Parameterized View of Coalition Resource Games

Yuk Hei Chan
Rajesh Chitnis
Kanthi K. Sarpatwar

December 1, 2010

Part I

A Primer on Parameterized Complexity

Classical View : An example

- Given a data set with conflicts, remove as few data points as possible to obtain a conflicts free data set

Classical View : An example

- Given a data set with conflicts, remove as few data points as possible to obtain a conflicts free data set - Vertex Cover Problem.

Classical View : An example

- Given a data set with conflicts, remove as few data points as possible to obtain a conflicts free data set - Vertex Cover Problem.
- Unfortunately, this problem is NP Complete. Classical Complexity tells us that in general there is no good algorithm.

Parameterized View

- What if the solution size is small? Is there an efficient algorithm?

Parameterized View

- What if the solution size is small? Is there an efficient algorithm?
- Yes - a simple branching algorithm will give us $O^*(2^k)$ (O^* notation suppresses the polynomial component.)

Parameterized View

- What if the solution size is small? Is there an efficient algorithm?
- Yes - a simple branching algorithm will give us $O^*(2^k)$ (O^* notation suppresses the polynomial component.)
  - For every edge, a solution must contain at least one end point.

Parameterized View

- What if the solution size is small? Is there an efficient algorithm?
- Yes - a simple branching algorithm will give us $O^*(2^k)$ (O^* notation suppresses the polynomial component.)
  - For every edge, a solution must contain at least one end point.
  - If G is the original graph and $e = (u, v)$ then G has a vertex cover of size k if and only if either $G - u$ or $G - v$ has a vertex cover of size $k - 1$.

Parameterized View

- What if the solution size is small? Is there an efficient algorithm?
- Yes - a simple branching algorithm will give us $O^*(2^k)$ (O^* notation suppresses the polynomial component.)
  - For every edge, a solution must contain at least one end point.
  - If G is the original graph and $e = (u, v)$ then G has a vertex cover of size k if and only if either $G - u$ or $G - v$ has a vertex cover of size $k - 1$.
  - Continuing in this fashion, we obtain a simple $O^*(2^k)$ algorithm.

Parameterized View

- What if the solution size is small? Is there an efficient algorithm?
- Yes - a simple branching algorithm will give us $O^*(2^k)$ (O^* notation suppresses the polynomial component.)
  - For every edge, a solution must contain at least one end point.
  - If G is the original graph and $e = (u, v)$ then G has a vertex cover of size k if and only if either $G - u$ or $G - v$ has a vertex cover of size $k - 1$.
  - Continuing in this fashion, we obtain a simple $O^*(2^k)$ algorithm.
- Best Known - $O^*(1.2738^k)$.

Reducing the Size of the Instance.

- Removing the isolated vertices does not affect the solution size. In fact it is not hard to see that we can remove vertices of degree 1 and 2.

Reducing the Size of the Instance.

- Removing the isolated vertices does not affect the solution size. In fact it is not hard to see that we can remove vertices of degree 1 and 2.
- Every vertex of degree $\geq k$ must belong to the solution.

Reducing the Size of the Instance.

- Removing the isolated vertices does not affect the solution size. In fact it is not hard to see that we can remove vertices of degree 1 and 2.
- Every vertex of degree $\geq k$ must belong to the solution.

Theorem

If G' is a graph instance obtained after applying the preprocessing rules and has a vertex cover of size at most k then $|V(G')| \leq \frac{k^2}{3} + k$.

Fixed Parameter Tractability and Kernelization

Fixed Parameter Tractability and Kernelization

Definition (Fixed Parameter Tractability)

Is there an algorithm of form $f(k)p(n)$ - p is a polynomial function and f is *any* arbitrary function.

Fixed Parameter Tractability and Kernelization

Definition (Fixed Parameter Tractability)

Is there an algorithm of form $f(k)p(n)$ - p is a polynomial function and f is *any* arbitrary function.

Definition (Kernelization)

Using polynomial time heuristics can we reduce the size of an instance to a function of its parameter alone. This will yield an algorithm of form $f(k) + p(n)$.

Fixed Parameter Tractability and Kernelization

Definition (Fixed Parameter Tractability)

Is there an algorithm of form $f(k)p(n)$ - p is a polynomial function and f is *any* arbitrary function.

Definition (Kernelization)

Using polynomial time heuristics can we reduce the size of an instance to a function of its parameter alone. This will yield an algorithm of form $f(k) + p(n)$.

Fixed Parameter Tractable $\iff$ Kernelizable.

Possible Scenarios

- Fixed Parameter Tractable - Vertex Cover Problem

Possible Scenarios

- Fixed Parameter Tractable - Vertex Cover Problem
- $\Omega(n^{f(k)})$ - Independent Set Problem.

Possible Scenarios

- Fixed Parameter Tractable - Vertex Cover Problem
- $\Omega(n^{f(k)})$ - Independent Set Problem.
- NP Complete, even when the parameter is fixed - k-colorability.

Independent Set Problem

- Set of vertices with no edges between them -

Independent Set Problem

- Set of vertices with no edges between them - very similar to the Vertex Cover Problem.

Independent Set Problem

- Set of vertices with no edges between them - very similar to the Vertex Cover Problem.
- Also NP Complete. Can we do something similar as in the case of Vertex Cover Problem? -

Independent Set Problem

- Set of vertices with no edges between them - very similar to the Vertex Cover Problem.
- Also NP Complete. Can we do something similar as in the case of Vertex Cover Problem? - Probably Not.

