Coq/Ssreflect for large case-based graph theory proofs

Ricardo Katz (CIFASIS-CONICET)

Daniel Severín (UNR, CONICET)

Coq Workshop 2021
July 2, 2021

This talk is about:

- a work in progress about a graph parameter, which is the “weighted version” of a well-known parameter from graph theory,

This talk is about:

- a work in progress about a graph parameter, which is the “weighted version” of a well-known parameter from graph theory,
- the formalization of some novel results, first as a personal challenge, but also as an experiment to see if it is feasible to use proof assistants to complement pen-and-paper proofs,

This talk is about:

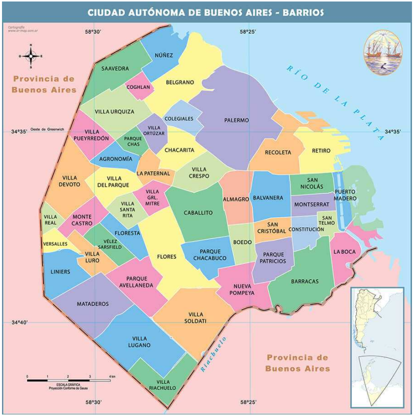
- a work in progress about a graph parameter, which is the "weighted version" of a well-known parameter from graph theory,
- the formalization of some novel results, first as a personal challenge, but also as an experiment to see if it is feasible to use proof assistants to complement pen-and-paper proofs,
- exploring the possibility of generating certificates about the value of the parameter for a given instance.

This talk is about:

- a work in progress about a graph parameter, which is the "weighted version" of a well-known parameter from graph theory,
- the formalization of some novel results, first as a personal challenge, but also as an experiment to see if it is feasible to use proof assistants to complement pen-and-paper proofs,
- exploring the possibility of generating certificates about the value of the parameter for a given instance.

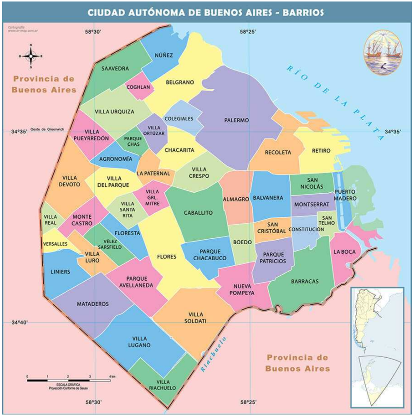
Disclaimer: we are by no means experts in the field

An example



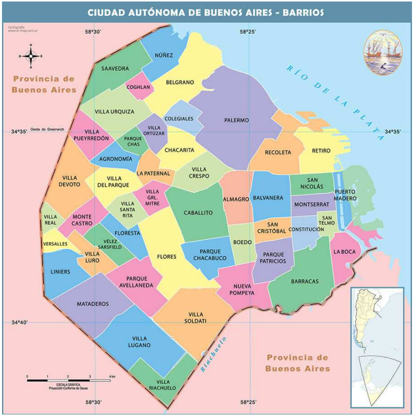
- Consider a city divided into areas where a service wants to be provided (e.g., mobile phone connection).

An example



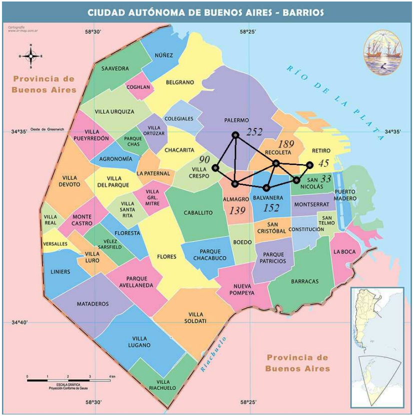
- Consider a city divided into areas where a service wants to be provided (e.g., mobile phone connection).
- Assume the service is properly provided to the area in which it is located and its neighbors.

An example



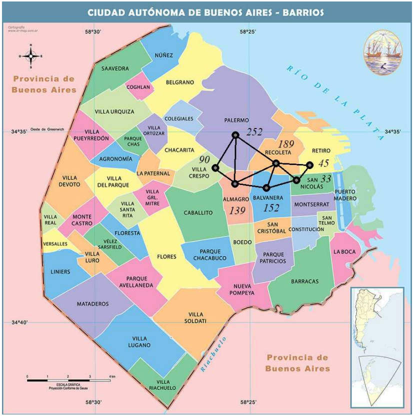
- Consider a city divided into areas where a service wants to be provided (e.g., mobile phone connection).
- Assume the service is properly provided to the area in which it is located and its neighbors.
- If each allocation of the service has a fixed high cost, no allocation to an area is desired if all the areas covered by this allocation are covered by other allocated areas.

An example

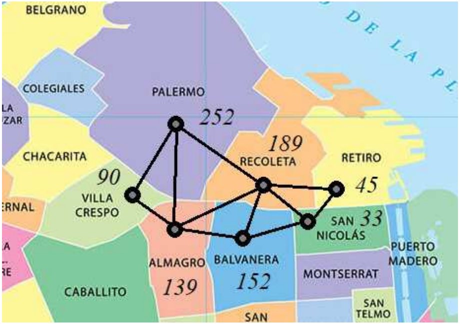


- The city is modeled by a graph whose vertices are the areas and there is an edge between two vertices if the corresponding areas are neighbors.

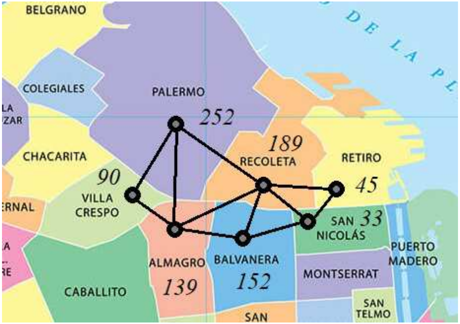
An example



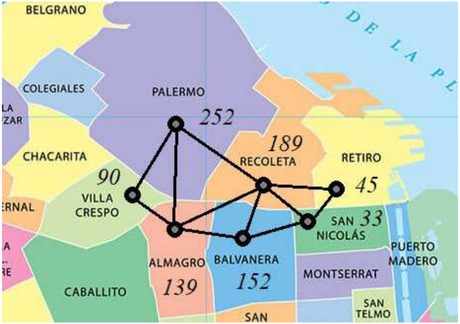
- The city is modeled by a graph whose vertices are the areas and there is an edge between two vertices if the corresponding areas are neighbors.
- There is a value associated with each area representing the profit of installing the service there.



