Tsinghua Science and Technology

Volume 19 | Issue 4 Article 3

2014

Kernelization in Parameterized Computation: A Survey

Qilong Feng
School of Information Science and Engineering, Central South University, Changsha 410083, China.

Qian Zhou
School of Information Science and Engineering, Central South University, Changsha 410083, China.

Wenjun Li
School of Information Science and Engineering, Central South University, Changsha 410083, China.

Jianxin Wang
School of Information Science and Engineering, Central South University, Changsha 410083, China.

TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214 03/08 pp338-345
Volume 19, Number 4, August 2014

Kernelization in Parameterized Computation: A Survey

Qilong Feng, Qian Zhou, Wenjun Li, and Jianxin Wang*

Abstract: Parameterized computation is a new method dealing with NP-hard problems, which has attracted a lot of attentions in theoretical computer science. As a practical preprocessing method for NP-hard problems, kernelizaiton in parameterized computation has recently become an active research area. In this paper, we discuss several kernelizaiton techniques, such as crown decomposition, planar graph vertex partition, randomized methods, and kernel lower bounds, which have been used widely in the kernelization of many hard problems.

Key words: parameterized computation; kernelization; parameterized algorithm; NP-hard

1 Introduction

Parameterized computation is a theory to cope with NP-hard problems[1], which has been widely applied to solving problems in the fields of bioinformatics[2], social science[3,4], networks[5-7], computation geometry[8], etc. A parameterized problem Q is a language that is a subset of $\Sigma^* \times N$, where Σ is a fixed alphabet and N is the set of all non-negative integers. Then, each element of Q is of the form (x,k), where k is the parameter. A parameterized problem is called fixed-parameter tractable if there exists an algorithm, for some computable function f, solving the problem in time $O(f(k)|x|^{O(1)})$ on input (x,k). The framework of parameterized computation includes some major algorithmic techniques such as kernelization, color-coding, iterative compression, bounded search tree, and randomized methods.

As an efficient preprocessing method, kernelization has become one of the most active research topics in parameterized computation. Given an instance (x,k) of parameterized problem A, a kernelization algorithm for problem A is a polynomial algorithm that transforms instance (x,k) to a new instance (x',k') such that $|x'| \leqslant g(k')$, $k' \leqslant k$, where g is a computable function that depends only on k', such that (x,k) is a yes-instance of problem A if and only if (x',k') is a yes-instance of problem A. The power of kernelization can be demonstrated by the following example of Vertex Cover problem. Much attention has been paid on the Vertex Cover problem and related problems[9]. For a graph $G=(V,E)$, and a subset $C \subseteq V$, if at least one endpoint of each edge in E is contained in C, then C is called a *vertex cover* of G. The definition of the Vertex Cover problem is that: for a given graph $G=(V,E)$ and a parameter k, to decide whether G contains a vertex cover of size at most k. By dealing with degree-0, degree-1 vertices, and the vertices with degree higher than k, a simple kernelization algorithm can be obtained with the following repetitive process. Obviously, all the isolated vertices (the vertices with degree 0) in G can be deleted. For a degree-1 vertex v in G, the neighbor of v can be directly put into C, and set $k'=k-1$. If there exists a vertex v in G with degree at least $k+1$, then v is in C, and set $k'=k-1$, otherwise a vertex cover of size k cannot be found. It is easy to see that the above kernelization algorithm for Vertex Cover can be done in polynomial time, and let the reduced instance be $(G'=(V',E'),k')$. Since the number of vertices in objective vertex cover is at most k', and the degree of each vertex in G' is bounded by k', the size of V' is bounded by k'^2.

In this paper, we discuss several kernelizaiton

• Qilong Feng, Qian Zhou, Wenjun Li, and Jianxin Wang are with the School of Information Science and Engineering, Central South University, Changsha 410083, China. E-mail: jxwang@csu.edu.cn.
∗ To whom correspondence should be addressed.
Manuscript received: 2014-06-09; accepted: 2014-06-17

techniques, including crown decomposition, planar graph vertex partition, randomized kernelization, and kernel lower bounds, and show the powerful applications of these techniques.

2 Kernelization Techniques

We first give some notations and terminologies used in the paper. For a graph $G = (V, E)$ and a vertex $v \in V$, let $N(v)$ be the set of neighbors of v, i.e., $N(v) = \{u|(u,v) \in E\}$. For a subset $V' \subset V$, let $G[V']$ denote the subgraph induced by the vertices in V', and let $N(V')$ denote the set of out neighbors of the vertices in V', i.e., $N(V') = (\cup_{v \in V'} N(v)) \backslash V'$. A set M of edges of G is called a matching of G, if no two edges in M share a common endpoint. For a matching M of G, let $V(M)$ denote the set of vertices contained in M, i.e., the set of matched vertices of G. For a vertex v in connected graph G, if $G[V \backslash \{v\}]$ contains at least two connected components, then v is called a cut-vertex. In graph G, a vertex with degree i is called a degree-i vertex.

