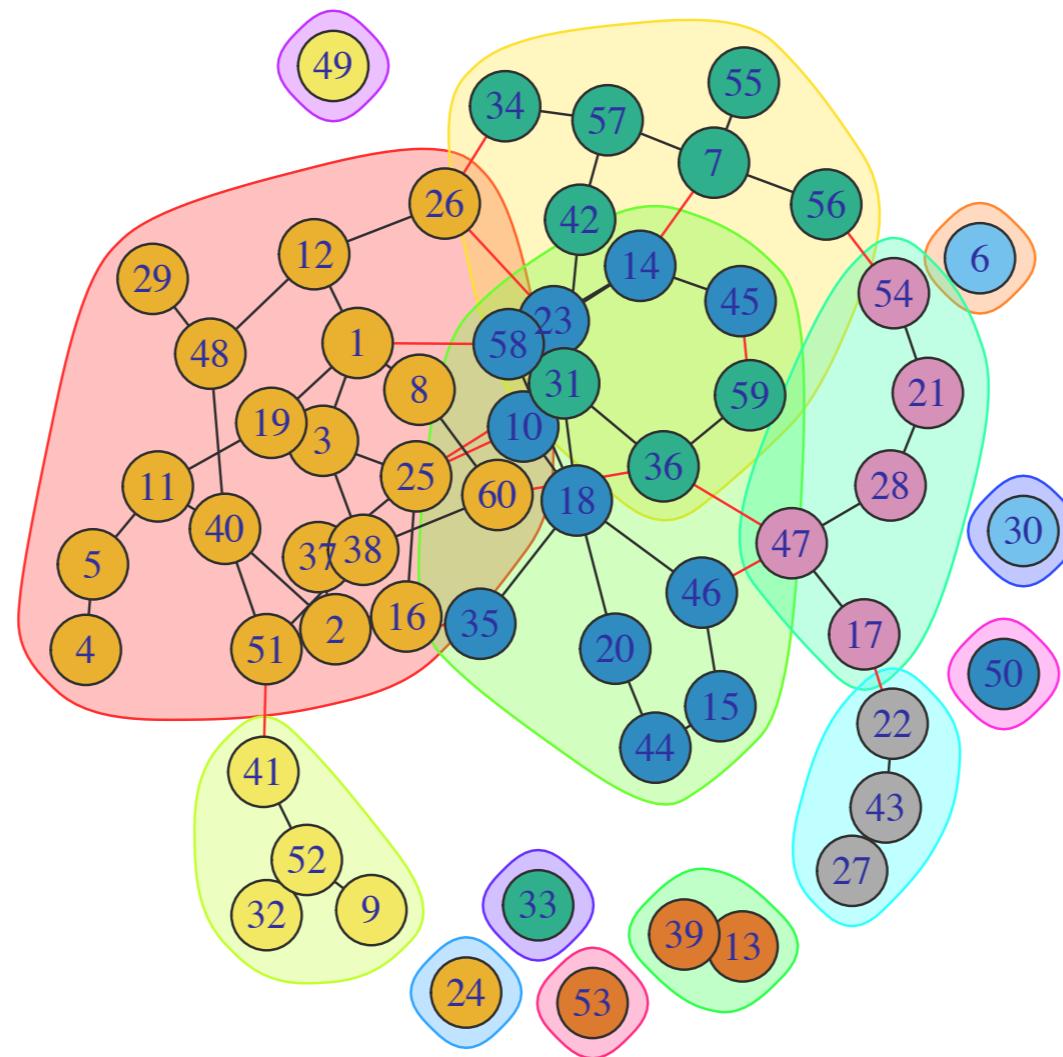
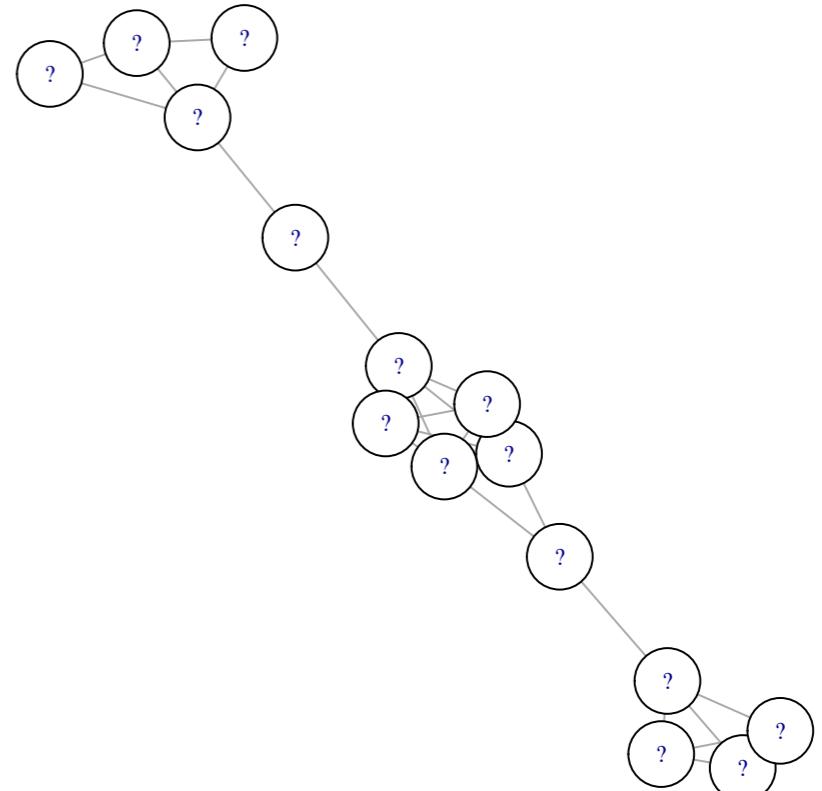
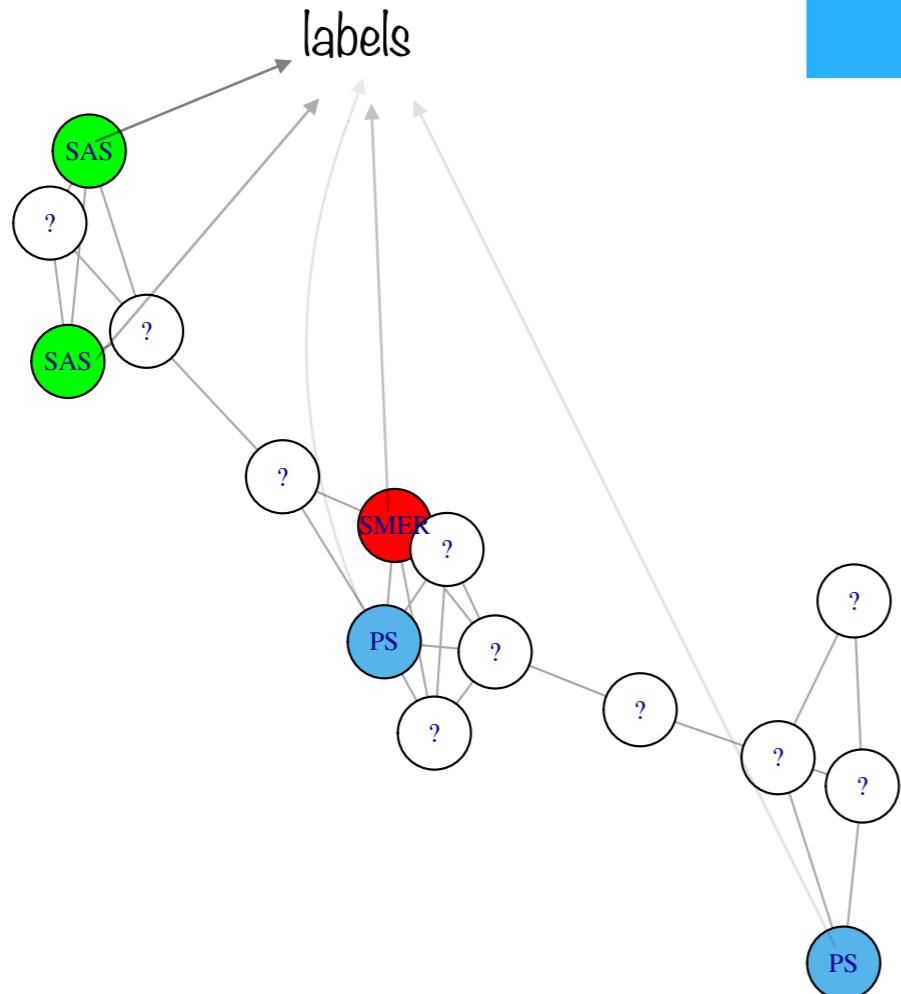