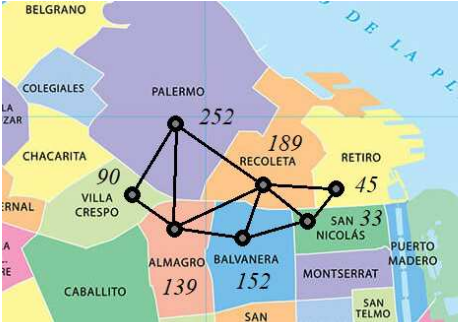
- It is desired to maximize the sum of these profits



- It is desired to maximize the sum of these profits, but...
  - ...if it is required that all areas are covered
    $\rightarrow$ Weighted Upper Domination Set (WUDS) $\Gamma_w(G)$
  - ...if it may leave some areas with no (or poor) service
    $\rightarrow$ Maximum Weighted Irredundant Set (MWIS) $IR_w(G)$

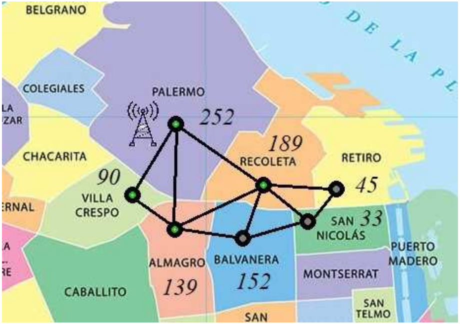


- It is desired to maximize the sum of these profits, but...
  - ...if it is required that all areas are covered
    → Weighted Upper Domination Set (WUDS) $\Gamma_w(G)$
  - ...if it may leave some areas with no (or poor) service
    → Maximum Weighted Irredundant Set (MWIS) $IR_w(G)$ ✓



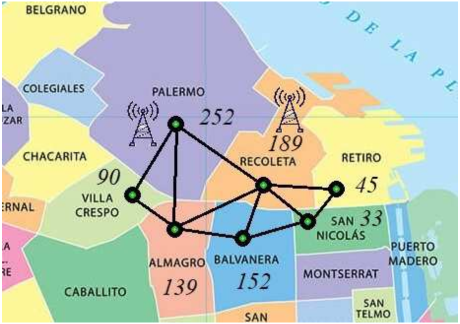
- It is desired to maximize the sum of these profits, but...
  - ...if it is required that all areas are covered
    $\rightarrow$ Weighted Upper Domination Set (WUDS) $\Gamma_w(G)$
  - ...if it may leave some areas with no (or poor) service
    $\rightarrow$ Maximum Weighted Irredundant Set (MWIS) $IR_w(G)$ ✓

- Since WUDS is more constrained that MWIS, $\Gamma_w(G) \leq IR_w(G)$



- It is desired to maximize the sum of these profits, but...
  - ...if it is required that all areas are covered
    $\rightarrow$ Weighted Upper Domination Set (WUDS) $\Gamma_w(G)$
  - ...if it may leave some areas with no (or poor) service
    $\rightarrow$ Maximum Weighted Irredundant Set (MWIS) $IR_w(G)$ ✓

- Since WUDS is more constrained that MWIS, $\Gamma_w(G) \leq IR_w(G)$



- It is desired to maximize the sum of these profits, but...
  - ...if it is required that all areas are covered
    $\rightarrow$ Weighted Upper Domination Set (WUDS) $\Gamma_w(G)$
  - ...if it may leave some areas with no (or poor) service
    $\rightarrow$ Maximum Weighted Irredundant Set (MWIS) $IR_w(G)$ ✓
- Since WUDS is more constrained that MWIS, $\Gamma_w(G) \leq IR_w(G)$

Introduction

Basic definitions

- $G = (V, E) \leftarrow$ simple graph

Introduction

Basic definitions

- $G = (V, E) \leftarrow$ simple graph
- $v \in V$
  - $N(v) \doteq$ vertices adjacent to v — open neighborhood
  - $N[v] \doteq N(v) \cup \{v\}$ — closed neighborhood

Introduction

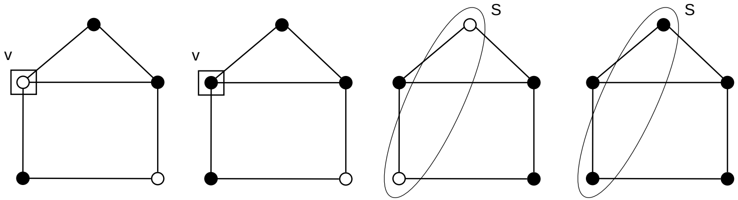
Basic definitions

- $G = (V, E) \leftarrow$ simple graph
- $v \in V$
  - $N(v) \doteq$ vertices adjacent to v — open neighborhood
  - $N[v] \doteq N(v) \cup \{v\}$ — closed neighborhood
- $S \subseteq V$
  - $N(S) \doteq \cup_{v \in V} N(v)$
  - $N[S] \doteq \cup_{v \in V} N[v]$

Introduction

Basic definitions

- $G = (V, E) \leftarrow$ simple graph
- $v \in V$
  - $N(v) \doteq$ vertices adjacent to v — open neighborhood
  - $N[v] \doteq N(v) \cup \{v\}$ — closed neighborhood
- $S \subseteq V$
  - $N(S) \doteq \cup_{v \in V} N(v)$
  - $N[S] \doteq \cup_{v \in V} N[v]$



N(v) N[v] N(S) N[S]