2.1 Crown decomposition

Crown decomposition is one of the most popular used techniques in parameterized computation. Many problems, such as Vertex Cover, P_2-Packing, Edge Disjoint Triangle Packing, and Vertex Disjoint Triangle Packing, can be preprocessed efficiently by applying the crown decomposition technique[10-15]. We first give the definition of crown decomposition.

Definition 1[11] A *crown decomposition* of graph G is a decomposition (H, C, R) of the vertices of G such that

(1) H (the head) separates C and R;

(2) $C = C_u \cup C_m$ (the crown) is an independent set such that $|C_m| = |H|$, and there exists a perfect matching between C_m and H.

We now take Vertex Cover problem to see how the crown decomposition technique is used in the kernelization of NP-hard problems. The relationship between vertex cover and crown decomposition is as follows.

Lemma 1[11] Given a graph $G = (V, E)$ and a crown decomposition (H, C, R) of G, there exists a minimum vertex cover of G containing all the vertices in H.

Based on above lemma, it is easy to get the following property.

Lemma 2[11] Given an instance (G, k) of the Vertex Cover problem, and a crown decomposition (H, C, R) of G, there is a vertex cover of size k in G if and only if there exists a vertex cover of size $k - |H|$ in graph $G[V \backslash (H \cup C)]$.

From the above two lemmas, we can see that for a given instance (G, k) of the Vertex Cover problem, as long as there exists a crown decomposition in G, instance (G, k) can be reduced to a smaller instance (denoted by (G', k') $(k' \leqslant k)$) such that (G, k) is a yes-instance of the Vertex Cover problem if and only if (G', k') is a yes-instance of the Vertex Cover problem. Based on this idea, we give the following kernelization steps for the Vertex Cover problem.

(1) Delete all the isolated vertices from G.

(2) Find a maximum matching M_1 in G, and if $|M_1| > k$, then return "No".

(3) Let $I = V - V(M_1)$ and find a maximum matching M_2 between I and $N(I)$; if $|M_2| > k$, then return "No".

(4) If there exists a crown decomposition (H, C, R) in graph $G[I \cup N(I)]$, then $G' = G[V \backslash (H \cup C)]$, $k' = k - |H|$, otherwise $G' = G, k' = k$; return (G', k').

It is easy to get that Steps 1-4 can be done in polynomial time. The correctness of the Steps 1-3 can be easily obtained based on the properties of vertex cover. By Lemma 2, Step 4 can handle the crown decomposition well if there exists one. Obviously, the number of vertices in M_1 is bounded by $2k$. Let $I_0 = I - V(M_2)$. It is easy to see that if $I_0 \neq \varnothing$, then a crown decomposition can be found, and graph G can be reduced correspondingly. If $I_0 = \varnothing$, then the number of vertices in $V(M_2) - V(M_1)$ is bounded by k. Therefore, after applying the above kernelization algorithm, the total number of vertices in G' is bounded by $3k$, i.e., the Vertex Cover problem admits a kernel of size $3k$.

Based on the above crown decomposition, many other forms of crown decomposition have been proposed to obtain kernels for NP-hard problems. We now take P_2-Packing problem to further illustrate the power of crown decomposition. A P_2 is a simple path of length three. For a set B of P_2s, if no two P_2s in B have common vertices, then B is called a P_2-Packing. The definition of the Parameterized P_2-Packing problem is that: given a graph $G = (V, E)$ and a parameter k, to decide whether there exists a P_2-Packing of size k in G. For the kernelization of the Parameterized P_2-Packing problem, two variations of crown decomposition are applied, as follows.

Definition 2[12] A *fat crown decomposition* of graph G is a decomposition (H, C, R) of the vertices of G such that

(1) H (the head) separates C and R;

(2) The induced subgraph $G[C]$ is a graph in which each connected component is isomorphic to the complete graph K_2 (i.e, a single edge);

(3) There is a matching M between H and a subset of vertices in C such that $|M| = |H|$, and each connected component in C has at most one vertex in M.

Definition 3[12] A *double crown decomposition* of graph G is a decomposition (H, C, R) of the vertices of G such that

(1) H (the head) separates C and R;

(2) $C = C_u \cup C_m \cup C'_m$ (the crown) is an independent set such that $|C_m| = |C'_m| = |H|$, and there exists a perfect matching between C_m and H, and a perfect matching between C'_m and H.

For an instance $(G = (V, E), k)$ of the Parameterized P_2-Packing problem, let $W = \{L_1, L_2, \cdots, L_h\}$ $(h \leqslant k-1)$ be a maximal P_2-Packing of G. Assume that $V(W)$ is the set of vertices contained in W, and let $Q = V \backslash V(W)$. It is easy to see that each connected component in $G[Q]$ is either an isolated vertex or a K_2. Assume that Q_0 is the set of isolated vertices in $G[Q]$, and Q_1 is the set of K_2s in $G[Q]$. For a vertex v in $G[Q]$, v is called a Q_0-vertex, and for an edge e in $G[Q]$, e is called a Q_1-edge in $G[Q]$. The following two rules are to reduce the number of Q_0-vertices and Q_1-edges, which play important role in the kernelization of Parameterized P_2-Packing problem[13].

