Tsinghua Science and Technology

Volume 19 | Issue 4 Article 8

2014

Counting Problems in Parameterized Complexity

Chihao Zhang
Department of Computer Science and Engineering, Shanghai Jiao Tong University, No. 800 Dongchuan Road, Shanghai 200240, China.

Yijia Chen
Department of Computer Science and Engineering, Shanghai Jiao Tong University, No. 800 Dongchuan Road, Shanghai 200240, China.

TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214 08/08 pp410-420
Volume 19, Number 4, August 2014

Counting Problems in Parameterized Complexity

Chihao Zhang and Yijia Chen*

Abstract: Parameterized complexity is a multivariate theory for the analysis of computational problems. It leads to practically efficient algorithms for many **NP**-hard problems and also provides a much finer complexity classification for other intractable problems. Although the theory is mostly on decision problems, parameterized complexity naturally extends to counting problems as well. The purpose of this article is to survey a few aspects of parameterized counting complexity, with a particular emphasis on some general frameworks in which parameterized complexity proves to be indispensable.

Key words: parameterized complexity; counting problems; dichotomy theorems

1 Introduction

Counting problems are well studied in classical complexity theory. Valiant[1] introduced the central class #**P** in 1970s as the counting version of **NP**, and classified counting problems in term of their computational complexity has become a very active research area ever since. Recently, many *dichotomy theorems* were obtained for various counting frameworks, in both exact counting and approximate counting settings (see Ref. [2] for a recent survey). These are the results showing that a large number of naturally defined problems are either #**P**-hard or solvable in polynomial time (i.e., in the class **FP**). For example, Dyer and Greenhill[3] proved that for any fixed graph H counting the homomorphisms from a graph G to H is either in **FP** or #**P**-complete. Remember that we do have problems whose hardness forms an infinite hierarchy within #**P**, an easy adaptation of Ladner's Theorem[4] for the counting complexity. Sometimes, counting problems behave very differently compared to their decision counterparts. There are hard counting problems whose underlying decision problems are easy, e.g., counting perfect matchings in bipartite graphs[5]. Furthermore, a celebrated result due to Toda[6] shows that every decision problem in the polynomial hierarchy can be reduced to #**P**. These results reveal an inherent difference between counting problems and decision problems.

Parameterized complexity[7-10] provides a two-dimensional view of computational problems in which every problem instance comes with a presumably small *parameter*. The theory was initiated by Downey and Fellows in 1990[11], and has been very successful mainly for decision problems. A comprehensive structural theory has been established around the central notion *fixed-parameter tractability*, which is a relaxation of polynomial-time tractability in classical complexity theory. However, it was not until about 10 years ago, Flum and Grohe[12], and independently McCartin[13], formalized the intractable theory of counting problems in parameterized complexity. In their seminal papers[12, 13], among many other things, a hierarchy of parameterized complexity classes #**W**$[t]$ for $t \geqslant 1$ was defined. It is generally believed that #**W**$[t]$-hard problems do not have fixed-parameter counting algorithms, i.e., algorithms with running time $f(k)\text{poly}(n)$ where n is the size of the input, k the parameter, and f a computable function. And these complexity classes can be viewed as the analogue of #**P**

• Chihao Zhang and Yijia Chen are with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, No. 800 Dongchuan Road, Shanghai 200240, China. E-mail: chihao.zhang@gmail.com and yijia.chen@cs.sjtu.edu.cn.
∗ To whom correspondence should be addressed.
Manuscript received: 2014-06-09; accepted: 2014-06-13

in the parameterized world.

On the algorithmic side, many fixed-parameter counting algorithms are derived directly from their decision versions. For instance, in the vertex cover problem, we want to decide whether in an n-vertex graph G there are k vertices that intersect every edge in G, i.e., a vertex cover of size k. The well-known $O(2^k n^2)$ time bounded search tree algorithm can be easily adapted to count the vertex covers of size k in G. However, just like in classical counting complexity, this is not always the case. A few hard parameterized counting problems have easy decision versions. One notable example is the following parameterized problem of counting matchings.

p-#MATCHING	
Instance:	A graph G and $k \in \mathbb{N}$.
Parameter:	k.
Problem:	Count the k-matchings in G.

Recall that a k-matching is a set of k vertex-disjoint edges. Although one can decide in polynomial time whether a graph contains a matching of size k[14], it has been shown recently[15] that p-#MATCHING is hard for #$\mathbf{W}[1]$, thus an $f(k)\mathrm{poly}(n)$ algorithm to count the number of k-matchings is unlikely to exist.

As one of the main focuses of this survey, we look at some counting frameworks that admit natural parameterizations. The first one is the aforementioned problem of counting graph homomorphisms. Given graphs G and H, the problem is to compute the number of homomorphisms, i.e., edge-preserving mappings, from G to H. Many combinatorial problems can be defined in this framework. For instance, if we fix H to be K_k (i.e., the k-vertex complete graph), then the number of homomorphisms from G to H is exactly the number of k-colorings in G. If we allow G to be an arbitrary graph and that H to be chosen from a family of graphs $\mathcal{C}$, then the computational complexity of the problem is well-understood: A dichotomy theorem due to Bulatov[16] states that the problem is either in **FP** or #**P**-complete, depending on $\mathcal{C}$, and the precise characterization of tractable $\mathcal{C}$ was also given (see also Refs. [17, 18]). On the other hand, we may allow H to be arbitrary and that G to be chosen from a family of graphs $\mathcal{C}$. In this setting, it turns out that a dichotomy theorem in classical complexity like above does not exist[19]. However, for a natural parameterization of the problem, i.e., one takes size $|G|$ of G as the parameter, we can obtain a clear-cut complexity classification. Under the same parameterization, the problem of counting embeddings and strong embeddings also admit such a classification. These dichotomy theorems suggest that parameterized complexity is really necessary for the understanding of those problems.

The treewidth of graph is one of most important parameters for fixed-parameter algorithms on graphs[20]. Very often, by bounding the treewidth of input graphs, **NP**-hard problems become tractable. Courcelle's famous meta-theorem[21] nicely explains many such results from a logic point of view. It asserts that every graph property expressible by Monadic Second-Order logic (MSO) can be decided in linear time on graphs of bounded treewidth. In other words, the model-checking problem for MSO on graphs, parameterized by the formula length and the treewidth, is fixed-parameter tractable. Arnborg et al.[22] generalized Courcelle's Theorem to the counting setting. It is worth noting that a lower bound result for Courcelle's Theorem[23] is also applicable in this counting version, which implies that the treewidth is the parameter of the problem and again parameterized complexity plays a central role towards the full picture of the complexity of the framework.