Stable (or independent) sets

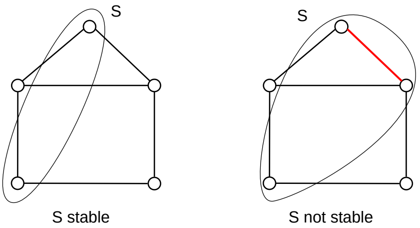
A set $S \subseteq V$ is *stable* if no vertex in S is adjancent to any other in S

- S *stable* $\iff N(S) \cap S = \emptyset$

Stable (or independent) sets

A set $S \subseteq V$ is *stable* if no vertex in S is adjancent to any other in S

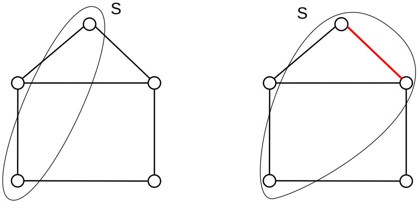
- S *stable* $\iff N(S) \cap S = \emptyset$



Stable (or independent) sets

A set $S \subseteq V$ is *stable* if no vertex in S is adjancent to any other in S

- S *stable* $\iff N(S) \cap S = \emptyset$



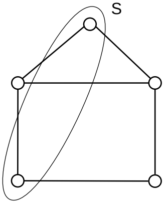
S stable S not stable

$\alpha(G)$ = stable set of maximum size

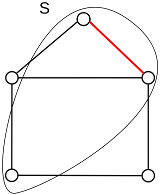
Stable (or independent) sets

A set $S \subseteq V$ is *stable* if no vertex in S is adjancent to any other in S

- S *stable* $\iff N(S) \cap S = \emptyset$



S stable



S not stable

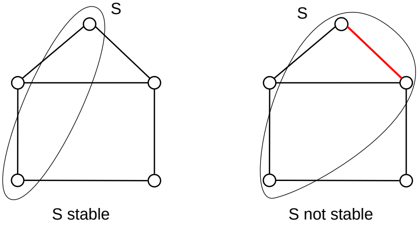
$\alpha(G) =$ stable set of maximum size

- $w(v) =$ weight of vertex v

Stable (or independent) sets

A set $S \subseteq V$ is *stable* if no vertex in S is adjancent to any other in S

- S *stable* $\iff N(S) \cap S = \emptyset$



$\alpha(G)$ = stable set of maximum size

- $w(v)$ = weight of vertex v
- $w(S) = \sum_{v \in V} w(v)$ weight of set S

$\alpha_w(G)$ = stable set of maximum weight

Irredundant sets

- u *dominates* $v \longleftrightarrow u = v$ or u and v adjacent

Irredundant sets

- *u dominates v* $\longleftrightarrow$ $u = v$ or u and v adjacent

> A set $D \subseteq V$ is *irredundant* if each vertex of D dominates at least one vertex not dominated by any other from D

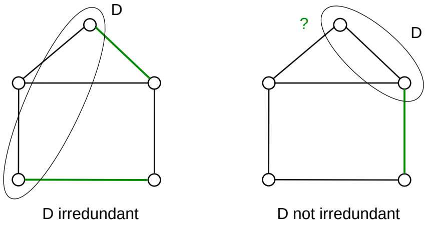
- D *irredundant* $\iff \forall\ v \in D,\ \ N[v] - N[D - \{v\}] \neq \emptyset$

Irredundant sets

- *u dominates v* $\longleftrightarrow$ $u = v$ or u and v adjacent

A set $D \subseteq V$ is *irredundant* if each vertex of D dominates at least one vertex not dominated by any other from D

- *D irredundant* $\iff \forall\ v \in D,\ \ N[v] - N[D - \{v\}] \neq \emptyset$

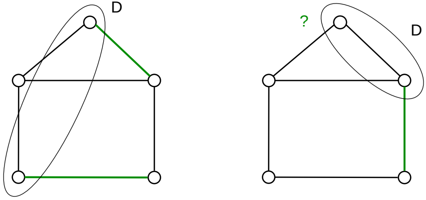


Irredundant sets

- *u dominates v* $\longleftrightarrow$ $u = v$ or u and v adjacent

> A set $D \subseteq V$ is *irredundant* if each vertex of D dominates at least one vertex not dominated by any other from D

- *D irredundant* $\Longleftrightarrow \forall\ v \in D,\ \ N[v] - N[D - \{v\}] \neq \emptyset$



D irredundant   D not irredundant

- $IR(G)$ = irredundant set of maximum size
- $IR_w(G)$ = irredundant set of maximum weight

Dependencies used in the formalization

Requirements: Coq 8.11+, MathComp 1.10+
graph-theory 0.8 [Doczkal, Pous 2020]

Dependencies used in the formalization

Requirements: Coq 8.11+, MathComp 1.10+
graph-theory 0.8 [Doczkal, Pous 2020]

- In particular, the definitions of parameters α_w and IR_w were previously formalized in `graph-theory` library [S.- 2019]:

```
Variable G : sgraph.
Variable weight : G → ℕ.
Definition weight_set S := \sum_(v ∈ S) weight v.
Definition α_w := weight_set (arg_max ∅ stable weight_set).
Definition IR_w := weight_set (arg_max ∅ irredundant weight_set).
```

(G denotes the graph and its set of vertices as well, due to a coercion)

Our work

An upper bound

$$IR_w(G) \leq w(V) - \delta_w(G),$$
$$\text{where } \delta_w(G) \doteq \min\{w(N[v] \setminus \{u\}) : v \in V,\ u \in N[v]\}$$

(particular case: $\delta_1(G)$ = minimum degree of G)

Our work

An upper bound

$IR_w(G) \leq w(V) - \delta_w(G),$
$$\text{where } \delta_w(G) \doteq \min\{w(N[v] \setminus \{u\}) : v \in V,\ u \in N[v]\}$$