Rule 1 If a P_2 L_i in W has two vertices such that each is adjacent to a different Q_0-vertex, then L_i can be replaced with a new P_2 such that the number of Q_0-vertices is decreased by two and the number of Q_1-edges is increased by one.

Rule 2 If a P_2 L_i in W has two vertices such that each is adjacent to a different Q_1-edge, then L_i can be replaced with two new P_2s such that the number of Q_1-edges is decreased by two, the number of Q_0-vertices is increased by one, and the size of W is increased by one.

Based on the above two reduction rules, the number of Q_0-vertices and Q_1-edges in the reduced graph can be bounded in the following way.

Lemma 3[13] Let $W = \{L_1, L_2, \cdots, L_h\}$ be a maximal P_2-Packing on which Rules 1 and 2 are not applicable, where $h \leqslant k-1$. If the number of Q_0-vertices is larger than $2k-3$, then there is a double crown, which can be constructed in linear time.

Lemma 4[13] Let $W = \{L_1, L_2, \cdots, L_h\}$ be a maximal P_2-Packing on which Rules 1 and 2 are not applicable, where $h \leqslant k-1$. If the number of Q_1-edges is larger than $k-1$, then there is a fat crown, which can be constructed in linear time.

For a given instance (G, k) of the Parameterized P_2-Packing problem, if there exists double crown or fat crown in graph G, then the instance (G, k) can be reduced correspondingly, as follows.

Lemma 5[12] For an instance (G, k) of the Parameterized P_2-Packing problem, if there exists a double crown (H, C, R) in G, then G has a P_2-Packing of size k if and only if graph $G[V \backslash (H \cup C)]$ has a P_2-Packing of size $k - |H|$.

Lemma 6[12] For an instance (G, k) of the Parameterized P_2-Packing problem, if there exists a fat crown (H, C, R) in G, then G has a P_2-Packing of size k if and only if graph $G[V \backslash (H \cup C)]$ has a P_2-Packing of size $k - |H|$.

Based on the above lemmas, the kernelization algorithm for the Parameterized P_2-Packing problem can be described as follows.

(1) Find a maximal P_2-Packing W in G, and if $|W| \geqslant k$, return "Yes".

(2) Repeatedly applying Rules 1 and 2 (keeping W maximal) until neither Rule 1 nor Rule 2 is applicable, and if $|W| \geqslant k$, then return "Yes".

(3) If there exists a double crown (H, C, R) in graph G, then $G' = G[V \backslash (H \cup C)]$, $k' = k - |H|$.

(4) If there exists a fat crown (H, C, R) in graph G, then $G' = G[V \backslash (H \cup C)]$, $k' = k - |H|$.

The above four steps can be done in polynomial time, and a reduced instance (G', k') can be obtained. By Lemma 5 and Lemma 6, (G, k) has a P_2-Packing of size k if and only if (G', k') has a P_2-Packing of size k'. We now show a bound for the number of vertices in G'. If $|W| \geqslant k$, obviously, (G, k) is a yes-instance. Assume that W' is the maximal P_2-Packing in G' obtained from W by using reduction Rules 1 and 2. Then, the number of vertices of W' is bounded by $3k'$. The number of Q_0-vertices and Q_1-edges in G' is bounded by $2k'-3$ and $k'-1$ respectively, otherwise a double crown or a fat crown can be found. Therefore, the total number of vertices in G' is bounded by $|W'| + |Q_0| + |Q_1| = 3(k'-1) + 2k' - 3 + 2(k'-1) = 7k' - 8$, bounding the size of the kernel of the Parameterized P_2-Packing problem.

2.2 Planar graph vertex partition

Kernelization for planar graph problems has been extensively studied, yielding several well-known kernelization techniques. Recently, Wang et al.[16] proposed a simple and powerful vertex partition technique for the kenrnelization of planar graph problems. The vertex partition kernelization technique works very well for the planar graph problems with small distance property, such as Planar Connected Vertex Cover, Planar Edge Dominating Set, and Planar Maximum Triangle Packing. We first define the distance property of graph problems.

Definition 4[16] A graph problem on input $G = (V, E)$ is said to admit the distance property with constant c_V if

(1) The problem asks for a solution S, which is a subgraph of G, satisfying a specified property, and

(2) For each solution S to the problem and its vertex set $V(S)$, the following conditions hold: $\forall u \in V, \exists v \in V(S) : d(u, v) \leqslant c_V$, where the distance $d(v, u)$ between vertices v and u is the length of a shortest path between v and u.