The Holant problems are another natural class of problems for which taking the treewidth as the parameter results in fixed-parameter algorithms. The problems were introduced in Refs. [24, 25], which stem from Valiant's holographic algorithms[26]. The framework is very expressive and includes counting graph homomorphisms and counting Constraint Satisfaction Problems (CSP) as special cases. The study of Holant problems on graphs of bounded treewidth is motivated by approximate counting Holant problems on planar graphs[27]. Unlike Courcelle's Theorem, the dependence on the treewidth in the running time of the algorithm is crucial to those approximation algorithms. Moreover, the so-called holographic reductions[26] are also applicable in this setting. Using this reduction, we can design algorithm for some problems that a direct dynamic programming on tree decompositions does not seem to apply. Thus, it is reasonable to expect that holographic reductions can help to deepen our understanding for the power of treewidth.

As many counting problems are proved to be intractable, we might take one step back and look for their *approximate solutions*, i.e., fixed-parameter algorithms that output a number within a limited range of the desired solution. As a matter of fact, many parameterized intractable counting problems, including p-#MATCHING, admit randomized approximation scheme[28]. These results refine the dichotomy theorems established in terms of exact counting for some frameworks mentioned above.

Organization

We start by introducing some key notions and parameterized counting classes in Section 2. A few important examples are then given in Section 3, including those problems exhibiting an interesting division between easy-decision and hard-counting. Section 4 is devoted to the role of the treewidth in parameterized counting. We put a particular emphasis on the Holant problems, since they are still relatively unknown in the parameterized complexity community, despite their continuing success in classical counting complexity. In Section 5 we present the dichotomy theorems on parameterized counting of homomorphisms, embeddings, and strong embeddings. The approximate parameterized counting is discussed in Section 6. Finally Section 7 concludes with some open problems.

2 Preliminaries

In this section, we follow Ref. [12] and define the complexity classes $\#\mathbf{W}[t]$ for $t \geqslant 1$.

We assume that the reader is familiar with basic parameterized complexity theory, thus in the following, we mainly focus on notions specific to counting problems.

Let $\{0,1\}$ be the alphabet. A parameterized counting problem is a pair (P,κ), where $P:\{0,1\}^* \to \mathbb{N}$ assigns every string in $\{0,1\}^*$ a non-negative number, and where $\kappa:\{0,1\}^* \to \mathbb{N}$ is a polynomial-time computable function called parameterization. Then (P,κ) is *fixed-parameter tractable* (FPT) if there is an algorithm A that takes as input $x \in \{0,1\}^*$ and computes $P(x)$ within time $f(\kappa(x))|x|^{O(1)}$, where f is an arbitrary computable function. The class **FPT** contains all fixed-parameter tractable counting problems.

We then introduce the notion of *fpt parsimonious reduction* and *fpt Turing reduction*. Let (P_1,κ_1) and (P_2,κ_2) be two parameterized counting problems.

Definition 1 An fpt parsimonious reduction from (P_1,κ_1) to (P_2,κ_2) is an algorithm A such that takes as input $x \in \{0,1\}^*$ and satisfies

(1) $P_1(x) = P_2(A(x))$ and $\kappa_2(A(x)) = f(\kappa_1(x))$ for some computable function f;

(2) $A(x)$ terminates in $h(\kappa_1(x))|x|^{O(1)}$ time for some computable function h.

In this case, we write $(P_1,\kappa_1) \leqslant^{\text{fpt}} (P_2,\kappa_2)$.

Definition 2 An fpt Turing reduction from (P_1,κ_1) to (P_2,κ_2) is an algorithm A with an oracle to P_2 such that takes as input $x \in \{0,1\}^*$ and satisfies that

(1) $A(x) = P(x)$;

(2) There exists a computable function f such that for every oracle call $P_2(y)$, it holds $\kappa_2(y) \leqslant f(\kappa_1(x))$;

(3) $A(x)$ terminates in time $h(\kappa_1(x))|x|^{O(1)}$ for some computable function h.

In this case, we write $(P_1,\kappa_1) \leqslant^{\text{fpt-T}} (P_2,\kappa_2)$.

Let $\mathcal{A}$ be a family of parameterized problems, we denote $[\mathcal{A}]^{\text{fpt}}$ the closure of $\mathcal{A}$ under fpt parsimonious reductions. Formally,

$$[\mathcal{A}]^{\text{fpt}} \triangleq \left\{(Q,\kappa) \;\middle|\; \exists (Q',\kappa') \in \mathcal{A},\ (Q,\kappa) \leqslant^{\text{fpt}} (Q',\kappa')\right\}.$$

For every $t \geqslant 0$ and $d \geqslant 1$, we inductively define $\Gamma_{t,d}$ and $\Delta_{t,d}$ as subclasses of propositional logic formulae.

$$\Gamma_{0,d} \triangleq \left\{ \bigwedge_{i=1}^{k} \lambda_i \;\middle|\; 1 \leqslant k \leqslant d \right\},$$

where each λ_i is a literal.

$$\Delta_{0,d} \triangleq \left\{ \bigvee_{i=1}^{k} \lambda_i \;\middle|\; 1 \leqslant k \leqslant d \right\},$$

where each λ_i is a literal.

$$\Gamma_{t+1,d} \triangleq \left\{ \bigwedge_{i=1}^{k} \delta_i \;\middle|\; k \geqslant 1 \right\},$$

where each $\delta_i \in \Delta_{t,d}$.

$$\Delta_{t+1,d} \triangleq \left\{ \bigvee_{i=1}^{k} \gamma_i \;\middle|\; k \geqslant 1 \right\},$$

where each $\gamma_i \in \Gamma_{t,d}$. Note $\Gamma_{1,d}$ is precisely the set of dCNF formulae and $\Delta_{1,d}$ the set of dDNF formulae. Then for every $t,d \geqslant 1$, we define the parameterized weighted satisfiability problem:

p-#WSAT($\Gamma_{t,d}$)

Instance:	A propositional formula $\varphi \in \Gamma_{t,d}$ and $k \in \mathbb{N}$.
Parameter:	k.
Problem:	Compute the number of assignments with exactly k variable set to TRUE that satisfy φ.