(particular case: $\delta_1(G)$ = minimum degree of G)

```
Let M n := (G == ∅) || [∃v, [∃u, (u ∈ N[v]) && (W(N[v] :\ u) == n)]].

Fact exM : ∃ n : ℕ, M n.

Definition δ_w := ex_minn exM.

Fact delta_w_min u v : (u ∈ N[v]) → δ_w ≤ W (N[v] :\ u).

Lemma bound_weight_irredundant S :
  (irredundant S) → W S ≤ W G − δ_w G.
Proof. (12 lines) Qed.

Lemma IR_w_leq_V_minus_delta_w : IR_w G weight ≤ W G − δ_w G.
```

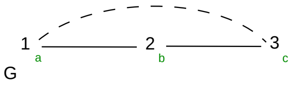
Our work

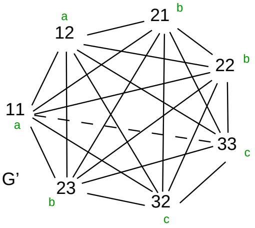
- u `-*-` $v \doteq u$ dominates v

Our work

- $u \text{ -*- } v \doteq u$ dominates v
  - $V' \doteq \{uv : u \in V,\ u \text{ -*- } v\}$,
  - $E' \doteq \{(uv, zr) : uv, zr \in V',\ uv \neq zr,\ v \text{ -*- } z \vee r \text{ -*- } u\}$,
  - $w'_{uv} \doteq w_u$ for all $uv \in V'$.
- $(G' = (V', E'), w')$

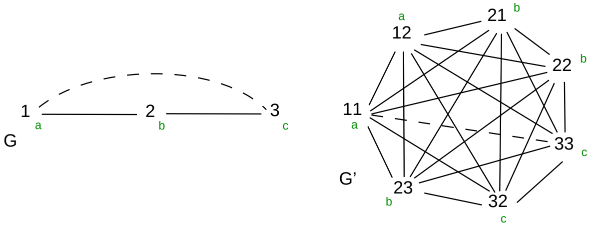
- $u \text{ -*- } v \doteq u$ dominates v
- $(\color{green}{G'} = (\color{blue}{V'}, \color{red}{E'}), \color{purple}{w'})$
  - $V' \doteq \{uv : u \in V,\ u \text{ -*- } v\}$,
  - $E' \doteq \{(uv, zr) : uv, zr \in V',\ uv \neq zr,\ v \text{ -*- } z \vee r \text{ -*- } u\}$,
  - $w'_{uv} \doteq w_u$ for all $uv \in V'$.





Our work

- $u \text{ -*- } v \doteq u$ dominates v
- $(G' = (V', E'), w')$
  - $V' \doteq \{uv : u \in V,\ u \text{ -*- } v\}$,
  - $E' \doteq \{(uv, zr) : uv, zr \in V',\ uv \neq zr,\ v \text{ -*- } z \vee r \text{ -*- } u\}$,
  - $w'_{uv} \doteq w_u$ for all $uv \in V'$.



A transformation between MWIS and Max. Weighted Stable Set Problem

$IR_w(G) = \alpha_{w'}(G')$

Our work

- u -*- $v \doteq u$ dominates v
- $(G' = (V', E'), w')$
  - $V' \doteq \{uv : u \in V,\ u$ -*- $v\}$,
  - $E' \doteq \{(uv, zr) : uv, zr \in V',\ uv \neq zr,\ v$ -*- $z \vee r$ -*- $u\}$,
  - $w'_{uv} \doteq w_u$ for all $uv \in V'$.

```
Inductive trfgraph_vert_type :=
  TrfGraphVert (x : G * G) of (x.1 -*- x.2).

Definition trfgraph_rel := [rel x y : trfgraph_vert_type | (x != y)
  && ((y.1 -*- x.2) || (y.2 -*- x.1))].

Definition trfgraph := SGraph trfgraph_sym trfgraph_irrefl.

Let G' := trfgraph G.
Let weight' := λ x : G', weight x.1.

Theorem IR_w_G_is_alpha_w_G' : IR_w G weight = alpha_w G' weight'.
```

Our work

- u -*- $v \doteq u$ dominates v
- $(G' = (V', E'), w')$
  - $V' \doteq \{uv : u \in V,\ u \text{ -*- } v\}$,
  - $E' \doteq \{(uv, zr) : uv, zr \in V',\ uv \neq zr,\ v \text{ -*- } z \vee r \text{ -*- } u\}$,
  - $w'_{uv} \doteq w_u$ for all $uv \in V'$.

```
Inductive trfgraph_vert_type :=
  TrfGraphVert (x : G * G) of (x.1 -*- x.2).

Definition trfgraph_rel := [rel x y : trfgraph_vert_type | (x != y)
  && ((y.1 -*- x.2) || (y.2 -*- x.1))].

Definition trfgraph := SGraph trfgraph_sym trfgraph_irrefl.

Let G' := trfgraph G.
Let weight' := λ x : G', weight x.1.

Theorem IR_w_G_is_alpha_w_G' : IR_w G weight = alpha_w G' weight'.
```

(300 total lines)

Our work

- $G \dot{\subset} H \doteq G$ is an induced subgraph of H

Basic properties of induced subgraphs

- $S \subset V(G) \implies G[S] \dot{\subset} G$
- $G_1 \dot{\subset} G_2 \wedge G_2 \dot{\subset} G_3 \implies G_1 \dot{\subset} G_3$

Our work

- $G \dot{\subset} H \doteq G$ is an induced subgraph of H

Basic properties of induced subgraphs

- $S \subset V(G) \implies G[S] \dot{\subset} G$
- $G_1 \dot{\subset} G_2 \wedge G_2 \dot{\subset} G_3 \implies G_1 \dot{\subset} G_3$