The vertex partition method proposed in Ref. [16] deals with the graph problems with distance property $c_V = 1$, and upper bounds on the kernel size can be directly obtained without introducing new reduction rules. Let A be a planar graph problem with distance property $c_V = 1$. Assume that $(G = (V, E), k)$ is an instance of problem A, and S is the vertex set of a solution of instance (G, k). Let $J = V \backslash S$. We now give a general framework using vertex partition method to analyze the kernels of planar graph problems.

(1) Define the following subsets,
$J_3 = \{v \in J \,\big|\, |N(v) \cap S| \geqslant 3\}$,
$J_2 = \{v \in J \,\big|\, |N(v) \cap S| = 2\}$,
$J_1 = \{v \in J \,\big|\, |N(v) \cap S| = 1\}$,
$J_0 = J \backslash (J_3 \cup J_2 \cup J_1)$.

(2) Based on the reduction rules, analyze the size of J_0, J_1, J_2, and J_3, respectively.

(3) The kernel size is bounded by $|S| + |J_0| + |J_1| + |J_2| + |J_3|$.

Key point to use the vertex partition method is to bound the sizes of J_0, J_1, J_2, and J_3. For the planar graph satisfying distance property $c_V = 1$, it is easy to get that $|J_0| = 0$. For J_3, the planarity of graph G can be used directly to bound the size of J_3, as follows.

Lemma 7[16] For an instance (G, k) of problem A, and the solution set $S \subset V$ of A, $|J_3| \leqslant \max\{0, 2(|S| - 2)\}$.

The basic idea to get the result in Lemma 7 is that: In the bipartite graph (denoted by B) with two components J_3 and S, based on the relationship between the number of edges and the number of vertices in triangle-free planar graph B, the size of J_3 can be bounded directly.

In order to bound J_2 efficiently, J_2 can be partitioned into the following subsets $J_2(uv) = \{w \in J_2 | N(w) \cap S = \{u, v\}\}$ for all distinct $u, v \in S$. By the planarity of graph G, the number of such subsets that are non-empty can be bounded, as follows.

Lemma 8[16] For an instance (G, k) of problem A, and the solution set $S \subset V$ of A, there are at most $\max\{1, 3|S| - 6\}$ pairs of distinct vertices u and v in S for which the corresponding subset $J_2(uv)$ is non-empty.

To obtain a kernel of problem A, the remaining problem is to bound the size of J_1. The analysis of the size of J_1 is closely related to the reduction rules of the problem, and may vary a lot for different problems.

We now take Planar Connected Vertex Cover problem to illustrate the powerful application of the above technique. For a graph G, and a connected vertex cover V' of G is a vertex cover satisfying that $G[V']$ is connected. The definition of the Planar Connected Vertex Cover problem is that: given a planar graph $G = (V, E)$ and a parameter k, to decide whether there exists a connected vertex cover of size at most k in G. Clearly, the Planar Connected Vertex Cover problem satisfies distance property $c_V = 1$. Let $N^1(v) = \{u | u \in N(v)$, u has degree one$\}$. The following reduction rules are used in the kernelization of the Planar Connected Vertex Cover problem[16].

Rule 3 If there is a vertex v with $|N^1(v)| \geqslant 2$, remove all but one vertex in $N^1(v)$.

Rule 4 For a degree-2 vertex v with $N(v) = \{u, w\}$, if v is not a cut-vertex, then remove v; if $N^1(u) = \varnothing$, then add a new degree-1 vertex adjacent to u, and if $N^1(w) = \varnothing$, then add a new degree-1 vertex adjacent to w.

It is easy to see that Rules 3 and 4 can be done in polynomial time. The kernelization for the Planar Connected Vertex Cover problem is to exhaustively apply the reduction Rules 3 and 4 on given instance (G, k), and denote the reduced instance by (G', k'). For the reduced instance (G', k'), either it can be solved trivially, or a kernel is obtained. We now analyze the size of J_1, J_2, and J_3, respectively. By Lemma 7, $|J_3| \leqslant \max\{0, 2|S| - 4\}$. Because of Rule 3, $|J_1| \leqslant |S|$. By Rule 4, $J_2 = \varnothing$. Therefore, the size of reduced instance (G', k') is bounded by $|S| + |J_1| + |J_3| \leqslant 4k - 4$, which is the kernel of the Planar Connected Vertex Cover problem.

2.3 Randomized kernelization

Recently, random methods have been applied to parameterized algorithm[17, 18] and kernelization, and the notion of randomized kernelization technique has been introduced. Employing algebraic, probabilistic methods, and matroid theory, randomized kernelization has been applied to solve problems like Parameterized Max-r-SAT[19], Parameterized Linear Ordering Above Tight Lower Bound[19], and Odd Cycle Transversal[20]. Moreover, based on the probability analysis of problem structure and expectation method, Chen and Lu[21] proposed an improved kernel for the Parameterized Set Splitting problem, which is highly applicable to other NP-hard problems. In this subsection, we take Parameterized Set Splitting problem as an example to show how to use probability analysis and expectation in kernelization.

