

Connected Dominating Set in Graphs with (no) Small Cycles

Fixed-Parameter Tractability
Kernelization Complexity

N Misra, G Philip, V Raman, and S Saurabh

The Institute of Mathematical Sciences, Chennai, India

ALGCO (LIRMM)
Montpellier, France
December 2, 2010

Outline

The Connected Dominating Set problem

- Introduction

- Parameterized Complexity

- Our Results

CDS and Girth

- Graphs of Girth 3 or 4 : W-hard

- Graphs of Girth at least 5 : FPT

- Graphs of Girth 5 or 6 : No Polynomial Kernels

- Graphs of Girth at least 7 : Cubic Kernel

Outline

The Connected Dominating Set problem

Introduction

Parameterized Complexity

Our Results

CDS and Girth

Graphs of Girth 3 or 4 : W-hard

Graphs of Girth at least 5 : FPT

Graphs of Girth 5 or 6 : No Polynomial Kernels

Graphs of Girth at least 7 : Cubic Kernel

Dominating Set

Definition

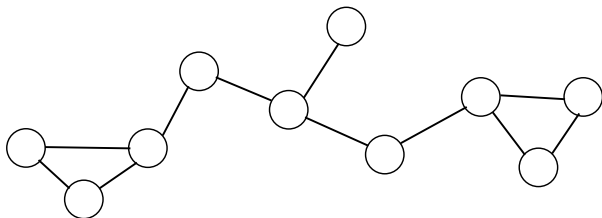
Let $G = (V, E)$ be a graph. A set $S \subseteq V$ of vertices of G is said to be a *dominating set* of G if every vertex which is not in S has at least one neighbour in S .

Dominating Set

Definition

Let $G = (V, E)$ be a graph. A set $S \subseteq V$ of vertices of G is said to be a *dominating set* of G if every vertex which is not in S has at least one neighbour in S .

Example

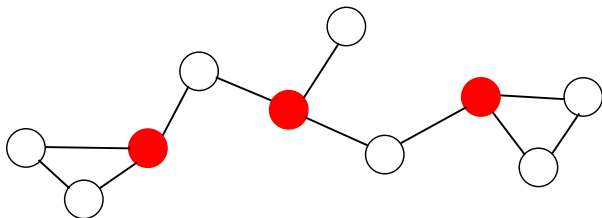


Dominating Set

Definition

Let $G = (V, E)$ be a graph. A set $S \subseteq V$ of vertices of G is said to be a *dominating set* of G if every vertex which is not in S has at least one neighbour in S .

Example



Connected Dominating Set

Definition

Let $G = (V, E)$ be a graph. A set $S \subseteq V$ of vertices of G is said to be a *connected* dominating set of G if

- S is a dominating set of G , and,
- the subgraph $G[S]$ induced by S is connected.

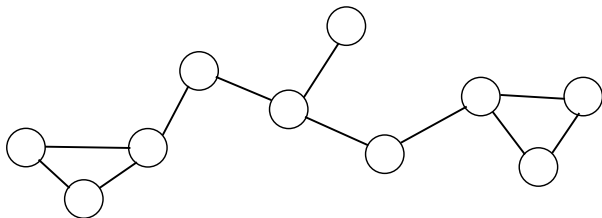
Connected Dominating Set

Definition

Let $G = (V, E)$ be a graph. A set $S \subseteq V$ of vertices of G is said to be a *connected dominating set* of G if

- S is a dominating set of G , and,
- the subgraph $G[S]$ induced by S is connected.

Example



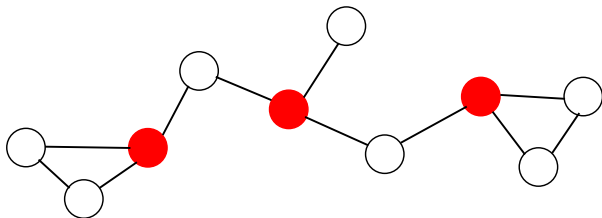
Connected Dominating Set

Definition

Let $G = (V, E)$ be a graph. A set $S \subseteq V$ of vertices of G is said to be a *connected dominating set* of G if

- S is a dominating set of G , and,
- the subgraph $G[S]$ induced by S is connected.

Example



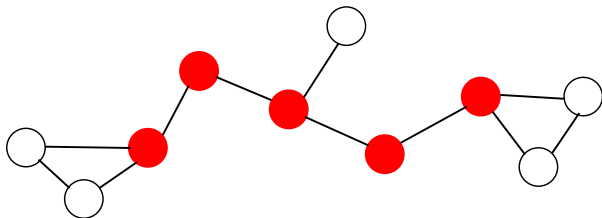
Connected Dominating Set

Definition

Let $G = (V, E)$ be a graph. A set $S \subseteq V$ of vertices of G is said to be a *connected dominating set* of G if

- S is a dominating set of G , and,
- the subgraph $G[S]$ induced by S is connected.

Example



The CDS problem

Definition (The CDS problem)

- Input: A graph G and a positive integer k .
- Question: Does G have a connected dominating set of size at most k ?

Classical Complexity

- The problem is NP-complete, even in planar graphs.
- The best known polynomial-time approximation factor is $\ln(\Delta) + 3$.

The CDS problem

Definition (The CDS problem)

- Input: A graph G and a positive integer k .
- Question: Does G have a connected dominating set of size at most k ?

Classical Complexity

- The problem is NP-complete, even in planar graphs.
- The best known polynomial-time approximation factor is $\ln(\Delta) + 3$.

