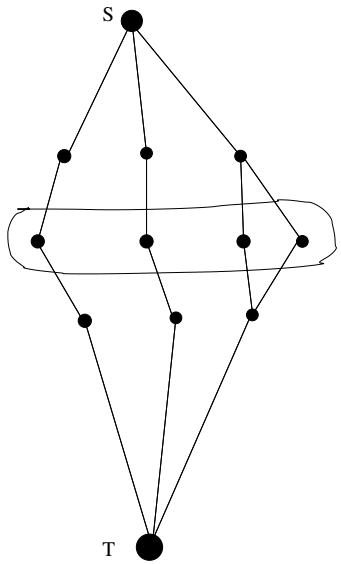
Treewidth reduction and algorithmic applications

Content

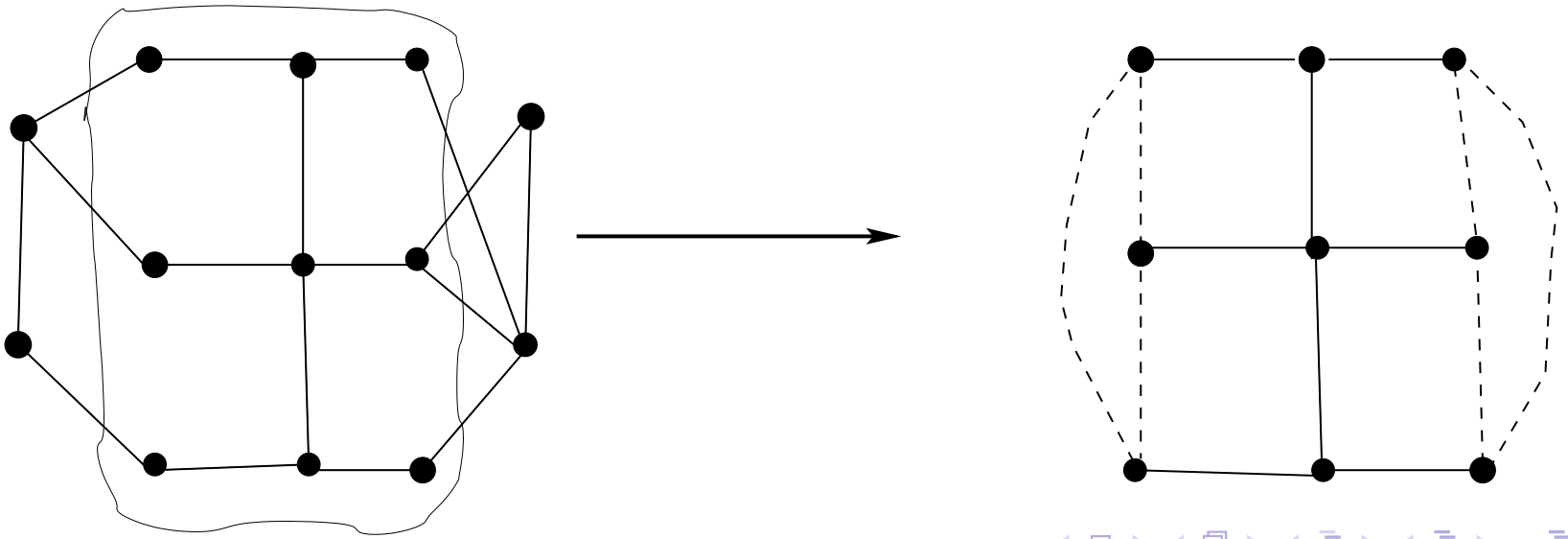
Minimal $s - t$ separator

- Given graph G and two vertices s and t.
- S is a separator if it disconnects s and t.
- Minimality: no proper subset of S disconnects s and t.



Torso graph

- Given a graph G and a set C of vertices.
- Take $G[C]$.
- Add edges between vertices adjacent to connected components of $G \setminus C$.
- The resulting graph is called $torso(G, C)$.



Treewidth reduction theorem

- Given graph G, two specified vertices s and t, an integer k.

Treewidth reduction theorem

- Given graph G, two specified vertices s and t, an integer k.
- Let C be the union of all *minimal* $s - t$ separators of size at most k.