For a set X, (X_1, X_2) is called a partition of X, if $X_1 \cup X_2 = X$ and $X_1 \cap X_2 = \varnothing$. For a subset F of X that has the partition (X_1, X_2), if $F \cap X_1 \neq \varnothing$ and $F \cap X_2 \neq \varnothing$, then F is split by partition (X_1, X_2). The Parameterized Set Splitting problem is that: given a ground set X, a collection $\mathcal{F}$ of subsets of X, and a parameter k, to decide whether there exists a partition of X that splits at least k sets in $\mathcal{F}$.

For an instance $(X, \mathcal{F}, k)$ of the Parameterized Set Splitting problem, it is easy to get the following reduction rule.

Rule 5 If a set S in $\mathcal{F}$ has only one element, then remove S from $\mathcal{F}$.

Repeatedly apply Rule 5 to $\mathcal{F}$. We now assume that each set in $\mathcal{F}$ has at least two elements. We first give the following kernel result for the Parameterized Set Splitting problem.

Lemma 9[21] For an instance $(X, \mathcal{F}, k)$ of the Parameterized Set Splitting problem, if $|\mathcal{F}| \geqslant 2k$, then there exists a partition of X splitting at least k sets in $\mathcal{F}$.

Lokshtanov and Sloper[22] applied the crown decomposition method to get the above kernel result. Chen and Lu[21] reanalyzed the kernel result in Lemma 9 using probabilistic method to get deterministic kernelization. We now give the general idea of the probabilistic method used in Ref. [21]. For an instance $(X, \mathcal{F}, k)$ of the Parameterized Set Splitting problem, let W denote the set of all elements contained in the sets in $\mathcal{F}$, and assume that $|W| = t$. Then, the elements in W can be partitioned into W_1 and W_2 by the following way: randomly pick $\lfloor t/2 \rfloor$ elements to put into W_1, and put the remaining $t - \lfloor t/2 \rfloor$ elements into W_2. Therefore, for any set S in $\mathcal{F}$, the probability that S is split is at least $2\binom{(t-2)}{\lfloor t/2 \rfloor - 1} \Big/ \binom{t}{\lfloor t/2 \rfloor} > 1/2$. For each set S in $\mathcal{F}$, let X_S be a random variable such that if S is split by (W_1, W_2), then $X_S = 1$, otherwise $X_S = 0$. Then, the expected number of split sets in $\mathcal{F}$ is $E\left(\sum_{S \in \mathcal{F}} X_S\right) \geqslant |\mathcal{F}|/2$. Therefore, if $|\mathcal{F}| \geqslant 2k$, the expected number of the split sets in $\mathcal{F}$ is at least k, which means that there must exist a partition of X such that at least k sets in $\mathcal{F}$ are split, i.e., $(X, \mathcal{F}, k)$ is a yes-instance of the Parameterized Set Splitting problem.

For an instance $(X, \mathcal{F}, k)$ of the Parameterized Set Splitting problem, if a set S in $\mathcal{F}$ contains at least k elements, then there always exists a partition of X that splits S, as the following rule.

Rule 6 If a set in $\mathcal{F}$ contains at least k elements, then remove S from $\mathcal{F}$ and decrease k by one.

Repeatedly apply Rules 5 and 6 to the instance $(X, \mathcal{F}, k)$, and let $(X', \mathcal{F}', k')$ be the reduced instance of the Parameterized Set Splitting problem. The idea to get improved kernel for the Parameterized Set Splitting problem is that for a set S in $\mathcal{F}'$, the more elements S contains, the better chance that S is split.

Lemma 10[21] For a reduced instance $(X', \mathcal{F}', k')$ of the Parameterized Set Splitting problem, if $\sum_{i=2}^{k'-1} \frac{2^i - 2}{2^i} m_i \geqslant k'$, then there exists a partition of X' such that at least k' sets in $\mathcal{F}'$ are split, where m_i is the number of sets in $\mathcal{F}'$ containing i elements.

We now give the general idea to prove the above lemma from probabilistic perspective. Assume that (X_1', X_2') is a random partition of X' obtained by putting the elements of X' into X_1', X_2' with probability 1/2 respectively. Then, for any set S in $\mathcal{F}'$ containing i elements, the probability that S is split is $\frac{2^i - 2}{2^i}$, and let X_S be a random variable such that if S is split, then $X_S = 1$, otherwise $X_S = 0$. Therefore, the expected number of split sets in $\mathcal{F}'$ is $E\left(\sum_{S \in \mathcal{F}'} X_S\right) =$

$\sum_{i=1}^{k'-1}\sum_{|S|=i} E(S \text{ is split}) = \sum_{i=1}^{k'-1}\frac{2^i-2}{2^i}m_i$. Therefore, if $\sum_{i=2}^{k'-1}\frac{2^i-2}{2^i}m_i \geqslant k'$, the expected number of the split sets in $\mathcal{F}$ is at least k', which means that there must exist a partition of X' such that at least k' sets in $\mathcal{F}'$ are split, i.e., $(X', \mathcal{F}', k')$ is a yes-instance of the Parameterized Set Splitting problem.