Outline

The Connected Dominating Set problem

Introduction

Parameterized Complexity

Our Results

CDS and Girth

Graphs of Girth 3 or 4 : W-hard

Graphs of Girth at least 5 : FPT

Graphs of Girth 5 or 6 : No Polynomial Kernels

Graphs of Girth at least 7 : Cubic Kernel

PC: Brief Overview 1/3

- Goal: Solve NP-hard problems in polynomial time . . .
 - . . . when some aspect (the *parameter*) of the input is bounded.
- A parameterized problem instance has two parts:
 - The input itself, and,
 - A *parameter*, usually a number denoted by k .

PC: Brief Overview 1/3

- Goal: Solve NP-hard problems in polynomial time . . .
 - . . . when some aspect (the *parameter*) of the input is bounded.
- A parameterized problem instance has two parts:
 - The input itself, and,
 - A *parameter*, usually a number denoted by k .

PC: Brief Overview 2/3

- A parameterized problem is *fixed-parameter tractable (FPT)* if
 - It can be solved in time $f(k) \cdot n^c$ time, where
 - c is a constant, $f()$ depends only on k .
- A *kernelization algorithm* for a parameterized problem
 - Runs in time polynomial in the input size, and
 - Outputs an **equivalent** instance — a *kernel* — of size bounded by $g(k)$.
- Fact: A parameterized problem is FPT if and only if it has a kernelization algorithm.

PC: Brief Overview 2/3

- A parameterized problem is *fixed-parameter tractable (FPT)* if
 - It can be solved in time $f(k) \cdot n^c$ time, where
 - c is a constant, $f()$ depends only on k .
- A *kernelization algorithm* for a parameterized problem
 - Runs in time polynomial in the input size, and
 - Outputs an **equivalent** instance — a *kernel* — of size bounded by $g(k)$.
- Fact: A parameterized problem is FPT if and only if it has a kernelization algorithm.

PC: Brief Overview 3/3

Given a parameterization of an NP-hard problem, we ask:

- Is the problem FPT?
 - Does it have an algorithm that runs in $f(k) \cdot n^c$ time?
 - There is a hardness theory to show that certain problems are *unlikely* to be FPT.
- If YES, does it have a small kernel?
 - Polynomial-size kernels are good, linear-size even better.
 - There is a hardness theory to show that certain problems are *unlikely* to have polynomial-size kernels.

PC: Brief Overview 3/3

Given a parameterization of an NP-hard problem, we ask:

- Is the problem FPT?
 - Does it have an algorithm that runs in $f(k) \cdot n^c$ time?
 - There is a hardness theory to show that certain problems are *unlikely* to be FPT.
- If YES, does it have a small kernel?
 - Polynomial-size kernels are good, linear-size even better.
 - There is a hardness theory to show that certain problems are *unlikely* to have polynomial-size kernels.

The CDS problem

Parameterized Version

- Input: A graph G and a positive integer k .
- Parameter: k
- Question: Does G have a connected dominating set of size at most k ?

The CDS problem

Parameterized Complexity: Known Results

- $W[2]$ -hard on graphs in general:
 - $O(f(k) \cdot \text{poly}(n))$ -time algorithms are unlikely to exist.
- FPT on graphs of bounded degeneracy (Golovach, Villanger : WG 2008).
 - Planar graphs, graphs of bounded treewidth, ...
- Has a *linear* kernel in apex-minor-free graphs (Fomin, Lokshtanov, Saurabh, Thilikos : SODA 2010).
- *Unlikely* to have polynomial kernels in graphs of bounded degeneracy (Cygan, Pilipczuk, Pilipczuk, Wojtaszczyk : WG 2010).

Outline

The Connected Dominating Set problem

Introduction

Parameterized Complexity

Our Results

CDS and Girth

Graphs of Girth 3 or 4 : W-hard

Graphs of Girth at least 5 : FPT

Graphs of Girth 5 or 6 : No Polynomial Kernels

Graphs of Girth at least 7 : Cubic Kernel

The CDS problem

Parameterized Complexity: Our Results

- We explore what happens when the *girth* of the input graph is bounded.
 - The girth of a graph G is the size (number of edges) of a smallest cycle in G .
 - “Orthogonal” to degeneracy:
 - There are graphs of fixed girth and arbitrarily large degeneracy, and vice versa.

The CDS problem

Parameterized Complexity: Our Results

- We completely characterize the parameterized complexity of CDS for different values of girth:
 - $W[2]$ -hard on graphs of girth 3 or 4.
 - FPT on graphs of girth at least 5.
 - Polynomial kernels unlikely in graphs of girth 5 or 6.
 - Cubic ($O(k^3)$) kernel in graphs of girth at least 7.

Outline

The Connected Dominating Set problem

Introduction

Parameterized Complexity

Our Results

CDS and Girth

Graphs of Girth 3 or 4 : W-hard

Graphs of Girth at least 5 : FPT

Graphs of Girth 5 or 6 : No Polynomial Kernels

Graphs of Girth at least 7 : Cubic Kernel

CDS in Graphs of Girth 3 or 4

W[2]-hardness Reduction

- Reduction from Dominating Set.
- Appears in earlier work of Raman and Saurabh.
- We merely observe that it works for CDS as well ... 😊