Definition 3 For every $t \geqslant 1$, #**W**$[t]$ is the family of parameterized counting problems that are fpt parsimonious reducible to p-#WSAT$(\Gamma_{t,d})$ for some $d \geqslant 1$, i.e., #**W**$[t] \triangleq [\{p\text{-\#WSAT}(\Gamma_{t,d}) | d \geqslant 1\}]^{\text{fpt}}$. (The reader is encouraged to define #**W**[SAT] and #**W**[P] corresponding to the decision classes **W**[SAT] and **W**[P].)

Clearly, #**W**$[t]$ is the counting version of the class **W**$[t]$. By standard means, it is not very difficult to show:

Theorem 1[12,13] p-#WSAT$(\Gamma_{1,2})$ is complete for #**W**[1], and p-#WSAT$(\Gamma_{t,2})$ is complete for #**W**$[t]$ for every $t \geqslant 2$.

The following result provides some evidence that #**W**$[t]$-hard problems are unlikely to be fixed-parameter tractable.

Theorem 2[12] If #**W**[1] $\subseteq$ **FPT**, then there is an algorithm which counts the satisfying assignments for a 3CNF formula with n variables in time $2^{o(n)}$, i.e., the counting 3SAT is solvable in sub-exponential time.

3 Some Intractable Counting Problems

As one can expect, there is a wealth of natural intractable parameterized counting problems, most of which are the counting versions of the corresponding intractable decision problems. We give a few important examples.

p-#SHORTTURINGMACHINEACCEPTANCE	
Instance:	A non-deterministic Turing machine $\mathbb{M}$ and $k \in \mathbb{N}$.
Parameter:	k.
Problem:	Count the accepting runs of $\mathbb{M}$ of length at most k on the empty input.

p-#CLIQUE	
Instance:	A graph G and $k \in \mathbb{N}$.
Parameter:	k.
Problem:	Count the cliques of size k in G.

p-#DOMINATINGSET	
Instance:	A graph G and $k \in \mathbb{N}$.
Parameter:	k.
Problem:	Count the dominating sets of size k in G.

Here, a graph $G = (V, E)$ is always undirected, simple, and without self-loops. Often, we use $V(G) \triangleq V$ to denote the vertex set of G and $E(G) \triangleq E$ the edge set. A subset $S \subseteq V$ is a clique if for every distinct $u, v \in S$ we have $\{u, v\} \in E$, while S is a dominating set if for every $u \in V$ either $u \in S$ or there is a vertex $v \in S$ with $\{u, v\} \in E$.

Theorem 3[12,13] The problems p-#SHORTTURINGMACHINEACCEPTANCE and p-#CLIQUE are both #**W**[1]-complete. p-#DOMINATINGSET is #**W**[2]-complete.

The above theorem can be proved along the line of the hardness proof for the decision versions.

In 1979 Valiant[5] showed that counting perfect matchings in bipartite graphs is #**P**-hard. This came as a surprise, since finding one perfect matching was long known to be solvable in polynomial time. Such phenomena exist in parameterized setting as well, among which the first one is the following result due to Flum and Grohe[12].

p-#PATH	
Instance:	A graph G and $k \in \mathbb{N}$.
Parameter:	k.
Problem:	Count the paths of length k in G.

p-#CYCLE	
Instance:	A graph G and $k \in \mathbb{N}$.
Parameter:	k.
Problem:	Count the cycles of length k in G.

Theorem 4[12] p-#PATH and p-#CYCLE are #**W**[1]-complete under fpt Turing reductions.

Since both p-PATH (i.e., finding a path of length k in G) and p-CYCLE (i.e., finding a cycle of length k in G) are fixed-parameter tractable[29,30], it is easy to see that we cannot prove the above hardness under fpt parsimonious reductions, unless it is fixed-parameter tractable to decide whether a graph G contains a k-clique, with k being the parameter.

Like many easy-decision-hard-counting problems, the proof of Theorem 4 is ingenious and apparently more difficult than proving the tractability of the decision problem. For another problem, however, the converse turns out to be true. That is, showing that the decision problem is tractable was much more demanding than establishing the hardness for the counting version.

p-EDGEINDUCEDSUBGRAPH	
Instance:	A graph G and $k \in \mathbb{N}$.
Parameter:	k.
Problem:	Decide whether G has an induced subgraph containing exactly k edges.

p-#EDGEINDUCEDSUBGRAPH	
Instance:	A graph G and $k \in \mathbb{N}$.
Parameter:	k.
Problem:	Count the induced subgraphs in G containing exactly k edges.

Recall that H is an induced subgraph of G if $V(H) \subseteq V(G)$ and

$$E(H) = \{\{u, v\} \in E(G) \mid u, v \in V(H)\}.$$

Theorem 5[31] p-EDGEINDUCEDSUBGRA-PH is fixed-parameter tractable, while p-#EDGEINDU-CEDSUBGRAPH is #**W**[1]-complete under fpt Turing reductions.

Given Valiant's result, one would reasonably guess that counting matchings of size k in bipartite graphs is #**W**[1]-hard. For counting k-matchings in general graphs, it was stated in Ref. [12] as a major open problem in the field. This was only settled very recently by Curticapean[15] based on some previous work[32]. The case for bipartite graphs is announced by Curticapean and Marx[33], when we are writing this survey. We will discuss this problem in more details in Section 5, as it fits into a broader context of embedding problems.

4 Parameterization by Treewidth

Treewidth measures how similar a graph is to a tree. The definition of treewidth and tree decomposition can be found in, e.g., Ref. [9, Chapter 11] and we omit them here. In the following, we use $tw(G)$ to denote the treewidth of a graph $G(V, E)$.

Interestingly, for many computational problems, the notion of treewidth provides the precise borderline between hard and easy instances[34]. In other words, treewidth *captures* tractability. In this section, we survey two frameworks for parameterized counting problems, in which such phenomena also occur.

4.1 Courcelle's theorem

Many intractable problems become tractable on graphs of bounded treewidth via dynamic programming on a tree decomposition. A canonical algorithm of this kind is described by Courcelle's Theorem[21], which states that every graph property expressible by MSO can be decided in FPT time where the parameter is the treewidth of the graph. The algorithm applies to counting problems as well.

