Stable Set Congestion Games on Chordal Graphs

Federico D'Onofrio[1] and Carla Michini[2]

[1]Department of Computer, Control, and Management Engineering A. Ruberti, Sapienza University of Rome
[2]Department of Industrial and Systems Engineering, University of Wisconsin-Madison

donofrio@diag.uniroma1.it

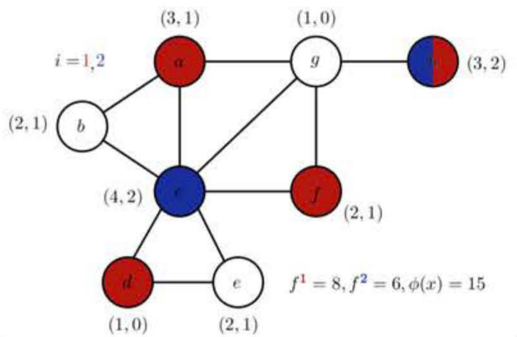
1. Introduction

- *Stable Set Congestion (SSC) games* are games of N players, each solving a Maximum Weight Stable Set problem on a graph $G = (V, E)$.

- A nonincreasing function $w_v : \{1, \dots, N\} \to \mathbb{Z}_+$ over the number of players selecting node $v \in V$ expresses the fact that a node might lose its value if many players use it.

Goal: compute a *Pure Nash Equilibrium (PNE)* of SSC games, i.e., a strategy profile s.t. no player has an incentive to unilaterally deviate from the stable set she selected.

2. SSC game example



3. Potential function

$X^i \subseteq \{0,1\}^V$: incidence vectors of stable sets.

$X = X^1 \times \cdots \times X^N$: strategy profiles.

Each PNE is local maximum of the *potential function*:

$$\phi(x) = \sum_{v \in V} \sum_{j=1}^{t_v(x)} w_v(j) \qquad x \in X$$

$t_v(x) :=$ nb. of players using node v in x.

4. The problem

Find a PNE by solving:

$$\begin{aligned} \max \ & \phi(x) \\ \text{s.t. } & x^i \in \mathrm{STAB}(G) \cap \{0,1\}^V \quad i = 1, \dots, N \end{aligned} \tag{1}$$

$\mathrm{STAB}(G) := conv\{\chi \in \{0,1\}^V : \chi$ is incidence vector of a stable set of $G\}$.

$$G \text{ perfect} \Rightarrow \mathrm{STAB}(G) = \{x \in \mathbb{R}^V : \sum_{v \in K} x_v \le 1 \quad K \text{ maximal clique}, \quad x_v \ge 0 \quad v \in V\}.$$

5. A two-phase approach

- Del Pia et al. (2017) propose an algorithm to compute in polynomial time a global maximum of (1) for SSC games on bipartite graphs.
- A generalization by Kleer and Shafer (2020) allows us to solve SSC games if STAB(G) is box-TDI and has the Integer Decomposition Property.

Phase 1 (*Aggregation*) Find the *aggregated strategy* z^*, i.e., $\forall$ node v find the nb. of players who select it in the global maximum of (1).

Phase 2 (*Decomposition*) Decompose z^* as $z = x^1 + \cdots + x^N$ s.t. $x^i \in \mathrm{STAB}(G) \cap \{0,1\}^V, i = 1, \dots, N$.

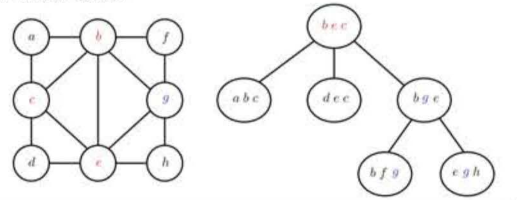
Our contribution. The previous approaches can not be directly applied to chordal graphs with treewidth k. We extend the two-phase approach to this case.

6. Chordal graphs and tree decompositions

Chordal graphs are perfect graphs s.t. every cycle of length greater than 3 has a chord.

A *tree decomposition* $(\{B_i | i \in I\}, T = (I, F))$ is a representation of G where T is a tree and each node i of T corresponds to a *bag* $B_i \subseteq V$.

Assume G is chordal. There exists a tree decomposition s.t. each bag is a maximal clique (*clique tree*) and it can be found in polynomial time. The size of the maximum clique minus one is called *treewidth* and it can be found in linear time.



7. The dynamic programming algorithm for the aggregation problem

Assume G is chordal with fixed treewidth k. Let $(\{B_i | i \in I\}, T = (I, F))$ be the clique tree of G rooted at node r. Our new dynamic programming algorithm exploits T and finds z^* in $\mathbf{O}(|\mathbf{V}|^2\mathbf{N}^{2k+1})$.

Consider the following *aggregation problem* $Agg(G)$:

$$\begin{aligned} \max \ & \phi^N(z) \\ \text{s.t. } & z \in N \cdot \mathrm{STAB}(G) \cap \mathbb{Z}^V \end{aligned} \tag{2}$$

where $N \cdot \mathrm{STAB}(G) := \{z \in [0, N]^V : \sum_{v \in K} z_v \le N \ \ \forall K \text{ maximal clique}\}$ and $\phi^N(z) := \sum_{v \in V} \sum_{j=1}^{z_v} w_v(j)$.

Main idea. $\forall$ node i of T and $\forall q \in \{0, 1, \dots, N\}^{B_i}$ compute value $f^q(i)$ corresponding to a partial solution of (2). Proceed from the leaves of T to r using information on previous nodes. Once in r construct the optimal solution z^* of (2).

More precisely, for $i \in I$:

- If i is a leaf, $\forall q \in \{0, 1, \dots, N\}^{B_i}$, if $\sum_{v \in B_i} q_v \le N$: $f^q(i) = \sum_{v \in B_i} \sum_{j=1}^{q_v} w_v(j)$; else $f^q(i) = -\infty$.
- If i has children $c_1, \dots, c_l$, $\forall q \in \{0, 1, \dots, N\}^{B_i}$, if $\sum_{v \in B_i} q_v \le N$:

$$f^q(i) = \sum_{v \in B_i} \sum_{j=1}^{q_v} w_v(j) + \sum_{h=1}^{l} \max\{f^y(c_h) - \sum_{v \in B_i \cap B_{c_h}} \sum_{j=1}^{q_v} w_v(j) : \\ y \in \{0, 1, \dots, N\}^{B_{c_h}}, \ y_v = q_v \text{ for } v \in B_i \cap B_{c_h}\}; \tag{3}$$

else $f^q(i) = -\infty$. For each child c_h store the solution $y^q_{c_h}$ used to compute $f^q(i)$.

Let G^i be the subgraph of G induced by all the nodes in the subtree of T rooted at i. We proved that $f^q(i)$ is the optimal value of $Agg(G^i)$ where the entries of z indexed by nodes in B_i have been fixed to q.

8. Decomposition as a coloring problem

We interpret the aggregated strategy z^* as a weight vector over V and we compute the decomposition of z^* in N stable sets by solving an exact weighted coloring problem. For chordal graph this problem can be solved in $\mathbf{O}(|\mathbf{V}|^2)$ (Hoàng, 1993).