CDS in Graphs of Girth 3 or 4

$W[2]$ -hardness Reduction

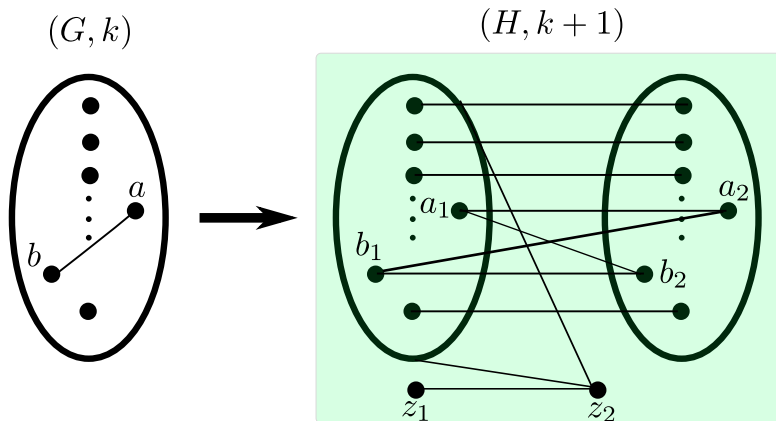


Figure: Reduction from Dominating Set to CDS

Outline

The Connected Dominating Set problem

Introduction

Parameterized Complexity

Our Results

CDS and Girth

Graphs of Girth 3 or 4 : W-hard

Graphs of Girth at least 5 : FPT

Graphs of Girth 5 or 6 : No Polynomial Kernels

Graphs of Girth at least 7 : Cubic Kernel

CDS in Graphs of Girth at least 5

FPT Algorithm: Outline

- The girth bound implies a degree-rule:
 - Every vertex of "high" degree must be in *any* small CDS (in fact, in any small *DS*).
- If there are too many high-degree vertices, then say NO.
- If the number of **undominated** vertices is too much, then say NO.
- The number of vertices that could possibly be part of a solution is bounded:
 - Guess a minimal DS from among these
 - Find a minimum-size Steiner tree for this DS.

CDS in Graphs of Girth at least 5

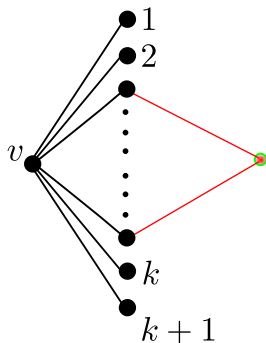
FPT Algorithm: The Degree Rule

- If a vertex v has degree more than k , then v must be in every dominating set of size at most k .

CDS in Graphs of Girth at least 5

FPT Algorithm: The Degree Rule

- If a vertex v has degree more than k , then v must be in every dominating set of size at most k .



CDS in Graphs of Girth at least 5

FPT Algorithm: The Degree Rule

- If a vertex v has degree more than k , then v must be in every dominating set of size at most k .
- We use colours for keeping track of the state.
 - **Red** vertices are in the dominating set that we construct.
 - Vertices dominated by the red ones are **green**.
 - Undominated vertices are **blue**.
- **Reduction Rule 0**: Colour all vertices **blue**.

CDS in Graphs of Girth at least 5

FPT Algorithm: The Degree Rule

- If a vertex v has degree more than k , then v must be in every dominating set of size at most k .
- **Reduction Rule 1:** If a vertex v has degree more than k :
 - Colour v **red**.
 - Colour all neighbours of v **green**.

CDS in Graphs of Girth at least 5

FPT Algorithm: The Degree Rule

- If a vertex v has degree more than k , then v must be in every dominating set of size at most k .
- **Reduction Rule 1:** If a vertex v has degree more than k :
 - Colour v **red**.
 - Colour all neighbours of v **green**.
- **Reduction Rule 2:** If there are more than $k(k+1)$ **blue** vertices left, say **NO**:
 - k vertices of degree at most k can dominate at most $k(k+1)$ **blue** vertices.

CDS in Graphs of Girth at least 5

FPT Algorithm: The Degree Rule

- If a vertex v has degree more than k , then v must be in every dominating set of size at most k .
- **Reduction Rule 1:** If a vertex v has degree more than k :
 - Colour v **red**.
 - Colour all neighbours of v **green**.
- **Reduction Rule 2:** If there are more than $k(k+1)$ **blue** vertices left, say **NO**:
- **Reduction Rule 3:** If there are more than k **red** vertices, say **NO**.

CDS in Graphs of Girth at least 5

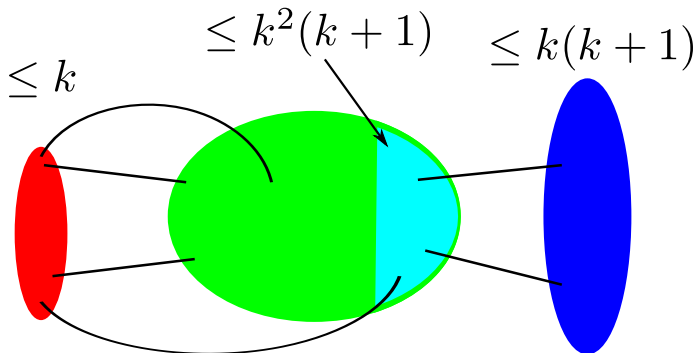
The FPT Algorithm

- There are at most
 - k red vertices.
 - $k(k + 1)$ blue vertices.
 - $k^2(k + 1)$ neighbours of blue vertices.

CDS in Graphs of Girth at least 5

The FPT Algorithm

- There are at most
 - k red vertices.
 - $k(k+1)$ blue vertices.
 - $k^2(k+1)$ neighbours of blue vertices.