Treewidth reduction theorem

- Given graph G, two specified vertices s and t, an integer k.
- Let C be the union of all *minimal* $s - t$ separators of size at most k.
- The graph $torso(G, C)$ has a treewidth bounded by a function of k.

Treewidth reduction theorem

- Given graph G, two specified vertices s and t, an integer k.
- Let C be the union of all *minimal* $s - t$ separators of size at most k.
- The graph $torso(G, C)$ has a treewidth bounded by a function of k.
- **Remark.** Minimality is essential: otherwise C will include all the vertices.

Constrained separation problems

- A classic problem: given graph G and vertices s, t what is the size of the smallest $s - t$ separator?
- Solvable in polynomial time by network flow techniques.

Constrained separation problems

- A classic problem: given graph G and vertices s, t what is the size of the smallest $s - t$ separator?
- Solvable in polynomial time by network flow techniques.
- When constraints are added on the separator, the problem usually becomes NP-hard.
- Example (stable cut problem): find a smallest $s - t$ separator S such that $G[S]$ is an independent set.
- We parameterize the problem by the size of the separator.

Treewdith reduction for the stable cut problem

- Finding a stable cut of size at most k is the same as to find a *minimal* stable cut of size at most k.

Treewdith reduction for the stable cut problem

- Finding a stable cut of size at most k is the same as to find a *minimal* stable cut of size at most k.
- Let C be the union of all minimal separators plus $\{s, t\}$.

Treewdith reduction for the stable cut problem

- Finding a stable cut of size at most k is the same as to find a *minimal* stable cut of size at most k.
- Let C be the union of all minimal separators plus $\{s, t\}$.
- By the treewdith reduction theorem $G^* = \mathit{torso}(G, C)$ has a treewidth bounded by a function of k.

Treewdith reduction for the stable cut problem

- Finding a stable cut of size at most k is the same as to find a *minimal* stable cut of size at most k.
- Let C be the union of all minimal separators plus $\{s, t\}$.
- By the treewdith reduction theorem $G^* = torso(G, C)$ has a treewidth bounded by a function of k.
- By a cosmetic modification, minimal $s - t$ separators in G and G^* induce *the same* graphs.

Treewdith reduction for the stable cut problem

- Finding a stable cut of size at most k is the same as to find a *minimal* stable cut of size at most k.
- Let C be the union of all minimal separators plus $\{s, t\}$.
- By the treewdith reduction theorem $G^* = torso(G, C)$ has a treewidth bounded by a function of k.
- By a cosmetic modification, minimal $s - t$ separators in G and G^* induce *the same* graphs.
- That is, **instead of solving the problem for G, we solve the problem for G^*.**
- A fixed-parameter algorithm immediately follows from Courcelle theorem.

A more general result

- Instead of being without edges $G[S]$ can belong to an arbitrary class that is:
  - Hereditary
  - Solvable

A more general result

- Instead of being without edges $G[S]$ can belong to an arbitrary class that is:
  - Hereditary
  - Solvable
- Being hereditary is needed to enable taking *minimal* separators.

A more general result

- Instead of being without edges $G[S]$ can belong to an arbitrary class that is:
  - Hereditary
  - Solvable
- Being hereditary is needed to enable taking *minimal* separators.
- Being solvable is needed to explicitly encode all graphs of size at most k in the formula.

A more general result

- Instead of being without edges $G[S]$ can belong to an arbitrary class that is:
  - Hereditary
  - Solvable
- Being hereditary is needed to enable taking *minimal* separators.
- Being solvable is needed to explicitly encode all graphs of size at most k in the formula.
- The formula will get *huge* but still bounded by a function of k.

Bipartization problems

- Stable bipartization problem: given a graph G, is it possible to remove an independent set of size at most (or exactly) k so that the resulting graph is bipartite?
- Put it differently: is the graph 3-colorable so that one of the color classes is of size at most (exactly) k?

Bipartization problems

- Stable bipartization problem: given a graph G, is it possible to remove an independent set of size at most (or exactly) k so that the resulting graph is bipartite?
- Put it differently: is the graph 3-colorable so that one of the color classes is of size at most (exactly) k?
- Was open since 2001 and received a considerable attention from the researchers.
- FPT by transformation from the stable separation.
- The result can be generalized to an arbitrary hereditary and solvable class.