Properties between subgraphs of transformed graphs

- $G \dot{\subset} G'$
- $G \dot{\subset} H \implies G' \dot{\subset} H'$

Our work

- $G \dot{\subset} H \doteq G$ is an induced subgraph of H

Basic properties of induced subgraphs

- $S \subset V(G) \implies G[S] \dot{\subset} G$
- $G_1 \dot{\subset} G_2 \wedge G_2 \dot{\subset} G_3 \implies G_1 \dot{\subset} G_3$

```
Definition induced_subgraph (G1 G2 : sgraph) :=
  exists2 h : G1 → G2, injective h & induced_hom h.

Notation "A ⊂̇ B" := (induced_subgraph A B).

Lemma induced_S_is_sub G (S : {set G}) : (induced S) ⊂̇ G.

Lemma subgraph_trans G1 G2 G3 : G1 ⊂̇ G2 → G2 ⊂̇ G3 → G1 ⊂̇ G3.
```

Our work

- $G \dot{\subset} H \doteq G$ is an induced subgraph of H

Basic properties of induced subgraphs

- $S \subset V(G) \implies G[S] \dot{\subset} G$
- $G_1 \dot{\subset} G_2 \wedge G_2 \dot{\subset} G_3 \implies G_1 \dot{\subset} G_3$

```
Definition induced_subgraph (G1 G2 : sgraph) :=
  exists2 h : G1 → G2, injective h & induced_hom h.

Notation "A ⊂̇ B" := (induced_subgraph A B).

Lemma induced_S_is_sub G (S : {set G}) : (induced S) ⊂̇ G.

Lemma subgraph_trans G1 G2 G3 : G1 ⊂̇ G2 → G2 ⊂̇ G3 → G1 ⊂̇ G3.
```

(would be great if some properties about induced subgraphs could be added to graph-theory library 😉)

Our work

- $G \dot{\subset} H \doteq G$ is an induced subgraph of H

Basic properties of induced subgraphs

- $S \subset V(G) \implies G[S] \dot{\subset} G$
- $G_1 \dot{\subset} G_2 \wedge G_2 \dot{\subset} G_3 \implies G_1 \dot{\subset} G_3$

Properties between subgraphs of transformed graphs

- $G \dot{\subset} G'$
- $G \dot{\subset} H \implies G' \dot{\subset} H'$

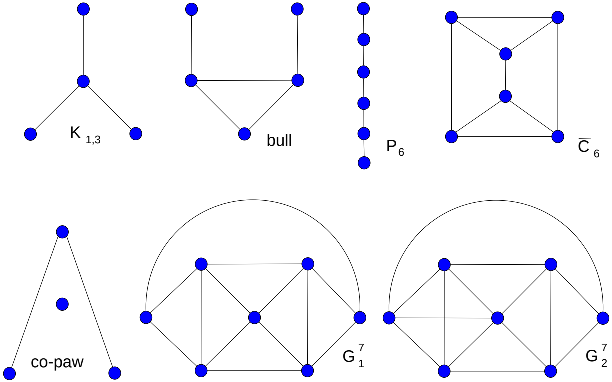
```
Lemma subgraph_G_G' G : G Ċ trfgraph G.

Lemma trfgraph_subgraph G H :
  G Ċ H → trfgraph G Ċ trfgraph H.
```

(70 total lines)

Our work

Main results (used for complexity results of MWIS)

- $[K_{1,3}]$: $K_{1,3} \dot{\subset} G \;\vee\; \text{bull} \dot{\subset} G \;\vee\; P_6 \dot{\subset} G \;\vee\; \overline{C_6} \dot{\subset} G \iff K_{1,3} \dot{\subset} G'$
- [co-paw]: $\text{co-paw} \dot{\subset} G \;\vee\; G_1^7 \dot{\subset} G \;\vee\; G_2^7 \dot{\subset} G \iff \text{co-paw} \dot{\subset} G'$



The easy part

Main results (used for complexity results of MWIS)

- $[K_{1,3}]$: $K_{1,3} \dot{\subset} G \vee$ bull$\dot{\subset} G$ $\vee P_6 \dot{\subset} G \vee \overline{C_6} \dot{\subset} G \iff K_{1,3} \dot{\subset} G'$
- [co-paw]: co-paw$\dot{\subset} G \vee G_1^7 \dot{\subset} G \vee G_2^7 \dot{\subset} G \iff$ co-paw$\dot{\subset} G'$

$\Longrightarrow$

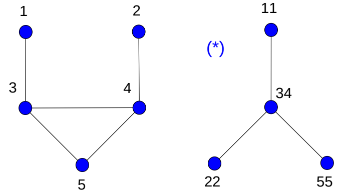
The easy part

Main results (used for complexity results of MWIS)

- $[K_{1,3}]$: $K_{1,3} \dot{\subset} G \;\vee\;$ $\color{red}{\text{bull} \dot{\subset} G}$ $\;\vee\; P_6 \dot{\subset} G \;\vee\; \overline{C_6} \dot{\subset} G \iff$ $\color{purple}{K_{1,3} \dot{\subset} G'}$
- $[\text{co-paw}]$: $\text{co-paw} \dot{\subset} G \;\vee\; G_1^7 \dot{\subset} G \;\vee\; G_2^7 \dot{\subset} G \iff \text{co-paw} \dot{\subset} G'$

$\Longrightarrow$

$$\left.\begin{array}{rcl} \color{blue}{(*)} & \longrightarrow & K_{1,3} \dot{\subset} \text{bull}' \\ \color{red}{\text{bull} \dot{\subset} G} & \Longrightarrow & \text{bull}' \dot{\subset} G' \end{array}\right\} \Longrightarrow \color{purple}{K_{1,3} \dot{\subset} G'}$$



bull

the $K_{1,3}$ in bull'

The easy part

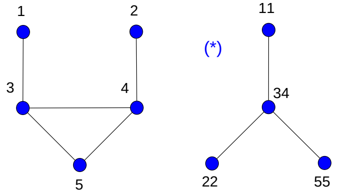
Main results (used for complexity results of MWIS)

- $[K_{1,3}]$: $K_{1,3} \dot{\subset} G \;\vee\; \text{bull} \dot{\subset} G \;\vee\; P_6 \dot{\subset} G \;\vee\; \overline{C_6} \dot{\subset} G \iff K_{1,3} \dot{\subset} G'$
- $[\text{co-paw}]$: $\text{co-paw} \dot{\subset} G \;\vee\; G_1^7 \dot{\subset} G \;\vee\; G_2^7 \dot{\subset} G \iff \text{co-paw} \dot{\subset} G'$