Theorem 6 (Courcelle's Theorem, Counting Version[22]) The following problem is fixed-parameter tractable.

Input:	A graph G and an MSO-formula $\Phi(X_1, X_2, \cdots, X_s)$.
Parameter:	$\lvert\Phi\rvert + tw(G)$.
Problem:	Compute the size of the set $\{(A_1, A_2, \cdots, A_s) \mid G \models \Phi(A_1, A_2, \cdots, A_s)\}$.

For example, the formula $\Phi(X_1, X_2, \cdots, X_k)$ below expresses the k-colorability of a graph $G(V, E)$:

$$\Phi(X_1, X_2, \cdots, X_k) \triangleq \forall x\left(\bigvee_{i=1}^{k}\left(x \in X_i \wedge \bigwedge_{j \neq i} x \notin X_j\right)\right) \wedge$$
$$\forall x, y\left(E(x, y) \rightarrow \neg\left(\bigvee_{i=1}^{k}(x \in X_i \wedge y \in X_i)\right)\right).$$

Thus counting the number of k-colorings of a graph G is fixed-parameter tractable parameterized by $k + tw(G)$. In fact, many natural problems can be defined in MSO. As a consequence, Theorem 6 encompasses a large number of FPT results, although it does not always lead to the best fixed-parameter algorithms.

In Ref. [23] Kreutzer and Tazari gave a lower bound for the complexity of model-checking MSO with respect to graphs of poly-logarithmic treewidth. It naturally translates to the counting version, which shows Theorem 6 is rather close to the best we can achieve. As the precise statement of the result is technically involved, we refer the reader to Ref. [23] for details.

We also remark that Frick generalized Theorem 6 to first-order logic with respect to graphs of bounded *local treewidth*[35], which is again built on the corresponding result for the decision problem[36].

4.2 Holant problems

Holant problem is a framework for counting problems introduced in Refs. [24, 25] in the context of studying the power of holographic algorithms. Many other counting frameworks, including counting graph homomorphisms and counting CSP, are special cases of Holant problems, see Ref. [37] for a survey on the relation between these frameworks.

A Holant problem is characterized by a family of vertex functions (or called signatures) $\mathcal{F}$. We denote it by HOLANT $(\mathcal{F})$. Let G be a graph, for every $v \in V(G)$, we use $N(v)$ to denote the set of edges incident to v. The HOLANT $(\mathcal{F})$ is formally defined as

> HOLANT $(\mathcal{F})$
> *Input:* $(G, \{f_v\}_{v\in V(G)})$ where $f_v : \{0,1\}^{N(v)} \to \mathbb{R} \in \mathcal{F}$ for every $v \in V(G)$.
> *Problem:* Compute $\mathrm{hol}(G) \triangleq \sum_{\sigma\in\{0,1\}^{E(G)}} \prod_{v\in V(G)} f_v(\sigma|_{N(v)})$, where $\sigma|_N(v)$ is the restriction of σ on $N(v)$.

To simplify the presentation, we consider the family of *symmetric* functions on Boolean domain. That is, every $f \in \mathcal{F}$ is of the form $f : \{0,1\}^d \to \mathbb{R}$ for some non-negative integers d and the function value only depends on the Hamming weight of the input. Thus f can be expressed as $[f_0, f_1, \cdots, f_d]$ where f_i is the value of f on inputs with Hamming weight i.

HOLANT $(\mathcal{F})$ is hard in general, however, we shall define a class of functions, that is easy on graphs of bounded treewidth.

Consider the following problem:

> p^*-tw-HOLANT $(\mathcal{F})$*
> *Instance:* $(G, \{f_v\}_{v\in V(G)})$ where $f_v \in \mathcal{F}$ for every $v \in V(G)$.
> *Parameter:* $tw(G)$.
> *Problem:* Compute $\mathrm{hol}(G) \triangleq \sum_{\sigma\in\{0,1\}^{E(G)}} \prod_{v\in V(G)} f_v(\sigma|_{N(v)})$.

We define the *pinning* operation on symmetric functions, which intuitively means that a part of the input is pinned to some fixed value. Let $k \leqslant d$ and $\sigma \in \{0,1\}^k$, then $\mathrm{PIN}_\sigma(f) : \{0,1\}^{d-k} \to \mathbb{R}$ is a $(d-k)$-ary function such that for every $\pi \in \{0,1\}^{d-k}$,

$$\mathrm{PIN}_\sigma(f)(\pi) = f(\sigma\pi),$$

where $\sigma\pi \in \{0,1\}^d$ is simply the concatenation of σ and π. Since f is symmetric, the operation is well defined.

We then define the notion of *regularity* by limiting the outcomes of pinning.

Definition 4 (Regularity) A symmetric function $f : \{0,1\}^d \to \mathbb{R}$ is called C-regular if for all $0 \leqslant k \leqslant d$,

$$|\{\mathrm{PIN}_\sigma(f)\,|\sigma \in \{0,1\}^k\}| \leqslant C.$$

Moreover, a family of functions $\mathcal{F}$ is C-regular if every $f \in \mathcal{F}$ is C-regular.

Example 1 The following are two examples of regular functions.

(1) Consider $\mathcal{F}_M \triangleq \{f_d \mid d \geqslant 0\}$ where $f_d : \{0,1\}^d \to \{0,1\}$ is a d-ary function such that

$$f_d(\sigma) = \begin{cases} 1, & \text{if } \|\sigma\| = 1; \\ 0, & \text{otherwise.} \end{cases}$$

Since f_d is symmetric, it can be written as $f_d = [0, 1, \underbrace{0, \cdots, 0}_{d-1\ 0\mathrm{s}}]$. The problem HOLANT $(\mathcal{F})$ is therefore the problem of counting perfect matchings in a graph. It is clear that $\mathcal{F}_M$ is 3-regular.

(2) Our second example is the PARITY problem, that is, $\mathcal{F}_P \triangleq \{f_d^0, f_d^1 | d \geqslant 0\}$ where $f_d^0, f_d^1 : \{0,1\}^d \to \{0,1\}$ are defined as $f_d^0(\sigma) = 1$ if and only if $\|\sigma\|$ is even and $f_d^1(\sigma) = 1$ if and only if $\|\sigma\|$ is odd. Equivalently, $f_d^0 = [1,0,1,0,\cdots]$ and $f_d^1 = [0,1,0,1,\cdots]$. It is easy to see that $\mathcal{F}_P$ is 2-regular.