CDS in Graphs of Girth at least 5

The FPT Algorithm

- There are at most
 - k red vertices.
 - $k(k + 1)$ blue vertices.
 - $k^2(k + 1)$ neighbours of blue vertices.
- Every inclusion-minimal **dominating set** of size at most k is contained in these $k + k(k + 1) + k^2(k + 1)$ vertices, say \mathcal{S} .

CDS in Graphs of Girth at least 5

The FPT Algorithm

- There are at most
 - k red vertices.
 - $k(k + 1)$ blue vertices.
 - $k^2(k + 1)$ neighbours of blue vertices.
- Every inclusion-minimal dominating set of size at most k is contained in these $k + k(k + 1) + k^2(k + 1)$ vertices, say \mathcal{S} .
- Go over all minimal dominating sets $D \subseteq \mathcal{S}$ of size at most k .

CDS in Graphs of Girth at least 5

The FPT Algorithm

- There are at most
 - k red vertices.
 - $k(k+1)$ blue vertices.
 - $k^2(k+1)$ neighbours of blue vertices.
- Every inclusion-minimal dominating set of size at most k is contained in these $k + k(k+1) + k^2(k+1)$ vertices, say \mathcal{S} .
- Go over all minimal dominating sets $D \subseteq \mathcal{S}$ of size at most k .
 - This can be done in $k^{O(k)}$ time.

CDS in Graphs of Girth at least 5

The FPT Algorithm

- There are at most
 - k red vertices.
 - $k(k+1)$ blue vertices.
 - $k^2(k+1)$ neighbours of blue vertices.
- Every inclusion-minimal dominating set of size at most k is contained in these $k + k(k+1) + k^2(k+1)$ vertices, say \mathcal{S} .
- Go over all minimal dominating sets $D \subseteq \mathcal{S}$ of size at most k .
- For each D , check if there is a Steiner tree on at most k vertices connecting D .
- This can be done in $2^{O(k)}$ time (Nederlof, ICALP 2009).

CDS in Graphs of Girth at least 5

The FPT Algorithm

- There are at most
 - k red vertices.
 - $k(k+1)$ blue vertices.
 - $k^2(k+1)$ neighbours of blue vertices.
- Every inclusion-minimal dominating set of size at most k is contained in these $k + k(k+1) + k^2(k+1)$ vertices, say \mathcal{S} .
- Go over all minimal dominating sets $D \subseteq \mathcal{S}$ of size at most k .
- For each D , check if there is a Steiner tree on at most k vertices connecting D .
- A CDS of the graph, of size at most k , if it exists, can thus be found in $O^*(k^{O(k)})$ time.

Outline

The Connected Dominating Set problem

Introduction

Parameterized Complexity

Our Results

CDS and Girth

Graphs of Girth 3 or 4 : W-hard

Graphs of Girth at least 5 : FPT

Graphs of Girth 5 or 6 : No Polynomial Kernels

Graphs of Girth at least 7 : Cubic Kernel

Kernel Lower Bound Machinery

- We make use of kernel lower bound techniques developed by
 - Bodlaender, Downey, Fellows, Hermelin : *JCSS* 2009
 - Bodlaender, Thomassé, Yeo : *ESA* 2009
- These techniques allow us to conclude that a problem does not have polynomial kernels . . .
 - *unless* $NP \subseteq CoNP/Poly$
 - or (weaker) PH collapses to the third level

Kernel Lower Bound Machinery

- We make use of kernel lower bound techniques developed by
 - Bodlaender, Downey, Fellows, Hermelin : *JCSS* 2009
 - Bodlaender, Thomassé, Yeo : *ESA* 2009
- These techniques allow us to conclude that a problem does not have polynomial kernels . . .
 - *unless* $NP \subseteq CoNP/Poly$
 - or (weaker) PH collapses to the third level

Kernel Lower Bound Machinery

Tools : Composition

Composition Algorithm

A composition algorithm for a parameterized problem

- takes as input a *sequence* $\langle (x_1, k), (x_2, k), \dots, (x_t, k) \rangle$
- runs in time polynomial in $\sum_{i=1}^t |x_i| + k$
- outputs an instance (y, k') :
 - (y, k') is YES iff **at least one** (x_i, k) is YES,
 - k' is polynomial in k .

The composition algorithm acts like an **OR** gate.

Kernel Lower Bound Machinery

Tools : Composition

Theorem

If a parameterized NP-complete problem has a composition algorithm, then it does not have polynomial kernels

... unless $NP \subseteq CoNP/Poly$.

Example

- k -path: Does graph G have a path of length at least k ?
- NP-hard (Hamiltonian Path)
- With k as the parameter, is compositional: take the disjoint union of all input graphs.



Kernel Lower Bound Machinery

Tools : Composition

Theorem

If a parameterized NP-complete problem has a composition algorithm, then it does not have polynomial kernels

... unless $NP \subseteq CoNP/Poly$.

Example

- k -path: Does graph G have a path of length at least k ?
- NP-hard (Hamiltonian Path)
- With k as the parameter, is compositional: take the disjoint union of all input graphs.

Kernel Lower Bound Machinery

Tools : Composition

Theorem

If a parameterized NP-complete problem has a composition algorithm, then it does not have polynomial kernels

... unless $NP \subseteq CoNP/Poly$.

Example

- k -path: Does graph G have a path of length at least k ?
- NP-hard (Hamiltonian Path)
- With k as the parameter, is compositional: take the disjoint union of all input graphs.

Kernel Lower Bound Machinery

Tools : Reduction

- Let \mathcal{A}, \mathcal{B} be parameterized problems.