$\Longrightarrow$

$$\left.\begin{array}{rcl} (*) & \longrightarrow & K_{1,3} \dot{\subset} \text{bull}' \\ \text{bull} \dot{\subset} G & \Longrightarrow & \text{bull}' \dot{\subset} G' \end{array}\right\} \Longrightarrow K_{1,3} \dot{\subset} G'$$

- $[K_{1,3}]$: 70 lines
- $[\text{co-paw}]$: 40 lines



bull　　the $K_{1,3}$ in bull'

The hard part

Main results (used for complexity results of MWIS)

- $[K_{1,3}]$: $K_{1,3} \dot{\subset} G \ \vee \ \text{bull} \dot{\subset} G \ \vee \ P_6 \dot{\subset} G \ \vee \ \overline{C_6} \dot{\subset} G \iff K_{1,3} \dot{\subset} G'$
- [co-paw]: $\text{co-paw} \dot{\subset} G \ \vee \ G_1^7 \dot{\subset} G \ \vee \ G_2^7 \dot{\subset} G \iff \text{co-paw} \dot{\subset} G'$

$\Longleftarrow$

The hard part

Main results (used for complexity results of MWIS)

- $[K_{1,3}]$: $K_{1,3} \dot{\subset} G \;\vee\; \text{bull} \dot{\subset} G \;\vee\; P_6 \dot{\subset} G \;\vee\; \overline{C_6} \dot{\subset} G \iff K_{1,3} \dot{\subset} G'$
- [co-paw]: $\text{co-paw} \dot{\subset} G \;\vee\; G_1^7 \dot{\subset} G \;\vee\; G_2^7 \dot{\subset} G \iff \text{co-paw} \dot{\subset} G'$

$\Longleftarrow$

- Based on a large number of cases that makes it almost impossible to present a proper pen-and-paper version of them. Indeed, due to the number of cases, it is difficult to be sure that all the possible cases are considered and that each of them is tackled properly $\longrightarrow$ formalization

The hard part

Main results (used for complexity results of MWIS)

- $[K_{1,3}]$: $K_{1,3} \dot\subset G \;\vee\; \text{bull} \dot\subset G \;\vee\; P_6 \dot\subset G \;\vee\; \overline{C_6} \dot\subset G \iff K_{1,3} \dot\subset G'$
- [co-paw]: $\text{co-paw} \dot\subset G \;\vee\; G_1^7 \dot\subset G \;\vee\; G_2^7 \dot\subset G \iff \text{co-paw} \dot\subset G'$

$\Longleftarrow$

- Based on a large number of cases that makes it almost impossible to present a proper pen-and-paper version of them. Indeed, due to the number of cases, it is difficult to be sure that all the possible cases are considered and that each of them is tackled properly $\longrightarrow$ formalization
  - $[K_{1,3}]$: 4400 lines 😱
  - [co-paw]: 1600 lines

Certificates

- It is not uncommon in the literature to report specific values/bounds of parameters without proof

Certificates

- It is not uncommon in the literature to report specific values/bounds of parameters without proof (e.g. experiments, results of heuristics)

[Rebennack et al. 2011]

Table 1 Computational results on DIMACS graphs

Instances				Branch & Cut		
				LB_{root}	LP value in root	
Name	$\|V\|$	Dens.	α	BC	BC	BC_{rank}
brock200_2	200	0.50	12	12	20.99	22.01
brock200_4	200	0.34	17	14	29.93	31.54
brock400_2	400	0.25	29	22	63.84	67.66
brock400_4	400	0.25	33	23	63.89	67.98
c_fat200-1	200	0.92	12	12	12.71	12.86
c_fat200-2	200	0.84	24	22	24.00	24.00

Table 2
Values and bounds for $\chi_\rho^{d,n}(\mathbb{Z}\square\mathbb{Z})$.

$d \backslash n$	1	2	3
1	13–15	2	2
2	∞	12–**18**[(1)]	8
3	∞	∞	16-**22**[(2)]
4	∞	∞	44–**97**[(2)]
5	∞	∞	199–?
6	∞	∞	∞

[Korže et al. 2019]

For the sake of interest, the following table gives some bounds computed by using these approaches.

[Goddard et al. 2008]

n	6	7	8	9	10	11
$\chi_b(Q_n) \geq$	15	28	63	132	285	610
$\chi_b(Q_n) \leq$	25	49	95	219	441	881

Certificates

- It is not uncommon in the literature to report specific values/bounds of parameters without proof (e.g. experiments, results of heuristics)

[Rebennack et al. 2011]

Table 1 Computational results on DIMACS graphs

Instances				Branch & Cut		
				LB_{root}	LP value in root	
Name	$\|V\|$	Dens.	α	BC	BC	BC_{rank}
brock200_2	200	0.50	12	12	20.99	22.01
brock200_4	200	0.34	17	14	29.93	31.54
brock400_2	400	0.25	29	22	63.84	67.66
brock400_4	400	0.25	33	23	63.89	67.98
c_fat200-1	200	0.92	12	12	12.71	12.86
c_fat200-2	200	0.84	24	22	24.00	24.00

Table 2
Values and bounds for $\chi_\rho^{d,n}(\mathbb{Z}\square\mathbb{Z})$.

$d \backslash n$	1	2	3
1	13–15	2	2
2	∞	12–**18**[1]	8
3	∞	∞	16-**22**[2]
4	∞	∞	44-**97**[2]
5	∞	∞	199–?
6	∞	∞	∞

[Korže et al. 2019]

For the sake of interest, the following table gives some bounds computed by using these approaches.