Summary of consequences

- We propose a framework that allows an easy fixed-parameter tractability proof for a large class of problems.

Summary of consequences

- We propose a framework that allows an easy fixed-parameter tractability proof for a large class of problems.
- It allowed us to solve at once 4 seemingly unrelated open problems scattered in the literature.

Summary of consequences

- We propose a framework that allows an easy fixed-parameter tractability proof for a large class of problems.
- It allowed us to solve at once 4 seemingly unrelated open problems scattered in the literature.
- The resulting algorithms are impractical due to to a superexponential dependence on k.

Proof cases

- The size of the smallest $s - t$ separator is larger than k (degenerated case).
- The size of the smallest $s - t$ separator is exactly k (the basic case).
- The size of the smallest $s - t$ separator is larger than k (inductive pumping of the basic case).

The degenerated case

- Assume that the smallest $s - t$ separator is of size larger than k.
- Then the union of minimal $s - t$ separators of size at most k is the empty set.
- Clearly, its treewidth is bounded.

The basic case

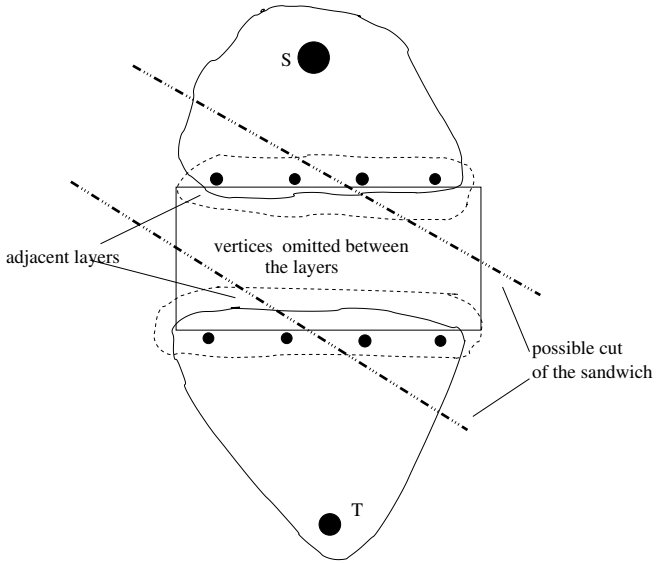
- Smallest $s - t$ separator is of size exactyly k.
- The vertices of smallest $s - t$ separators can be organised into 'layers' of size k so that there are edges only between adjacent layers.
- Not only the treewidth but the pathwidth of this graph is bounded.

The main case: notational preparation

- Let $r < k$ be the smallest size of a $s - t$ separator.
- Assume that for a vertex v a smallest minimal $s - t$ separator including v is of size r'.
- We say that the *excess* of v is $r' - r$.

What happens inside the sandwich

- Consider the vertices sandwiched between two adjacent layers as a result of torso operation.
- **Key fact:** vertices of excess x participate in a minimal separator of size $r + x$ cutting the sandwich.



Catching bugs in the sandwich

- Vertices of smallest size separators are spent to building the layers.
- Then, to cut the sandwich, at least $r + 1$ vertices will be needed.
- The layerisation for all possible sandwich cuts will catch all vertices of excess 1.

Pumped sandwich including vertices of excess 1

- To include the rest of the vertices of the desired set, more pumping iterations are performed.
- The number of iterations is at most $k - r$, so the treewdith can be bounded by a function of k.

Concluding remarks

- Given a graph G, two vertices s, t, and an integer k we can construct a graph G^*:

Concluding remarks

- Given a graph G, two vertices s, t, and an integer k we can construct a graph G^*:
  - With the treewidth bounded by a function of k.
  - All minimal $s - t$ separators of size at most k in G remain such ones in G^*.
  - The adjacency relation between vertices of these separators is preserved.

Concluding remarks

- Given a graph G, two vertices s, t, and an integer k we can construct a graph G^*:
  - With the treewidth bounded by a function of k.
  - All minimal $s - t$ separators of size at most k in G remain such ones in G^*.
  - The adjacency relation between vertices of these separators is preserved.
- Instead of solving a constrained separation problem on G, it can be solved on G^*.
- Fixed-parameter tractability immediately follows using standard techniques.
- The result is a general methodology for establishing of fixed-parameter tractability.