We remark that the notion of regularity can be easily generalized to functions with larger domain and even to asymmetric functions.

It turns out that regularity implies fixed-parameter tractability on graphs of bounded treewidth.

Theorem 7[27] If $\mathcal{F}$ is C-regular for some constant C, then p^*-tw-HOLANT $(\mathcal{F})$ is fixed-parameter tractable.

The algorithm in the above theorem is based on dynamic programming on tree decompositions. The property of being C-regular is used to bound the size of the table computed in the algorithm.

It is interesting to ask whether the regularity of $\mathcal{F}$ captures the fixed-parameter tractability of p^*-tw-HOLANT (F). However, this is not the case. Consider the following problem. Let $\lambda > 0$ be a positive real. Denote $\mathcal{H} \triangleq \{h_d \mid d \geqslant 0\}$ where $h_d : \{0,1\}^d \to \mathbb{R}$ is given by

$$h_d(\sigma) = \begin{cases} 0, & \text{if } \|\sigma\| \text{ is odd;} \\ \lambda^{\|\sigma\|/2}, & \text{if } \|\sigma\| \text{ is even.} \end{cases}$$

*Here, following Ref. [9], we use the notation p^*-tw to emphasize that the parameter $tw(G)$ only can be computed by a fixed-parameter algorithm, assuming $\mathbf{P} \neq \mathbf{NP}$.

It is clear that $\mathcal{H}$ is not C-regular for any constant C, but it can be reduced to a 3-regular Holant problem in the following way.

Let $G = (V, E)$ be an instance of p^*-tw-HOLANT $(\mathcal{H})$. Define $G' = (V', E')$ as the incidence graph of G. That is, $V' \triangleq V \cup \{v_e \mid e \in E\}$ and $E' \triangleq \{\{u, v_e\}, \{v, v_e\} \mid e = \{u, v\} \in E\}$.

Since each v_e is of degree 2, we assign it a function $[1, 0, \lambda]$. For every other vertex $u \in V'$, we assign a function $f_u = [1, 0, 1, 0, \cdots]$ which is a function that belongs to $\mathcal{F}_P$ we have defined above. Thus, G' is an instance of p^*-tw-HOLANT $(\mathcal{F}_P \cup \{[1, 0, \lambda]\})$ and $\mathcal{F}_P \cup \{[1, 0, \lambda]\}$ is clearly 3-regular. It is not hard to verify that $\mathrm{hol}(G) = \mathrm{hol}(G')$. Moreover, it is well known that $tw(G') = tw(G)$. Thus we have constructed an fpt parsimonious reduction from p^*-tw-HOLANT $(\mathcal{H})$ to p^*-tw-HOLANT $(\mathcal{F}_P \cup \{[1, 0, \lambda]\})$ and it follows from Theorem 7 that both problems are fixed-parameter tractable.

To sum up, we ask the following question:

Question 1 Let $\mathcal{F}$ be a family of symmetric Boolean functions. What is the sufficient and necessary condition under which p^*-tw-HOLANT $(\mathcal{F})$ is fixed-parameter tractable?

As a final remark, we note that the running time of the algorithm[27] is in fact $2^{O(k)}\mathrm{poly}(n)$ where k is the treewidth of the input graph. A related lower bound result was proved in the language of *graphical model*[38] which can be viewed as a special case of Holant problems with regular functions. The result along with a recent established structural theorem for graphs with large treewidth[39] together imply that for some constant $C > 0$, an algorithm with running time $2^{O(k^{1/C})}\mathrm{poly}(n)$ is unlikely to exist. This fact again justifies that the treewidth is the correct parameter here.

5 Counting Homomorphisms, Embeddings, and Strong Embeddings

Ladner proved that there is an **NP**-problem that is neither in **P** nor **NP**-hard[4] if **NP** $\neq$ **P**. His proof can be easily adapted to show that similarly we can construct a counting problem in #**P** $\setminus$ **FP** which is not #**P**-hard. However, for a large number of natural problem classes, e.g., various forms of constraint satisfaction problems, no intermediate problem exists, i.e., every problem in such a class is either in **FP** or #**P**-hard. A very active line of research has been pursuing such dichotomy theorems both for decision and for counting problems (e.g., Ref. [40]), and has seen most of its success in the latter. As we shall see in this section, parameterized counting complexity also offers complete classification for some important graph-theoretic problems, one being the homomorphism problem which is equivalent to certain constraint satisfaction problems. It is remarkable that such dichotomy theorems provably do not transfer to the classical setting.

Let G and H be two graphs and h a mapping from $V(G)$ to $V(H)$.

(1) If $(\{u, v\} \in E(G) \Longrightarrow \{f(u), f(v)\} \in E(H))$ for every $u, v \in V(G)$, then h is a *homomorphism* from G to H.

(2) If h is injective and $(\{u, v\} \in E(G) \Longrightarrow \{f(u), f(v)\} \in E(H))$ for every $u, v \in V(G)$, then h is an *embedding* from G to H.

(3) If h is injective and $(\{u, v\} \in E(G) \Longleftrightarrow \{f(u, f(v)\} \in E(H))$ for every $u, v \in V(G)$, then h is a *strong embedding* from G to H.

Let $k \in \mathbb{N}$. Recall that K_k is the complete graph on k vertices. It is straightforward to verify that for every graph G and a mapping $h : V(K_k) \to V(G)$,

$$h \text{ is a homomorphism from } K_k \text{ to } G \iff$$
$$h \text{ is an embedding from } K_k \text{ to } G \iff$$
$$h \text{ is a strong embedding from } K_k \text{ to } G.$$

As consequences, counting homomorphisms, embeddings, and strong embeddings are all #**P**-complete for general graphs, and the corresponding parameterized problems are #**W**[1]-complete. Therefore, it is then natural to find some appropriate restrictions that make those problems tractable.

Let $\mathcal{C}$ be a class of graphs. Then, the *parameterized counting homomorphism problem on* $\mathcal{C}$ is

p-#HOM($\mathcal{C}$)	
Instance:	Graphs G and H with $G \in \mathcal{C}$.
Parameter:	$\lvert G \rvert$.
Problem:	Count the homomorphisms from G to H.

The *parameterized counting embedding problem on* $\mathcal{C}$ and the *parameterized counting strong embedding problem on* $\mathcal{C}$, denoted by p-#EMBED($\mathcal{C}$) and p-#STRONGEMBED($\mathcal{C}$) respectively, are defined similarly.