[Goddard et al. 2008]

n	6	7	8	9	10	11
$\chi_b(Q_n) \geq$	15	28	63	132	285	610
$\chi_b(Q_n) \leq$	25	49	95	219	441	881

- We can take advantage of the formalized theory to generate Coq files that certifies a value/bound of graph parameters; here, $IR_w(G)$

Certificates

$IR_w(G) \geq b$ (obtained by an heuristic)

unweighted Name	$\|V(G)\|$	b	Time (s.)	unweighted Name	$\|V(G)\|$	b	Time (s.)
myciel3	11	5	0.7	4-Insertions_3	79	38	8.8
myciel4	23	11	0.8	3-FullIns_3	80	37	8.2
queen5_5	25	5	0.4	jean	80	38	8.0
1-FullIns_3	30	14	1.1	queen9_9	81	13	1.8
queen6_6	36	7	0.6	david	87	36	6.9
2-Insertions_3	37	18	1.6	mug88_1	88	33	6.5
myciel5	47	23	3.0	mug88_25	88	33	6.4
queen7_7	49	9	0.9	1-FullIns_4	93	45	14.5
2-FullIns_3	52	25	3.2	myciel6	95	47	19.3
3-Insertions_3	56	27	3.8	queen8_12	96	16	3.0
queen8_8	64	11	1.4	mug100_1	100	36	7.6
1-Insertions_4	67	32	6.0	mug100_25	100	36	8.2
huck	74	27	9.2	queen10_10	100	15	3.0
unweighted				weighted			
buenosaires	48	16	1.4	buenosaires	48	1634	16.7

Certificates

$IR_w(G) \geq b$ (obtained by an heuristic)

unweighted Name	$\lvert V(G) \rvert$	b	Time (s.)	unweighted Name	$\lvert V(G) \rvert$	b	Time (s.)
myciel3	11	5	0.7	4-Insertions_3	79	38	8.8
myciel4	23	11	0.8	3-FullIns_3	80	37	8.2
queen5_5	25	5	0.4	jean	80	38	8.0
1-FullIns_3	30	14	1.1	queen9_9	81	13	1.8
queen6_6	36	7	0.6	david	87	36	6.9
2-Insertions_3	37	18	1.6	mug88_1	88	33	6.5
myciel5	47	23	3.0	mug88_25	88	33	6.4
queen7_7	49	9	0.9	1-FullIns_4	93	45	14.5
2-FullIns_3	52	25	3.2	myciel6	95	47	19.3
3-Insertions_3	56	27	3.8	queen8_12	96	16	3.0
queen8_8	64	11	1.4	mug100_1	100	36	7.6
1-Insertions_4	67	32	6.0	mug100_25	100	36	8.2
huck	74	27	9.2	queen10_10	100	15	3.0
unweighted				weighted			
buenosaires	48	16	1.4	buenosaires	48	1634	16.7

Certificates

$IR_w(G) \geq b$ (obtained by an heuristic)

unweighted Name	$\lvert V(G)\rvert$	b	Time (s.)	unweighted Name	$\lvert V(G)\rvert$	b	Time (s.)
myciel3	11	5	0.7	4-Insertions_3	79	38	8.8
myciel4	23	11	0.8	3-FullIns_3	80	37	8.2
queen5_5	25	5	0.4	jean	80	38	8.0
1-FullIns_3	30	14	1.1	queen9_9	81	13	1.8
queen6_6	36	7	0.6	david	87	36	6.9
2-Insertions_3	37	18	1.6	mug88_1	88	33	6.5
myciel5	47	23	3.0	mug88_25	88	33	6.4
queen7_7	49	9	0.9	1-FullIns_4	93	45	14.5
2-FullIns_3	52	25	3.2	myciel6	95	47	19.3
3-Insertions_3	56	27	3.8	queen8_12	96	16	3.0
queen8_8	64	11	1.4	mug100_1	100	36	7.6
1-Insertions_4	67	32	6.0	mug100_25	100	36	8.2
huck	74	27	9.2	queen10_10	100	15	3.0
unweighted				weighted			
buenosaires	48	16	1.4	buenosaires	48	1634	16.7

Certificates

$IR_w(G) \geq b$ (obtained by an heuristic)

unweighted Name	$\lvert V(G)\rvert$	b	Time (s.)	unweighted Name	$\lvert V(G)\rvert$	b	Time (s.)
myciel3	11	5	0.7	4-Insertions_3	79	38	8.8
myciel4	23	11	0.8	3-FullIns_3	80	37	8.2
queen5_5	25	5	0.4	jean	80	38	8.0
1-FullIns_3	30	14	1.1	queen9_9	81	13	1.8
queen6_6	36	7	0.6	david	87	36	6.9
2-Insertions_3	37	18	1.6	mug88_1	88	33	6.5
myciel5	47	23	3.0	mug88_25	88	33	6.4
queen7_7	49	9	0.9	1-FullIns_4	93	45	14.5
2-FullIns_3	52	25	3.2	myciel6	95	47	19.3
3-Insertions_3	56	27	3.8	queen8_12	96	16	3.0
queen8_8	64	11	1.4	mug100_1	100	36	7.6
1-Insertions_4	67	32	6.0	mug100_25	100	36	8.2
huck	74	27	9.2	queen10_10	100	15	3.0
unweighted				weighted			
buenosaires	48	16	1.4	buenosaires	48	1634	16.7

Future work

- Improve performance for the weighted cases.
- Experiments with larger instances.

Future work

- Improve performance for the weighted cases.
- Experiments with larger instances.
- Checking primal bound (some set S is irredundand) is easy: O(|V|.|S|).
  However what about checking dual bound? (in order to obtain an exact value)
  Is it feasible for small-medium sized instances?

Future work

- Improve performance for the weighted cases.
- Experiments with larger instances.
- Checking primal bound (some set S is irredundand) is easy: $O(|V|.|S|)$.
  However what about checking dual bound? (in order to obtain an exact value)
  Is it feasible for small-medium sized instances?
  (and this question is open for any graph parameter)

Future work

- Improve performance for the weighted cases.
- Experiments with larger instances.
- Checking primal bound (some set S is irredundand) is easy: $O(|V|.|S|)$.
  However what about checking dual bound? (in order to obtain an exact value)
  Is it feasible for small-medium sized instances?
  (and this question is open for any graph parameter)

  Roadmap:
  - Propose an integer programming model for the parameter ✓
  - Solve an instance via B&B emphasizing reducing size of the tree ✓
  - Formalize concepts about linear programming [Allamigeon, Katz 2019]
  - ...about integer programming and B&B resolution
  - ...and perform the generator that uses the "trace" left by the solver

Thanks!