Based on Lemma 10, we can get the following result.

Theorem 1[21] For a reduced instance $(X', \mathcal{F}', k')$ of the Parameterized Set Splitting problem by applying Rules 5 and 6, then $|\mathcal{F}'| < 2k' - \sum_{i=3}^{k'-1}\frac{2^{i-1}-2}{2^{i-1}}m_i$, where m_i is the number of sets in $\mathcal{F}'$ containing i elements.

3 Kernelization Lower Bound

Kernelizaiton lower bound technique has attracted some attention in parameterized computation. Recently, several frameworks for proving lower bound of kernel size have been developed for some NP-hard problems. Bodlaender et al.[23] proved that many parameterized problems are unlikely to have polynomial size kernel under certain complexity-theoretical hypothesis. Based on the kernel lower bound framework in Ref. [23], many lower bound results are given to show that some parameterized problems are unlikely to have polynomial kernel[24-30]. As for the problems with polynomial kernels, Dell and van Melkebeek[31] developed a framework to prove lower bounds of the polynomial kernels. For example, in Ref. [31], they gave that Vertex Cover problem does not have a kernel of size $O(k^{2-\varepsilon})$ unless coNP$\subseteq$NP/poly. Kratsch et al.[32] applied the framework in Ref. [31] to get kernel lower bound for the Parameterized Point Line Cover problem. Dell and Marx[33] further developed the framework in Ref. [31], and proved kernel lower bounds of packing problems. In this subsection, we use Parameterized Point Line Cover problem to illustrate the powerful technique developed in Ref. [31].

Oracle communication protocol plays an important role for the kernel lower bound technique in Ref. [31], whose definition is as follows.

Definition 5[31] An oracle communication protocol for a language L is a communication protocol between two players. The first player is given the input x and has to run in time polynomial in the length of the input; the second player is computationally unbounded but is not given any part of x. At the end of the protocol the first player should be able to decide whether $x \in L$. The cost of the protocol is the number of bits of communication from the first player to the second player.

The relationship between lower bound and the oracle communication protocol is as follows.

Theorem 2[31] Let $d \geqslant 3$ be an integer and ε be a positive real. If coNP $\nsubseteq$ NP/poly, there is no protocol of cost $O(n^{d-\varepsilon})$ to decide whether an n-variable d-CNF (each clause has size d) formula is satisfiable, even when the first player is conondeterministic.

Based on Theorem 2, Dell and Melkebeek[31] obtained the following result, which is used in Ref. [32] to get the kernel lower bound of Parameterized Point Line Cover problem.

Theorem 3[32] The Vertex Cover problem admits no oracle communication protocol of cost $O(k^{2-\varepsilon})$ for deciding whether a graph G has a vertex cover of size at most k, for any $\varepsilon > 0$, unless coNP $\subseteq$ NP/poly.

The definition of the Parameterized Point Line Cover problem is that: given a set P of n points in the plane, and a parameter k, to decide whether there exists at most k lines to cover all the points in P. In order to get the kernel lower bound of the Parameterized Point Line Cover problem, the following problem is used.

Parameterized Line Point Cover problem: Given a set L of n lines in the plane, and a parameter k, is there a set of at most k points in the plane that covers all the lines in L?

We now give the relationship between Parameterized Point Line Cover problem and the Parameterized Line Point Cover problem.

Lemma 11[32] There is a polynomial-time reduction from the Parameterized Line Point Cover problem to the Parameterized Point Line Cover problem which preserves the parameter k.

The main part of getting the lower bound for the Parameterized Point Line Cover problem is to get a reduction from the Vertex Cover problem to the Parameterized Line Point Cover problem, as follows.

Lemma 12[32] There is a polynomial-time reduction from the Vertex Cover problem to the Parameterized Line Point Cover problem which maps instances (G, k) of Vertex Cover problem to equivalent instances $(L, 2k)$ of the Parameterized Line Point Cover problem.

Based on Lemma 11 and Lemma 12, it is easy to get the kernel lower bound of the Parameterized Point Line

Cover problem.

Theorem 4[32] Let $\varepsilon > 0$. The Parameterized Point Line Cover problem admits no oracle communication protocol of cost $O(k^{2-\varepsilon})$ for deciding instances (P, k), unless coNP⊆NP/poly.

4 Conclusions

Kernelization in parameterized computation has become a topic of great attention, and many new techniques have been proposed. In this paper, we discuss several kernelization techniques that have been used widely in kernelization of parameterized problems. There are other kernelization results discussed in Refs. [24, 34-40].

