

## EP2

Štefan Slavkovský

April 2024

### Rock and Ruck: solution

Solution is inspired by paper "Extremal Independent Set Reconfiguration".

First we need to realize that all Ricks positions can be represented by the pair of vertices. Invalid states are edges in graph and nothing more.

Therefore valid states are edges in complement of graph.

Running BFS on vertices on  $\overline{G}$ , (more formally: running BFS on  $Line(\overline{G})$ ), will find path from starting edge to final edge, or will decide that there is no path.

Problem we need to face is that creating complement of graph is  $\mathcal{O}(n^2)$  time operation.

We will present solution in  $\mathcal{O}(n + m + q)$  time.

First we will create graph  $H$  using following construction:

$$\begin{aligned} D &= \{v \mid v \in V(G) \wedge \deg(v) \geq \frac{n-1}{2}\} \\ S &= V(G) \setminus D \\ V(H) &= D \cup \{x\} \\ E(H) &= E(G[D]) \cup \{\{x, d\} \mid d \in D \wedge \exists s \in S : \{s, d\} \in E(G)\} \end{aligned}$$

Less formally:  $D$  is set of vertices with high degree. For  $H$  we take vertices and edges between  $D$  as they are.  $S$  is set of vertices with small degree. We represent whole set  $S$  with vertex  $x$  in  $H$ . Edge is put between  $x$  and vertex  $d$  from  $D$ , if there is edge between any vertex  $s$  from  $S$  and  $d$  in  $G$ .

This construction is taken from paper mention at the beginning of the solution.

First thing to note is that there is path between all vertices of  $S$  in  $\overline{G}$ , therefore we can represent whole set as single vertex  $x$ .

Second and most important thing is that we can bound size of  $H$ .

$$(V(H) - 1) \frac{n-1}{2} \leq \sum_{v \in D} \deg_G(v) \leq 2m$$

$$\mathcal{O}(V(H)) = \mathcal{O}\left(\frac{m}{n}\right)$$

All operation above can be easily implemented in linear time. Moreover creating complement of  $H$ , takes only  $\mathcal{O}(V(H)^2) = \mathcal{O}\left(\frac{m^2}{n^2}\right) \leq \mathcal{O}\left(\frac{m^2}{m}\right) = \mathcal{O}(m)$  time and space.

Finally we can compute to which component which vertex belongs. Than we can just need to output ‘YES’ if  $a_0$  component is same as  $a_k$  component in  $\overline{H}$ .