A bit of history

- [Cockayne, Hedetniemi, Miller 1978] Initial results about Γ (upper domination) and *IR*, in particular $\alpha(G) \leq \Gamma(G) \leq IR(G)$

A bit of history

- [Cockayne, Hedetniemi, Miller 1978] Initial results about Γ (upper domination) and IR, in particular $\alpha(G) \leq \Gamma(G) \leq IR(G)$

- Several works followed where the equality of these parameters holds for certain families of graphs, e.g. [Jacobson, Peters 1990]

A bit of history

- [Cockayne, Hedetniemi, Miller 1978] Initial results about Γ (upper domination) and IR, in particular $\alpha(G) \leq \Gamma(G) \leq IR(G)$
- Several works followed where the equality of these parameters holds for certain families of graphs, e.g. [Jacobson, Peters 1990]
- [McRae 1994] Several (NP-completeness) complexity results were given for Γ and IR

A bit of history

- [Cockayne, Hedetniemi, Miller 1978] Initial results about Γ (upper domination) and IR, in particular $\alpha(G) \leq \Gamma(G) \leq IR(G)$
- Several works followed where the equality of these parameters holds for certain families of graphs, e.g. [Jacobson, Peters 1990]
- [McRae 1994] Several (NP-completeness) complexity results were given for Γ and IR
- Up to 2002, as reported by [Favaron et al. 2002], more that 1500 research papers about dominating sets have been published; in particular, more than 100 explore properties of irredundant sets in graphs. Research in these topics are still active, e.g. [Monnot, Fernau, Manlove 2021]

A bit of history

- [Brešar et al. 2014] Results about Grundy domination parameter γ_{gr}; Γ, IR and γ_{gr} are 3 points of view that model the worst case of a greedy heuristic that finds a dominating set; also $IR(G) \leq \gamma_{gr}(G)$

A bit of history

- [Brešar et al. 2014] Results about Grundy domination parameter γ_{gr}; Γ, IR and γ_{gr} are 3 points of view that model the worst case of a greedy heuristic that finds a dominating set; also $IR(G) \leq \gamma_{gr}(G)$
- [Campêlo, S.- 2021] Heuristic and exact approaches for solving γ_{gr}

A bit of history

- [Brešar et al. 2014] Results about Grundy domination parameter γ_{gr}; Γ, IR and γ_{gr} are 3 points of view that model the worst case of a greedy heuristic that finds a dominating set; also $IR(G) \leq \gamma_{gr}(G)$
- [Campêlo, S.- 2021] Heuristic and exact approaches for solving γ_{gr}

Regarding weighted parameters:

- [Boyaci, Monnot 2017] Some complexity results about WUDS (Γ_w)

A bit of history

- [Brešar et al. 2014] Results about Grundy domination parameter γ_{gr}; Γ, IR and γ_{gr} are 3 points of view that model the worst case of a greedy heuristic that finds a dominating set; also $IR(G) \leq \gamma_{gr}(G)$
- [Campêlo, S.- 2021] Heuristic and exact approaches for solving γ_{gr}

Regarding weighted parameters:

- [Boyaci, Monnot 2017] Some complexity results about WUDS (Γ_w)
- [S.- 2019] Γ_w and IR_w (among other parameters) and the proof of $\Gamma_w(G) \leq IR_w(G)$ were formalized in Coq/Ssreflect

Bibliography

- E. Cockayne, S. Hedetniemi, D. Miller. Properties of hereditary hypergraphs and middle graphs. Canad. Math. Bull. 21:461–468, 1978
- M. Jacobson, K. Peters. Chordal graphs and upper irredundance, upper domination and independence. Discr. Math. 86:59–69, 1990
- A. McRae. Generalizing NP-Completeness proofs for bipartite graphs and chordal graphs. PhD. dissert, Clemson University, 1994
- O. Favaron, T. Haynes, S. Hedetniemi, M. Henning, D. Knisley. Total irredundance in graphs. Discr. Math. 256:115–127, 2002
- W. Goddard, S. M. Hedetniemi, S. T. Hedetniemi, J. Harris, D. Rall. Broadcast Chromatic Numbers of Graphs. Ars Comb. 86, 2008
- S. Rebennack, M. Oswald, D. Theis, H. Seitz, G. Reinelt, P. Pardalos. A Branch and Cut solver for the maximum stable set problem. J. Comb. Optim. 21:434–457, 2011
- B. Brešar, T. Gologranc, M. Milanič, D. Rall, R. Rizzi. Dominating sequences in graphs. Discr. Math. 336:22–36, 2014
- A. Boyaci, J. Monnot. Weighted upper domination number. ENDM 62:171–176, 2017
- D. Korže, Ž. Markuš, A. Vesel. A heuristic approach for searching (d,n)-packing colorings of infinite lattices. Discr. Appl. Math. 257:353–358, 2019
- X. Allamigeon, R. Katz. A Formalization of Convex Polyhedra Based on the Simplex Method. J. Autom. Reason. 63:323–345, 2019
- D. S.-. Formalization of the Domination Chain with Weighted Parameters. presented at ITP 2019
- C. Doczkal, D. Pous. Graph theory in coq: Minors, treewidth, and isomorphisms. J. Autom. Reason. 64:795–825, 2020
- J. Monnot, H. Fernau, D. Manlove. Algorithmic aspects of upper edge domination. Theor. Comput. Sci. 877:46–57, 2021
- M. Campêlo, D. S.-. An integer programming approach for solving a generalized version of the Grundy domination number. Discret. Appl. Math. 301:26–48, 2021