Example 2 Let

$$\mathcal{C}_{\text{clique}} \triangleq \{K_k \mid k \in \mathbb{N}\},$$

then p-#HOM($\mathcal{C}_{\text{clique}}$) is essentially the problem p-#CLIQUE, since

the number of the homomorphism from K_k to $G =$
$k! \times$ (the number of the k-cliques in a graph G).

Building on a deep dichotomy theorem of Grohe[41] concerning the decision version, the parameterized complexity of p-#STRONGEMBED($\mathcal{C}$) is now completely settled.

Theorem 8[42] Let $\mathcal{C}$ be an effectively enumerable class of graphs.

(1) If $\mathcal{C}$ is of bounded treewidth, then p-#HOM($\mathcal{C}$) is fixed-parameter tractable. In fact, the underlying classical problem #HOM($\mathcal{C}$) is solvable in polynomial time.
(2) If $\mathcal{C}$ is of unbounded treewidth, then p-#HOM($\mathcal{C}$) is hard for #**W**[1].

Corollary 1 Assuming **FPT** $\neq$ #**W**[1], then the problem #HOM($\mathcal{C}$) is polynomial time computable if and only if $\mathcal{C}$ has bounded treewidth.

Observe that the above corollary gives a criterion of the polynomial time solvability of #HOM($\mathcal{C}$), so it is tempting to replace the assumption **FPT** $\neq$ #**W**[1] by **FP** $\neq$ #**P**. However, this does not seem possible as witnessed by the following result of "Ladner type".

Theorem 9[19] There exists a polynomial time decidable class $\mathcal{C}$ of graphs such that #HOM($\mathcal{C}$) is neither in **FP** nor #**P**-hard.

Next we turn to the strong embedding problems. It turns out that there is a rather straightforward reduction from homomorphisms to strong embeddings. Let G and H be two graphs. Then their *product* $G \times H$ is defined by

$$V(G \times H) \triangleq V(G) \times V(H),$$
$$E(G \times H) \triangleq \{\{(u_1, v_1), (u_2, v_2)\} \mid \{u_1, u_2\} \in E(G) \text{ and } \{v_1, v_2\} \in E(H)\}.$$

One easily verifies that

there is a homomorphism from G to H $\Longleftrightarrow$
there is a strong embedding from G to $G \times H$.

This leads to a proof of the following dichotomy theorem for the strong embedding problems.

Theorem 10[19] Let $\mathcal{C}$ be an effectively enumerable class of graphs.

(1) If $\mathcal{C}$ is finite, then #STRONGEMBED($\mathcal{C}$), the classical problem underlying p-#STRONGEMBED($\mathcal{C}$), is solvable in polynomial time.
(2) If $\mathcal{C}$ is infinite, then p-#STRONGEMBED($\mathcal{C}$) is hard for #**W**[1].

For a long time, a dichotomy theorem for the embedding problems had been elusive. Recall that the matching number of a graph G is the size of a largest matching in G. It is a folklore result that if the matching number of every graph in a class $\mathcal{C}$ is bounded, then the classical #EMBED($\mathcal{C}$) is solvable in polynomial time. Only very recently, the converse is announced by Curticapean and Marx.

Theorem 11[33] Let $\mathcal{C}$ be an effectively enumerable class of graphs of unbounded matching number. Then p-#EMBED($\mathcal{C}$) is hard for #**W**[1].

One main ingredient of the proof of Theorem 11 is a parameterized analogue of the classical #**P**-hardness of counting perfect matchings due to Valiant[5]. Recall the parameterized counting matching problem p-#MATCHING introduced on Section Introduction.

Theorem 12[15] p-#MATCHING is #**W**[1]-complete under fpt Turing reductions.

6 Approximate Counting

Provided the hardness result for p-#PATH, the problem is unlikely to admit FPT algorithm. It is natural to ask for the approximate solution, which is an analogue of the Fully-Polynomial time Randomized Approximation Scheme (FPRAS) for computing permanent in classical setting[43]. In light of this, Arvind and Raman[28] defined the parameterized version of FPRAS, which they called the Fixed-Parameter Tractable Randomized Approximation Scheme (FPTRAS).

Definition 5 (Fixed-parameter tractable randomized approximation scheme). Let (P, κ) be a parameterized counting problem. A randomized algorithm A is a fixed-parameter tractable randomized approximation scheme if it satisfies

(1) It takes as input (x, ε) for $x \in \{0,1\}^*$, $0 < \varepsilon < 1$ and terminates in time $f(\kappa(x))g(|x|, 1/\varepsilon)$ for an arbitrary computable function f and a function g polynomial in $|x|$ and $1/\varepsilon$.
(2) It holds that
$$\Pr((1-\varepsilon)P(x) \leqslant A(x,\varepsilon) \leqslant (1+\varepsilon)P(x)) \geqslant 3/4.$$

Theorem 13[28] Let $\mathcal{C}$ be a class of graphs of bounded treewidth, then p-#EMBED($\mathcal{C}$) has an FPTRAS.

Since matchings and paths are graphs of bounded treewidth, the theorem implies FPTRAS for both p-#MATCHING and p-#PATH.

The FPTRAS is obtained by a combination of the color-coding method and the sampling scheme, see Ref. [9, Section 14.5] for an alternate and simpler exposition.

Recently, Jerrum and Meeks[44] extended the FPTRAS to count induced subgraphs with property Φ as long as Φ is *monotone* and *every minimal graph with property Φ is of bounded treewidth*.

Formally, a graph property $\Phi = \{\phi_k\}_{k\geqslant 1}$ is a collection of predicates where $k \to \phi_k$ is computable. Each $\phi_k : 2^{\binom{k}{2}} \to \{0, 1\}$ encodes the family of k-vertex graphs that satisfies Φ. Φ is monotone if for every $k \geqslant 1$ and every two k-vertex graphs H and H' that H is a subgraph of H', it holds that H' satisfies Φ if H satisfies Φ.

Consider the following problem:

p-#INDUCEDSUBGRAPHWITHPROPERTY(Φ)	
Instance:	A graph $G = (V, E)$ and an integer k.
Parameter:	k.
Problem:	Compute the number of k-vertex induced subgraphs of G that satisfies Φ.