Definition (Polynomial Parameter Transformation)

A *polynomial parameter transformation* (**PPT**) from \mathcal{A} to \mathcal{B} is a function f such that:

- f is polynomial-time computable.
- If $f(x, k) = (x', k')$, then
 - $(x, k) \in \mathcal{A} \iff (x', k') \in \mathcal{B}$.
 - $k' \leq p(k)$ for some polynomial p .

Kernel Lower Bound Machinery

Tools : Reduction

Theorem

Let \mathcal{A}, \mathcal{B} be parameterized problems where \mathcal{A} is NP-complete and \mathcal{B} is in NP. Suppose there is a PPT from \mathcal{A} to \mathcal{B} . Then, if \mathcal{B} has a polynomial kernel, then \mathcal{A} also has a polynomial kernel.



Kernel Lower Bound Machinery

Tools : Summary

- Composition: **OR**-like algorithm, implies no poly-kernel unless . . .
- Reduction: In polynomial time, and at most polynomial increase in the parameter. Helps propagate no-poly-kernel bounds.

CDS in Graphs of Girth 5 or 6

Kernel Lower Bound

- We could not devise a composition algorithm
 - A composition algorithm is sometimes *very* hard to find
 - ... and may be it doesn't exist (we don't know).
- We invented an intermediate problem, Fair Connected Colours (FCC).
 - It's a *snap* to compose FCC.
- We found a PPT (reduction) from FCC to CDS in graphs of girth 5 or 6.
 - Not easy either, but it could be done ☺.



CDS in Graphs of Girth 5 or 6

Kernel Lower Bound

- We could not devise a composition algorithm
 - A composition algorithm is sometimes *very* hard to find
 - ... and may be it doesn't exist (we don't know).
- We invented an intermediate problem, Fair Connected Colours (FCC).
 - It's a snap to compose FCC.
- We found a PPT (reduction) from FCC to CDS in graphs of girth 5 or 6.
 - Not easy either, but it could be done ☺.



CDS in Graphs of Girth 5 or 6

Kernel Lower Bound

- We could not devise a composition algorithm
 - A composition algorithm is sometimes *very* hard to find
 - ... and may be it doesn't exist (we don't know).
- We invented an intermediate problem, Fair Connected Colours (FCC).
 - It's a snap to compose FCC.
- We found a PPT (reduction) from FCC to CDS in graphs of girth 5 or 6.
 - Not easy either, but it could be done ☺.



CDS in Graphs of Girth 5 or 6

Fair Connected Colours (FCC)

- Input: A graph G whose vertices are *properly* coloured with k colours such that all neighbours of each vertex have distinct colours.
- Parameter: k
- Question: Does G contain a tree T on k vertices (as a subgraph) where all vertices in T have different colours?

CDS in Graphs of Girth 5 or 6

Fair Connected Colours (FCC)

- FCC is NP-complete
 - Reduction from SAT
- FCC is compositional
 - Just take disjoint union!
- So from the earlier theorem, we have:
 - FCC has no polynomial kernel unless $P = NP$.

CDS in Graphs of Girth 5 or 6

Fair Connected Colours (FCC)

- FCC is NP-complete
 - Reduction from SAT
- FCC is compositional
 - Just take disjoint union!
- So from the earlier theorem, we have:
 - FCC has no polynomial kernel unless $P = NP$

CDS in Graphs of Girth 5 or 6

Fair Connected Colours (FCC)

- FCC is NP-complete
 - Reduction from SAT
- FCC is compositional
 - Just take disjoint union!
- So from the earlier theorem, we have:
 - FCC has no polynomial kernel unless . . .

CDS in Graphs of Girth 5 or 6

Reduction from FCC : Rough Picture

- Given an instance (G, k) of FCC, we construct an instance $(H, k^2 + k)$ of CDS such that:
 - H has girth 6.
 - A colourful tree in G becomes a connected dominating set in H :
 - The one vertex in each colour class dominates (well, sort of) all of that class.
 - The underlying tree provides connectivity.
 - And vice versa.
 - The $+k^2$ is spent for increasing the girth, and for domination.
 - The modification for girth 5 is not difficult.

CDS in Graphs of Girth 5 or 6

Reduction from FCC : Rough Picture

- Given an instance (G, k) of FCC, we construct an instance $(H, k^2 + k)$ of CDS such that:
 - H has girth 6.
 - A colourful tree in G becomes a connected dominating set in H :
 - The one vertex in each colour class dominates (well, sort of) all of that class.
 - The underlying tree provides connectivity.
 - And vice versa.
 - The $+k^2$ is spent for increasing the girth, and for domination.
 - The modification for girth 5 is not difficult.