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Nos. 61173051, 61103033, and 61232001).

References

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

[2] T. Piovesan and S. Kelk, A simple fixed parameter tractable algorithm for computing the hybridization number of two (not necessarily binary) trees, *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, vol. 10, no. 1, pp. 18-25, 2013.

[3] H. Liu and D. Zhu, Parameterized complexity of control by voter selection in Maximin, Copeland, Borda, Bucklin, and Approval election systems, *Theoretical Computer Science*, vol. 498, pp. 115-123, 2013.

[4] N. Betzler, J. Guo, and R. Niedermeier, Parameterized computational complexity of Dodgson and Young elections, *Information and Computation*, vol. 208, no. 2, pp. 165-177, 2010.

[5] J. Wang, W. Luo, Q. Feng, J. Guo, and J. Chen, Improved linear problem kernel for planar connected dominating set, *Theoretical Computer Science*, vol. 511, pp. 2-12, 2013.

[6] J. Wang, W. Luo, Q. Feng, and J. Guo, Parameterized complexity of Min-power multicast problems in wireless ad hoc networks, *Theoretical Computer Science*, vol. 508, pp. 16-25, 2013.

[7] W. Luo, J. Wang, J. Guo, and J. Chen, Parameterized complexity of max-lifetime target coverage in wireless sensor networks, *Theoretical Computer Science*, vol. 518, pp. 32-41, 2014.

[8] J. Wang, P. Tan, J. Yao, Q. Feng, and J. Chen, On the minimum link-length rectilinear spanning path problem: Complexity and algorithms, *IEEE Transactions on Computers*, 2014, Doi.ieeecomputersociety.org/10.1109/TC.2013.163.

[9] J. Wang, W. Li, S. Li and J. Chen, On the parameterized vertex cover problem for graphs with perfect matching, *Science China Information Sciences*, vol. 57, pp. 1-12, 2014.

[10] F. Abu-Khzam, M. Fellows, M. A. Langston, and W. H. Suters, Crown structures for vertex cover kernelization, *Theory of Computing Systems*, vol. 41, pp. 411-430, 2007.

[11] F. Dehne, M. Fellows, F. Rosamond, and P. Shaw, Greedy localization, iterative compression and modeled crown reductions: New FPT techniques and improved algorithms for max set splitting and vertex cover, in *Proceedings of the First International Workshop on Parameterized and Exact Computation (IWPEC)*, vol. 3162 of LNCS, 2004, pp. 271-281.

[12] E. Prieto and C. Sloper, Looking at the stars, *Theoretical Computer Science*, vol. 351, pp. 437-445, 2006.

[13] J. Wang, D. Ning, Q. Feng, and J. Chen, An improved kernelization for P_2-packing, *Information Processing Letters*, vol. 110, pp. 188-192, 2010.

[14] M. Chlebłka and J. Chlebłkovb, Crown reductions for the minimum weighted vertex cover problem, *Discrete Applied Mathematics*, vol. 156, pp. 292-312, 2008.

[15] L. Mathieson, E. Prieto, and P. Shaw, Packing edge disjoint triangles: A parameterized view, in *Proccedings of the International Workshop on Parametrized and Exact Computation (IWPEC)*, vol. 3162 of LNCS, 2004, pp. 127-137.

[16] J. Wang, Y. Yang, J. Guo, and J. Chen, Planar graph vertex partition for linear problem kernels, *Journal of Computer and System Sciences*, vol. 79, no. 5, pp. 609-621, 2013.

[17] Q. Feng, J. Wang, S. Li, and J. Chen, Randomized parameterized algorithms for $P2$-packing and co-path packing problems, *Journal of Combinatorial Optimization*, 2014, DOI 10.1007/s10878-013-9691-z.

[18] Q. Feng, J. Wang, and J. Chen, Matching and weighted P2-packing: Algorithms and kernels, *Theoretical Computer Science*, vol. 522, pp. 85-94, 2014.

[19] G. Gutin, E. J. Kim, S. Szeider, and A. Yeo, A probabilistic approach to problems parameterized above or below tight bounds, *J. Comput. Syst. Sci.*, vol. 77, pp. 422-429, 2011.

[20] S. Kratsch and M.Wahlström, Compression via matroids: A randomized polynomial kernel for odd cycle transversal, in *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2012, pp. 94-103.

[21] J. Chen and S. Lu, Improved parameterized set splitting algorithms: A probabilistic approach, *Algorithmica*, vol. 54, no. 4, pp. 472-489, 2008.

[22] D. Lokshtanov and C. Sloper, Fixed parameter set splitting, linear kernel and improved running time, in *ACiD, volume 4 of Texts in Algorithmics*, pp. 105-113, 2005.

[23] H. L. Bodlaender, R. G. Downey, M. R. Fellows, and D. Hermelin, On problems without polynomial kernels, *J. Comput. Syst. Sci.*, vol. 75, pp. 423-434, 2009.