Theorem 14[44] If Φ is a monotone property and there is a universal constant $w > 0$ such that for every integer k, every edge minimal k-vertex graph satisfying Φ has treewidth at most w, then p-#INDUCEDSUBGRAPHWITHPROPERTY(Φ) has an FPTRAS.

The above algorithm can be applied to, for example, counting connected k-vertex induced graphs, which was shown to be #**P**-hard in the same paper.

Theorem 5 also holds for *labelled* graphs. This enable us to count the number of subgraphs instead of induced subgraphs in some cases, by assigning a label to each vertex.

Approximate counting in the framework of parameterized complexity is still at its infancy. Most of known algorithmic results are based on direct sampling. It is interesting to ask, whether other more sophisticated sampling methods, e.g., the Markov chain Monte-Carlo method, is applicable in this setting.

On the other hand, one may seek deterministic approximation schemes for problems like p-#PATH and p-#MATCHING. Recently, the technique of *correlation decay* is successful applied in classical setting to obtain FPTAS for problems that either a randomized approximation is known (e.g., Ref. [45]), or unknown (e.g., Ref. [46]). We also ask whether the method can be applied in the parameterized setting.

7 Conclusions

In this short article, we surveyed various aspects of parameterized counting complexity, a theory combining parameterized (decision) complexity and classical counting complexity. From the algorithmic perspective, many #**P**-hard problems become fixed-parameter tractable when we choose some appropriate parameters. In fact, there exist general frameworks in which a large number of such results can be explained, i.e., Courcelle's Theorem and the framework of Holant problems. We also discussed some randomized approximation counting algorithms.

On the complexity side, we have a much refined view of counting problems. It results in a much richer structural theory as witnessed by an apparently infinite hierarchy of #**W**-classes whose underlying classical problems are all #**P**-complete. Similar to the classical setting, we have observed a few important problems whose decision versions are easy and whose counting versions are hard. The following is one of the authors' favorite open problems, which somewhat goes the other way around.

Question 2 Is there a problem whose decision version is **W**[1]-hard, while the counting version is *not* #**W**[1]-hard?

The similar problem with respect to #**P** has been open since the very beginning of counting complexity.

As more sweeping results, we have seen a number of parameterized dichotomy theorems on counting graph homomorphisms, embeddings, and strong embeddings, the first one being of particular importance because of its equivalence to constraint satisfaction problems. It is very interesting to note that such dichotomies do not exist in the classical setting due to the corresponding Ladner type of results.

Our survey is by no means comprehensive. Notably, we have not elaborated on the usefulness of *kernelization* for parameterized counting problems. Kernelization is one of main tools for designing fixed-parameter algorithms, and it has been one of most active areas in parameterized complexity recently. However, despite the initial attempt by

Thurley[47], a generally accepted notion of counting kernelization is still missing.

Finally let us mention another important open problem which was first suggested in Ref. [12]. Recall Toda's Theorem, already mentioned in the introduction, states that every problem in the polynomial hierarchy can be solved by a decision algorithm using a #**P**-oracle[6]. It would be very nice to have a parameterized version of Toda's Theorem. Given that there is no single parameterized class that corresponds to #**P**, the following is one of the many possibilities.

Question 3 Can every problem in **W**[P] be decided by a fixed-parameter algorithm using a #**W**[1]-oracle?

There have been some efforts along the line of the classical proof (see Ref. [48]), but the success is very limited.

References

[1] L. G. Valiant, The complexity of enumeration and reliability problems, *SIAM Journal on Computing*, vol. 8, no. 3, pp. 410-421, 1979.

[2] X. Chen, Guest column: Complexity dichotomies of counting problems, *ACM SIGACT News*, vol. 42, no. 4, pp. 54-76, 2011.

[3] M. Dyer and C. Greenhill, The complexity of counting graph homomorphisms, in *Random Structures & Algorithms*, vol. 17. John Wiley & Sons, Inc., 2000, pp. 260-289.

[4] R. E. Ladner, On the structure of polynomial time reducibility, *Journal of the ACM*, vol. 22, no. 1, pp. 155-171, 1975.

[5] L. G. Valiant, The complexity of computing the permanent, *Theoretical Computer Science*, vol. 8, pp. 189-201, 1979.

[6] S. Toda, PP is as hard as the polynomial-time hierarchy, *SIAM Journal on Computing*, vol. 20, no. 5, pp. 865-877, 1991.

[7] R. G. Downey and M. R. Fellows, *Parameterized Complexity*, vol. 3. Springer Heidelberg, 1999.

[8] R. Niedermeier, *Invitation to Fixed-Parameter Algorithms*. Oxford University Press, 2002.

[9] J. Flum and M. Grohe, *Parameterized Complexity Theory*, vol. 3. Springer, 2006.

[10] R. G. Downey and M. R. Fellows, *Fundamentals of Parameterized Complexity*. Springer, 2013.

[11] R. Downey, The birth and early years of parameterized complexity, in *The Multivariate Algorithmic Revolution and Beyond*. Springer, 2012, pp. 17-38.

[12] J. Flum and M. Grohe, The parameterized complexity of counting problems, *SIAM Journal on Computing*, vol. 33, no. 4, pp. 892-922, 2004.

[13] C. McCartin, Parameterized counting problems, *Annals of Pure and Applied Logic*, vol. 138, nos. 1-3, pp. 147-182, 2006.

[14] J. Edmonds, Paths, trees, and flowers, *The Canadian Journal of Mathematics*, vol. 17, pp. 449-467, 1965.

[15] R. Curticapean, Counting matchings of size k is #W[1]-hard, in *Proceedings of the 40th International Colloquium on Automata, Languages and Programming (ICALP'13, Track A)*, 2013, pp. 352-363.

[16] A. A. Bulatov, The complexity of the counting constraint satisfaction problem, *Journal of the ACM*, vol. 60, no. 5, p. 34, 2013.

[17] M. E. Dyer and D. M. Richerby, On the complexity of #CSP, in *Proceedings of the 42nd ACM Symposium on Theory of Computing (STOC'10)*, ACM, 2010, pp. 725-734.

[18] M. Dyer and D. Richerby, The #CSP dichotomy is decidable, in *Proceedings of the 28th International Symposium on Theoretical Aspects of Computer Science (STACS'11)*, vol. 9, pp. 261-272, 2011.