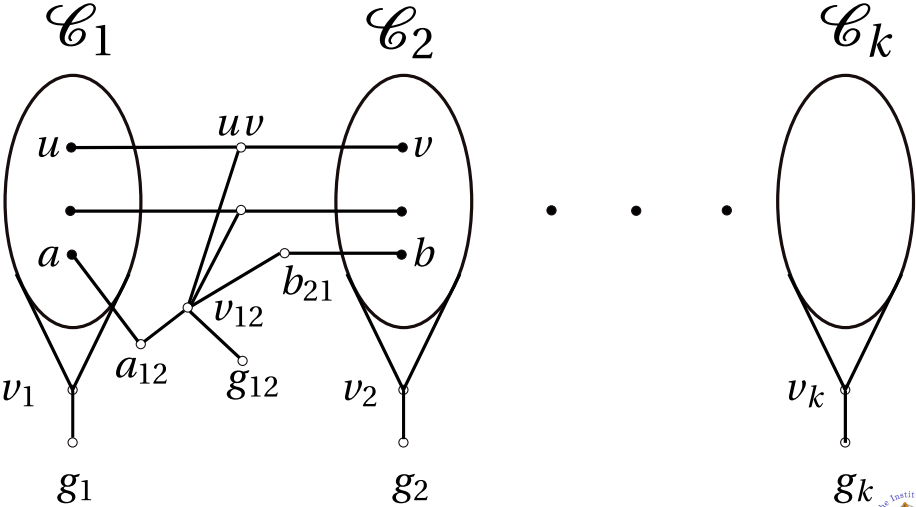
CDS in Graphs of Girth 5 or 6

Reduction from FCC : Construction

- Add a global vertex v_i to each colour class C_i .
- Attach a pendant (guard) vertex g_i to each v_i .

CDS in Graphs of Girth 5 or 6

Reduction from FCC : Construction



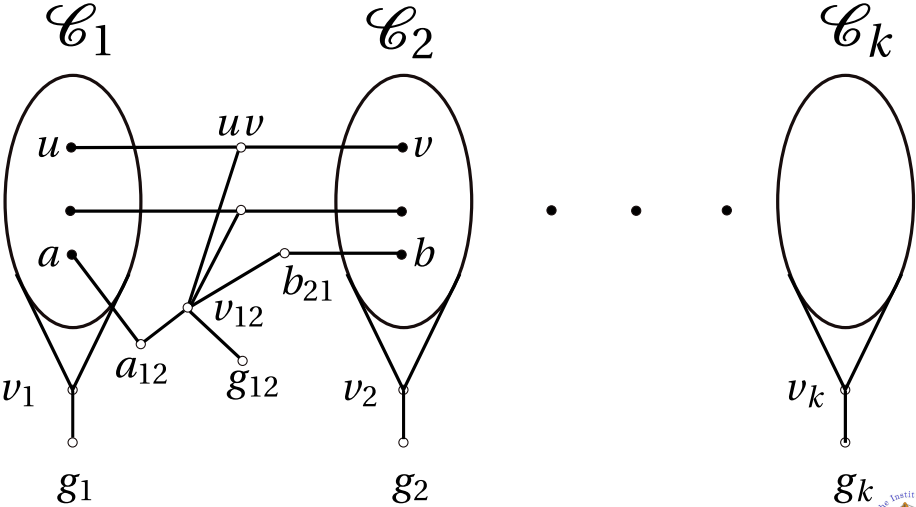
CDS in Graphs of Girth 5 or 6

Reduction from FCC : Construction

- Add a global vertex v_i to each colour class C_i .
- Attach a pendant (guard) vertex g_i to each v_i .
- Subdivide each edge $\{u, v\}$ of G with a new vertex uv .

CDS in Graphs of Girth 5 or 6

Reduction from FCC : Construction



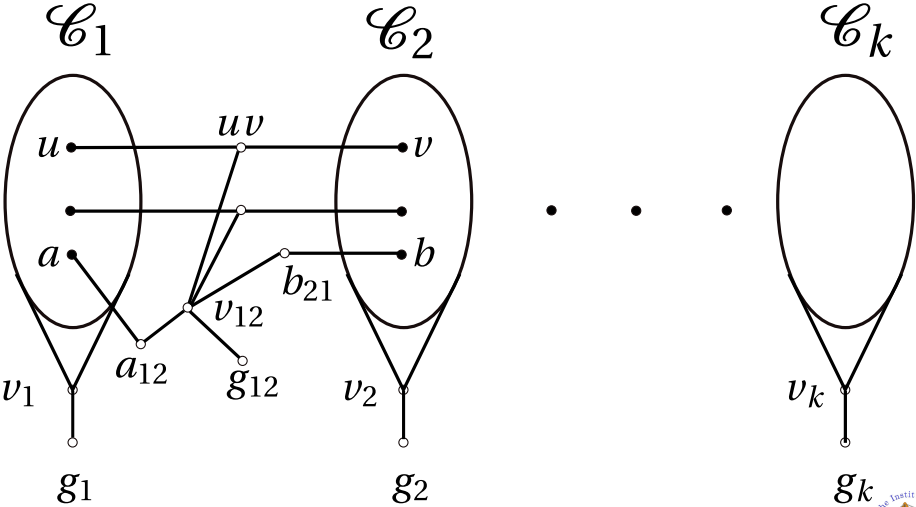
CDS in Graphs of Girth 5 or 6

Reduction from FCC : Construction

- Add a global vertex v_i to each colour class C_i .
- Attach a pendant (guard) vertex g_i to each v_i .
- Subdivide each edge $\{u, v\}$ of G with a new vertex uv .
- For every two colour classes C_i, C_j ,
 - Add two new vertices v_{ij}, g_{ij} and the edge $\{v_{ij}, g_{ij}\}$