Independent Set Problem

- Set of vertices with no edges between them - very similar to the Vertex Cover Problem.
- Also NP Complete. Can we do something similar as in the case of Vertex Cover Problem? - Probably Not.
- It is W[1] Complete.

Para-NP Complete

- Consider the problem of deciding whether a graph is properly colorable using k-colors.

Para-NP Complete

- Consider the problem of deciding whether a graph is properly colorable using k-colors.
- NP Complete even for $k = 3$.

Para-NP Complete

- Consider the problem of deciding whether a graph is properly colorable using k-colors.
- NP Complete even for $k = 3$.
- Para-NP Complete - NP complete even when the parameter is fixed.

Part II

Coalition Resource Games: Introduction

Terminology

Coalition Resource Game Γ is a $(n+5)$-tuple given by

$$\Gamma = \langle Ag, G, R, G_1, G_2, \ldots, G_n, \mathbf{en}, \mathbf{req} \rangle$$

where :

- $Ag = \{a_1, a_2, \ldots, a_n\}$ is the set of agents
- $G = \{g_1, g_2, \ldots, g_m\}$ is the set of possible goals
- $R = \{r_1, r_2, \ldots, r_t\}$ is the set of resources
- For each $i \in Ag$, we associate a goal set $G_i \subseteq G$.
- $\mathbf{en} : Ag \times R \to \mathbb{N} \cup \{0\}$ is the endowment function
- $\mathbf{req} : G \times R \to \mathbb{N} \cup \{0\}$ is the requirement function

Terminology(Continued)

- For a coalition C of agents,

$$\mathbf{en}(C, r) = \sum_{i \in C} \mathbf{en}(i, r) \quad \forall r \in R$$

- For a set of Goals G',

$$\mathbf{req}(G', r) = \sum_{g \in G'} \mathbf{req}(g, r) \quad \forall r \in R$$

Terminology(Continued)

- G' satisfies agent i if $G_i \cap G' \neq \emptyset$ and satisfies a coalition C if it satisfies every member of C.

Terminology(Continued)

- G' satisfies agent i if $G_i \cap G' \neq \emptyset$ and satisfies a coalition C if it satisfies every member of C.
- G' is feasible for coalition C if $\forall r \in R$ we have $\mathbf{req}(G', r) \leq \mathbf{en}(G', r)$.

Terminology(Continued)

- G' satisfies agent i if $G_i \cap G' \neq \emptyset$ and satisfies a coalition C if it satisfies every member of C.
- G' is feasible for coalition C if $\forall r \in R$ we have $\mathbf{req}(G', r) \leq \mathbf{en}(G', r)$.
- A coalition C is successful if there exists a non-empty set of goals G' that satisfies C and is feasible for it.

Terminology(Continued)

- G' satisfies agent i if $G_i \cap G' \neq \emptyset$ and satisfies a coalition C if it satisfies every member of C.
- G' is feasible for coalition C if $\forall r \in R$ we have $\mathbf{req}(G', r) \leq \mathbf{en}(G', r)$.
- A coalition C is successful if there exists a non-empty set of goals G' that satisfies C and is feasible for it.

Definition

$succ(C) = \{G' \mid G' \subseteq G, G' \neq \emptyset \text{ and } G' \text{ is successful for } C\}$

Coalition Based Problems

1. SUCCESSFUL COALITION (SC)
   - Instance : A CRG Γ and a coalition C
   - Question : Is C successful ?

Coalition Based Problems

1. SUCCESSFUL COALITION (SC)
    - Instance : A CRG Γ and a coalition C
    - Question : Is C successful ?
2. EXISTENCE OF SUCCESSFUL COALITION OF SIZE k (ESCK)
    - Instance : A CRG Γ and an integer k
    - Question : Does there exist a successful coalition of size exactly k ?

Coalition Based Problems

1. SUCCESSFUL COALITION (SC)
   - Instance : A CRG Γ and a coalition C
   - Question : Is C successful ?
2. EXISTENCE OF SUCCESSFUL COALITION OF SIZE k (ESCK)
   - Instance : A CRG Γ and an integer k
   - Question : Does there exist a successful coalition of size exactly k ?
3. MAXIMAL COALITION (MAXC)
   - Instance : A CRG Γ and a coalition C
   - Question : Is every proper superset of C not successful ?

Coalition Based Problems

1. SUCCESSFUL COALITION (SC)
   - Instance : A CRG Γ and a coalition C
   - Question : Is C successful ?
2. EXISTENCE OF SUCCESSFUL COALITION OF SIZE k (ESCK)
   - Instance : A CRG Γ and an integer k
   - Question : Does there exist a successful coalition of size exactly k ?
3. MAXIMAL COALITION (MAXC)
   - Instance : A CRG Γ and a coalition C
   - Question : Is every proper superset of C not successful ?
4. MAXIMAL SUCCESSFUL COALITION (MAXSC)
   - Instance : A CRG Γ and a coalition C
   - Question : Is C successful and every proper superset of C not successful ?

Some Results

	SC	ESCK	MAXC	MAXSC
	NP-complete	NP-hard	co-NP-complete	D^p-complete
$\|G\|$	FPT	FPT	FPT	FPT
$\|C\|$		W[1]-hard	W[1]-hard	W[1]-hard
$\|R\|$	Para-NP-hard		Para-NP-hard	Para-NP-hard
$\|Ag\| + \|R\|$	FPT		FPT	FPT