of homomorphisms from G to H. Observe that this number is

$$Z_A(G) = \sum_{\sigma: V \to \{1,\ldots,k\}} \prod_{e=\{u,v\} \in E} A_{\sigma(u)\sigma(v)}. \tag{1}$$

Borrowing from the physics terminology, we call Z_A the *partition function* of A (or H). We denote the problem of computing $Z_A(G)$ for a given graph G by EVAL(A). Of course if we define Z_A as in (1), the problem is not only meaningful for matrices A that are adjacency matrices of multi-graphs, but for arbitrary square matrices A. We may view such matrices as adjacency matrices of weighted graphs (omitting edges of weight 0). We call a symmetric matrix A *connected* (*bipartite*) if the corresponding graph is connected (*bipartite*, respectively).

We prove the following classification result:

Theorem 1. *Let A be a symmetric matrix with non-negative real entries.*

(1) *If A is connected and not bipartite*, *then* EVAL(A) *is in polynomial time if the row rank of A is at most* 1; *otherwise* EVAL(A) *is* #P-*hard*.
(2) *If A is connected and bipartite*, *then* EVAL(A) *is in polynomial time if the row rank of A is at most* 2; *otherwise* EVAL(A) *is* #P-*hard*.
(3) *If A is not connected*, *then* EVAL(A) *is in polynomial time if each of its connected components satisfies the corresponding condition stated in* (1) *or* (2); *otherwise* EVAL(A) *is* #P-*hard*.

Note that this generalises Dyer and Greenhill's [5] classification result for graphs without multiple edges, whose adjacency matrices are symmetric 0-1 matrices.

Our proof builds on interpolation techniques similar to those used by Dyer and Greenhill, recent results on counting the number of solutions to constraint satisfaction problems due to Dalmau and the first author [1], and a considerable amount of polynomial arithmetic. Even though we present the proof in the language of constraint satisfaction problems here, in finding the proof it has been very useful to jump back and forth between the H-colouring and constraint satisfaction perspective.

Let us now explain the result for constraint satisfaction problems. A *constraint language* Γ on a finite *domain* D is a set of relations on D. An instance of the problem CSP(Γ) is a triple $(V, D, \mathcal{C})$ consisting of a set V of *variables*, the domain D, and a set $\mathcal{C}$ of *constraints* $\langle s, \rho \rangle$, where ρ is a relation in Γ and s is a tuple of variables whose length matches the arity of ρ. A *solution* is a mapping $\sigma : V \to D$ such that for each constraint $\langle (v_1, \ldots, v_r), \rho \rangle \in \mathcal{C}$ we have $(\sigma(v_1), \ldots, \sigma(v_r)) \in \rho$. There has been considerable interest in the complexity of constraint satisfaction problems [16,14,7,2,3], which has mainly been driven by Feder and Vardi's [7] *dichotomy question*, asking whether for all languages Γ the problem CSP(Γ) is either solvable in polynomial time or NP-complete. A similar dichotomy question can be asked for the problem #CSP(Γ) of counting the solutions for a given instance [4,1].

We consider constraint languages Γ consisting of two equivalence relations α, β. Suppose that α has k equivalence classes and β has ℓ equivalence classes. Then Γ can be described by a $(k \times \ell)$-matrix $B = (B_{ij})$, where B_{ij} is the number of elements in the intersection of the ith class of α and the jth class of β. We show that, provided that the matrix is "indecomposable" (in a sense made precise in Section 2.2), the problem #CSP(Γ) is in polynomial time if the row rank of B is 1 and #P-hard otherwise. There is also a straightforward extension to "decomposable" matrices (see Corollary 15 for the precise statement). In [1], it has been shown that if #CSP(Γ) is in polynomial time, then Γ has a so-called *Mal'tsev polymorphism*. The result of this paper provides a further necessary condition for Γ to give rise to a counting problem solvable in polynomial time.

We can generalise our result for CSP whose language consists of two equivalence relations to *weighted CSP*, where each domain element d carries a non-negative real weight $\omega(d)$. The weight of a solution $\sigma : V \to D$ is defined to be the product $\prod_{v \in V} \omega(\sigma(v))$, and the goal is to compute the weighted sum over all solutions (see Theorem 14 for the precise statement of our result). As an important intermediate step, we even prove our classification result for weights that are polynomials with integer coefficients.