CDS in Graphs of Girth 5 or 6

Reduction from FCC : Construction



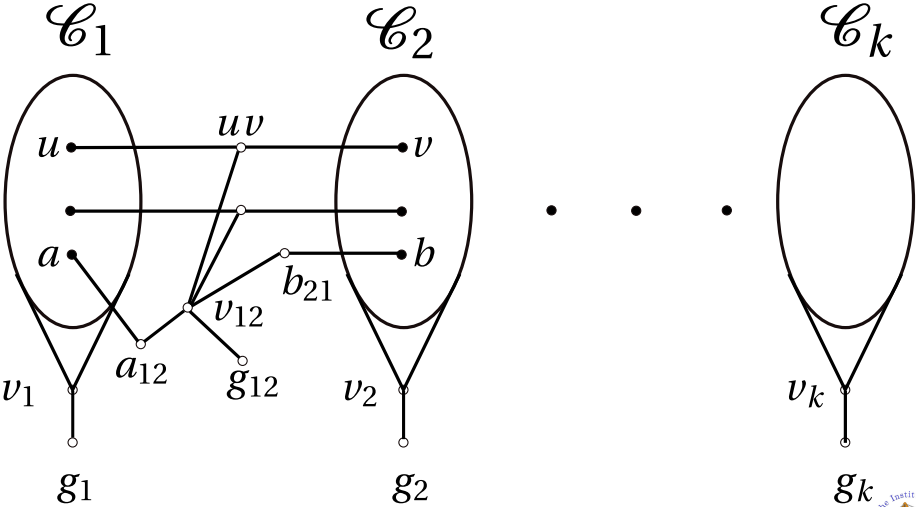
CDS in Graphs of Girth 5 or 6

Reduction from FCC : Construction

- Add a global vertex v_i to each colour class \mathcal{C}_i .
- Attach a pendant (guard) vertex g_i to each v_i .
- Subdivide each edge $\{u, v\}$ of G with a new vertex uv .
- For every two colour classes $\mathcal{C}_i, \mathcal{C}_j$,
 - Add two new vertices v_{ij}, g_{ij} and the edge $\{v_{ij}, g_{ij}\}$
 - For each edge $\{u, v\}; u \in \mathcal{C}_i, v \in \mathcal{C}_j$ in G , add the edge $\{uv, v_{ij}\}$.

CDS in Graphs of Girth 5 or 6

Reduction from FCC : Construction



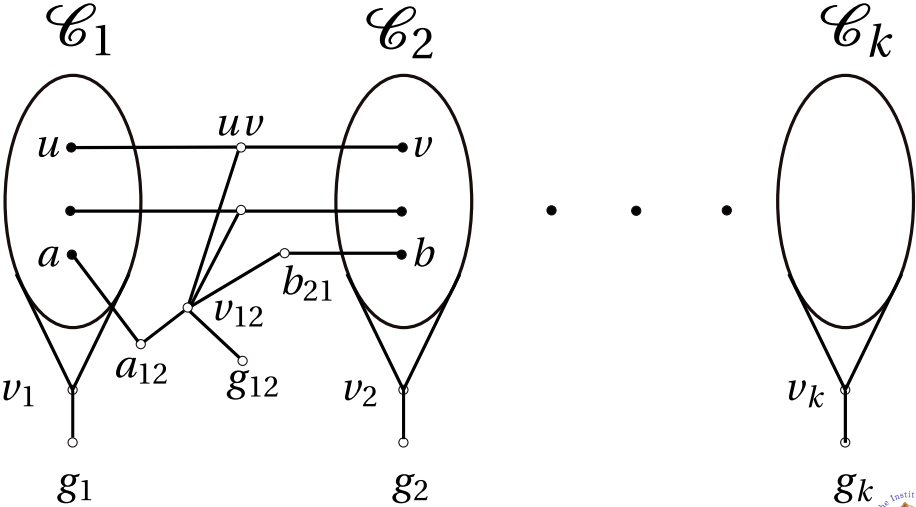
CDS in Graphs of Girth 5 or 6

Reduction from FCC : Construction

- Add a global vertex v_i to each colour class \mathcal{C}_i .
- Attach a pendant (guard) vertex g_i to each v_i .
- Subdivide each edge $\{u, v\}$ of G with a new vertex uv .
- For every two colour classes $\mathcal{C}_i, \mathcal{C}_j$,
 - Add two new vertices v_{ij}, g_{ij} and the edge $\{v_{ij}, g_{ij}\}$
 - For each edge $\{u, v\}; u \in \mathcal{C}_i, v \in \mathcal{C}_j$ in G , add the edge $\{uv, v_{ij}\}$.
 - If $u \in \mathcal{C}_i$ has no neighbour in \mathcal{C}_j , add a new vertex u_{ij} and the edges $\{u, u_{ij}\}, \{u_{ij}, v_{ij}\}$

CDS in Graphs of Girth 5 or 6

Reduction from FCC : Construction



CDS in Graphs of Girth 5 or 6

Reduction from FCC : Correctness

Forward direction

- To a colourful tree on k vertices in G we add:
 - The $k - 1$ “edge” vertices;
 - The k global vertices for the colour classes;
 - The $\binom{k}{2}$ vertices v_{ij} , and,
 - And $\binom{k}{2} - (k - 1)$ vertices to dominate the vertices v_{ij}
- ... to get a CDS of H on $k^2 + k$ vertices.

CDS in Graphs of Girth 5 or 6

Reduction from FCC : Correctness

Reverse direction

- Any CDS of G must contain:
 - The k vertices v_i , and the $\binom{k}{2}$ vertices v_{ij}
 - Domination constraint
 - And at least one (distinct) neighbour of each of these vertices
 - Connectivity constraint
 - This uses up the budget of $k^2 + k!$
 - The vertices v_i and v_{ij} are all leaves in the CDS.

CDS in Graphs of Girth 5 or 6

Reduction from FCC : Correctness

Reverse direction

- Prune the leaves v_i and v_{ij} of the CDS
- The remaining part T'
 - is connected,
 - contains exactly one vertex in each colour class, and,
 - contains exactly $\binom{k}{2} + k$ vertices
- For each v_{ij} ,
 - If its neighbour in T' is a leaf, delete it.
 - Otherwise, it is one of the “edge” vertices: contract it
- This leaves a connected graph T on exactly k vertices, one from each colour class.