[19] Y. Chen, M. Thurley, and M. Weyer, Understanding the complexity of induced subgraph isomorphisms, in *Proceedings of the 35th International Colloquium on Automata, Languages and Programming (ICALP'08, Track A)*, 2008, pp. 587-596.

[20] H. L. Bodlaender, Fixed-parameter tractability of treewidth and pathwidth, in *The Multivariate Algorithmic Revolution and Beyond*. Springer, 2012, pp. 196-227.

[21] B. Courcelle, Graph rewriting: An algebraic and logic approach, in *Handbook of Theoretical Computer Science*. 1990, pp. 194-242.

[22] S. Arnborg, J. Lagergren, and D. Seese, Easy problems for tree-decomposable graphs, *Journal of Algorithms*, vol. 12, no. 2, pp. 308-340, 1991.

[23] S. Kreutzer and S. Tazari, Lower bounds for the complexity of monadic second-order logic, in *Proceedings of the 25th Annual IEEE Symposium on Logic in Computer Science (LICS'10)*, 2010, pp. 189-198.

[24] J.-Y. Cai, P. Lu, and M. Xia, Computational complexity of holant problems, *SIAM Journal on Computing*, vol. 40, no. 4, pp. 1101-1132, 2011.

[25] J.-Y. Cai, P. Lu, and M. Xia, Holant problems and counting CSP, in *Proceedings of the 41st Annual ACM Symposium on Theory of Computing (STOC'09)*, 2009, pp. 715-724.

[26] L. G. Valiant, Holographic algorithms, *SIAM Journal on Computing*, vol. 37, no. 5, pp. 1565-1594, 2008.

[27] Y. Yin and C. Zhang, Approximate counting via correlation decay on planar graphs, in *Proceedings of the 24th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'13)*, SIAM, 2013, pp. 47-66.

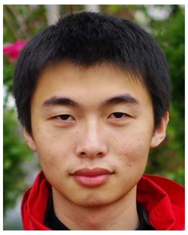
[28] V. Arvind and V. Raman, Approximation algorithms for some parameterized counting problems, in *Proceedings of the 13th International Symposium on Algorithms and Computation*, Springer-Verlag, 2002, pp. 453-464.

[29] N. Alon, R. Yuster, and U. Zwick, Color-coding, *Journal of the ACM*, vol. 42, no. 4, pp. 844-856, 1995.

[30] J. Plehn and B. Voigt, Finding minimally weighted subgraphs, in *Proceedings of the 16th International Workshop on Graph-Theoretic Concepts in Computer Science (WG'90)*, 1990, pp. 18-29.

[31] B. Lin and Y. Chen, The parameterized complexity of k-edge induced subgraphs, in *Proceedings of the 39th International Colloquium on Automata, Languages and Programming (ICALP'12, Track A)*, 2012, pp. 641-652.

[32] M. Bläser and R. Curticapean, Weighted counting of k-matchings is #W[1]-hard, in *Proceedings of the 7th International Symposium on Parameterized and Exact Computation (IPEC'12)*, 2012, pp. 171-181.
[33] R. Curticapean and D. Marx, Private communication, 2014.
[34] D. Marx, Can you beat treewidth? *Theory of Computing*, vol. 6, pp. 85-112, 2010.
[35] M. Frick, Generalized model-checking over locally tree-decomposable classes, *Theory of Computing Systems*, vol. 37, no. 1, pp. 157-191, 2004.
[36] M. Frick and M. Grohe, Deciding first-order properties of locally tree-decomposable structures, *Journal of the ACM*, vol. 48, no. 6, pp. 1184-1206, 2001.
[37] P. Lu, Complexity dichotomies of counting problems, in *Electronic Colloquium on Computational Complexity (ECCC)*, vol. 18, p. 93, 2011.
[38] V. Chandrasekaran, N. Srebro, and P. Harsha, Complexity of inference in graphical models, in *Proceedings of the 24th Conference in Uncertainty in Artificial Intelligence (UAI'08)*, pp. 70-78, 2008.
[39] C. Chekuri and J. Chuzhoy, Polynomial bounds for the grid-minor theorem, *arXiv preprint arXiv:1305.6577*, 2013.
[40] T. Feder and M. Y. Vardi, The computational structure of monotone monadic SNP and constraint satisfaction: A study through datalog and group theory, *SIAM Journal on Computing*, vol. 28, no. 1, pp. 57-104, 1998.
[41] M. Grohe, The complexity of homomorphism and constraint satisfaction problems seen from the other side, *Journal of the ACM*, vol. 54, no. 1, 2007.
[42] V. Dalmau and P. Jonsson, The complexity of counting homomorphisms seen from the other side, *Theoretical Computer Science*, vol. 329, nos. 1-3, pp. 315-323, 2004.
[43] M. Jerrum, A. Sinclair, and E. Vigoda, A polynomial-time approximation algorithm for the permanent of a matrix with nonnegative entries, *Journal of the ACM*, vol. 51, no. 4, pp. 671-697, 2004.
[44] M. Jerrum and K. Meeks, The parameterised complexity of counting connected subgraphs and graph motifs, 2013, arXiv: 1308. 1575.
[45] M. Bayati, D. Gamarnik, D. Katz, C. Nair, and P. Tetali, Simple deterministic approximation algorithms for counting matchings, in *Proceedings of the 39th Annual ACM Symposium on Theory of Computing (STOC'07)*, 2007, pp. 122-127.
[46] D. Weitz, Counting independent sets up to the tree threshold, in *Proceedings of the 38th Annual ACM Symposium on Theory of Computing (STOC'06)*, 2006, pp. 140-149.
[47] M. Thurley, Kernelizations for parameterized counting problems, in *Proceedings of the 4th International Conference on Theory and Applications of Models of Computation (TAMC'07)*, Springer, 2007, pp. 703-714.
[48] J. A. Montoya and M. Müller, Parameterized random complexity, *Theory of Computing Systems*, vol. 52, no. 2, pp. 221-270, 2013.



Chihao Zhang is currently a PhD candidate at Department of Computer Science and Engineering, Shanghai Jiao Tong University. He received his master degree in computer science from Shanghai Jiao Tong University. His main research interest lies in the design and analysis of algorithms.

Yijia Chen is a professor at Department of Computer Science and Engineering, Shanghai Jiao Tong University. He received his PhD degree in computer science from Shanghai Jiao Tong University and PhD degree in mathematics from University of Freiburg. He mainly works in logic and complexity.