[24] H. L. Bodlaender, B. M. P. Jansen, and S. Kratsch, Cross-composition: A new technique for kernelization lower bounds, in *Proceedings of the 28th International Symposium on Theoretical Aspects of Computer Science (STACS)*, 2011, pp. 165-176.

[25] H. L. Bodlaender, S. Thomass, and A. Yeo, Kernel bounds for disjoint cycles and disjoint paths, in *Proceedings of the 17th Annual European Symposium on Algorithms (ESA)*, vol. 5757 of LNCS, 2009, pp. 635-646.

[26] Y. Chen, J. Flum, and M. Mller, Lower bounds for kernelizations and other preprocessing procedures, *Theory of Computing Systems*, vol. 48, no. 4, pp. 803-839, 2011.

[27] M. Dom, D. Lokshtanov, and S. Saurabh, Incompressibility through colors and IDs, in *Proceedings of the 36th International Colloquium on Automata, Languages and Programming (ICALP)*, vol. 5555 of LNCS, 2009, pp. 378-389.

[28] H. Fernau, F. V. Fomin, D. Lokshtanov, D. Raible, S. Saurabh, and Y. Villanger, Kernel(s) for problems with no kernel: On outtrees with many leaves, in *Proceedings of the 26th International Symposium on Theoretical Aspects of Computer Science (STACS)*, 2009, pp. 421-432.

[29] N. Misra, V. Raman, and S. Saurabh, Lower bounds on kernelization, *Discrete Optimization*, vol. 8, no. 1, pp. 110-128, 2011.

[30] S. Kratsch and M. Wahlström, Two edge modification problems without polynomial kernels, in *Proceedings of the 4th International Workshop on Parameterized and Exact Computation (IWPEC)*, vol. 5917 of LNCS, 2009, pp. 264-275.

[31] H. Dell and D. van Melkebeek, Satisfiability allows no nontrivial sparsification unless the polynomial-time hierarchy collapses, in *Proceedings of the 42th Annual ACM Symposium on Theory of Computing (STOC)*, 2010, pp. 251-260.

[32] S. Kratsch, G. Philip, and S. Ray, Point line cover: The easy kernel is essentially tight, in *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2014, pp. 1596-1606.

[33] H. Dell and D. Marx, Kernelization of packing problems, in *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2012, pp. 68-81.

[34] H. L. Bodlaender, B. M. P. Jansen, and S. Kratsch, Preprocessing for treewidth: A combinatorial analysis through kernelization, in *Proceedings of the 38th International Colloquium Automata, Languages and Programming (ICALP)*, 2011, pp. 437-448.

[35] F. V. Fomin, D. Lokshtanov, N. Misra, G. Philip, and S. Saurabh, Hitting forbidden minors: Approximation and kernelization, in *Proceedings of the 28th International Symposium on Theoretical Aspects of Computer Science (STACS)*, 2011, pp. 189-200.

[36] F. Fomin, D. Lokshtanov, S. Saurabh, and D. Thilikos, Linear kernels for (connected) dominating set on graphs with excluded topological subgraphs, in *Proceedings of the 30th International Symposium on Theoretical Aspects of Computer Science (STACS)*, 2013, pp. 92-103.

[37] L. Fortnow and R. Santhanam, Infeasibility of instance compression and succinct PCPs for NP, in *Proceedings of the 40th Annual ACM Symposium on Theory of Computing (STOC)*, 2008, pp. 133-142.

[38] B. M. Jansen and H. L. Bodlaender, Vertex cover kernelization revisited, *Theory of Computing Systems*, vol. 53, no. 2, pp. 263-299, 2013.

[39] S. Kratsch, Co-nondeterminism in compositions: A kernelization lower bound for a ramsey-type problem, in *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2012, pp. 114-122.

[40] N. Misra, G. Philip, and V. Raman, The kernelization complexity of connected domination in graphs with (no) small cycles, *Algorithmica*, vol. 68, no. 2, pp. 504-530, 2014.

Qilong Feng received the PhD degree in computer science from Central South University, China, in 2010. Currently, he is a lecturer at School of Information Science and Engineering, Central South University, China. His current research interests include computer algorithms and parameterized algorithms.

Qian Zhou is a master candidate in computer science at School of Information Science and Engineering, Central South University, China. Her current research interests include computer algorithms and parameterized algorithms.

Wenjun Li is a PhD candidate in computer science at School of Information Science and Engineering, Central South University, China. His current research interests include computer algorithms and parameterized algorithms.

Jianxin Wang received the PhD degree in computer science from Central South University, China, in 2001. Currently, he is a professor at School of Information Science and Engineering, Central South University, China. His current research interests include algorithm analysis and optimization, computer network, and bioinformatics. He has published more than 100 papers in various international journals and refereed conferences. He is serving as the program committee chair or member of several international conferences. He is a senior member of Institute of Electrical and Electronics Engineers (IEEE).