CDS in Graphs of Girth 5 or 6

Summary

- Fair Connected Colours is NP-complete and compositional.
- Fair Connected Colours reduces to Connected Dominating Set
 - In polynomial time, and with polynomial increase in the parameter.
- These together imply that Connected Dominating Set is not likely to have polynomial kernels.

Outline

The Connected Dominating Set problem

Introduction

Parameterized Complexity

Our Results

CDS and Girth

Graphs of Girth 3 or 4 : W-hard

Graphs of Girth at least 5 : FPT

Graphs of Girth 5 or 6 : No Polynomial Kernels

Graphs of Girth at least 7 : Cubic Kernel

CDS in Graphs of Girth at Least 7

Cubic Kernel: Outline

- The main reduction rule is the same degree rule as before.
- As before, we use colours for keeping track of the state:
 - **Red** vertices are in the dominating set that we construct.
 - Vertices dominated by the red ones are **green**.
 - Undominated vertices are **blue**.
- The larger girth (7) allows us to bound the number of **green** vertices as well.
 - In girth-5 graphs, we could only bound the number of **green** vertices *which were adjacent to blue* vertices.

CDS in Graphs of Girth at Least 7

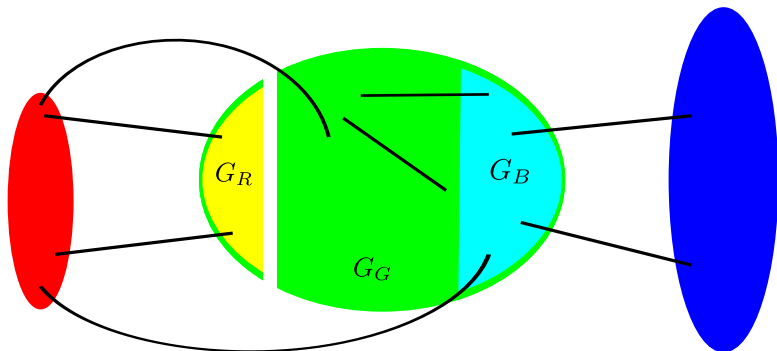
Cubic Kernel: Reduction Rules

- **Reduction Rule 0:** Colour all vertices blue.
- **Reduction Rule 1:** If $\deg(v) > k$, colour v red and all neighbours of v green.
- **Reduction Rule 2:** If there are more than k red vertices or more than $k(k + 1)$ blue vertices left, say **NO**.
- **Reduction Rule 3:** Say **NO** if there is an isolated blue vertex.
- **Reduction Rule 4:** If u is a pendant blue or green vertex adjacent to a vertex v :
 - Remove u .
 - Colour v red.
 - Colour all blue neighbours of v green.

CDS in Graphs of Girth at Least 7

Cubic Kernel: Bounding Green Vertices, Part 1

- Divide green vertices into three:
 - G_B : has *at least one* blue neighbour.
 - G_R : has *only* red neighbours.
 - G_G : no blue neighbour, at least one green neighbour.



CDS in Graphs of Girth at Least 7

Cubic Kernel: Bounding Green Vertices, Part 1

- Divide **green** vertices into three:
 - G_B : has *at least one* **blue** neighbour.
 - G_R : has *only* **red** neighbours.
 - G_G : no **blue** neighbour, at least one **green** neighbour.
- $|G_B| \leq k \cdot |B| \leq k(k^2 + k)$
 - The degree rule.
- $|G_R| \leq \binom{|R|}{2} \leq \binom{k}{2}$
 - The pendant rule, and no two vertices have more than one common neighbour.

CDS in Graphs of Girth at Least 7

Cubic Kernel: Bounding Green Vertices, Part 2

- Bounding $|G_G|$
 - **Green** vertices which have **green** neighbours.
 - Suffices to bound the number of *edges* E_G with *both* end points **green**.
 - $|G_G| \leq 2 \cdot |E_G|$
 - Each **green** vertex is adjacent to some **red** vertex.
 - For any pair x, y of **red** vertices, there “corresponds” at most one edge $\{u, v\} \in E_G$.
 - Otherwise there is a 6-cycle!
 - $|G_G| \leq 2 \cdot |E_G| \leq 2 \cdot \binom{|R|}{2} \leq 2 \cdot \binom{k}{2}$

CDS in Graphs of Girth at Least 7

Cubic Kernel: Summary

- At most k red vertices.
- At most $k(k + 1)$ blue vertices.
- At most $k(k^2 + k) + 3 \cdot \binom{k}{2}$ green vertices.
- A coloured kernel on at most $k^3 + \frac{7}{2}k^2 + \frac{k}{2}$ vertices.
- To get a “plain” kernel:
 - Attach a new pendant vertex to each red vertex.
 - Remove all colours.
 - A kernel for CDS on at most $k^3 + \frac{7}{2}k^2 + \frac{3}{2}k$ vertices.

Recapitulating ...

- Connected Dominating Set parameterized by solution size k is:
 - W[2]-hard on graphs in general.
 - W[2]-hard on graphs of girth at most 4.
 - FPT on graphs of girth at least 5.
 - No polynomial kernel (...) in graphs of girth 5 or 6.
 - A cubic ($O(k^3)$) kernel in graphs of girth at least 7.

Je vous remercie de votre attention!