# Projekt - Label propagation algoritmus

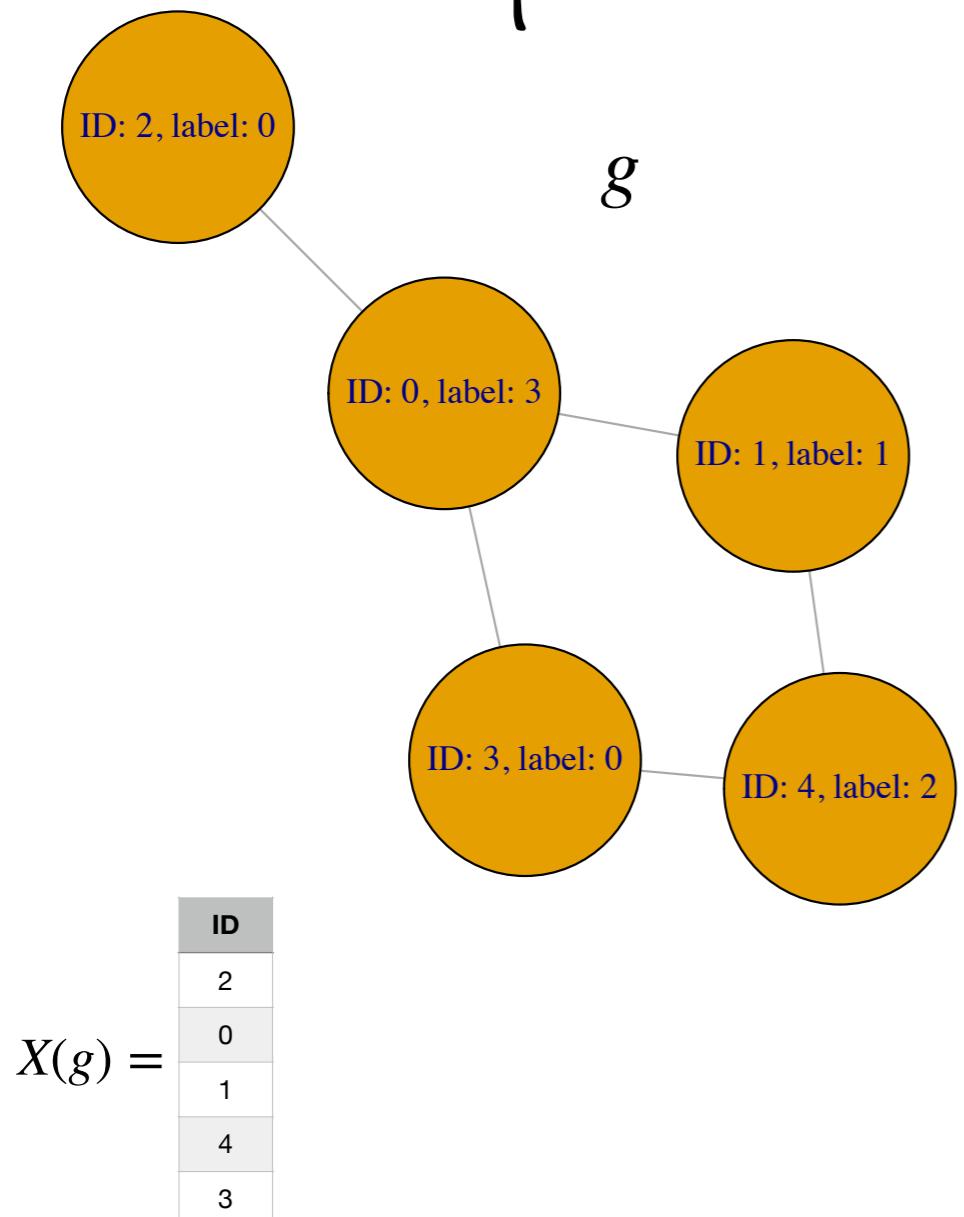


# Použitie

- zhľukovanie komunit
- rýchly stochastický algoritmus
- aplikovateľnosť pre
  - úplne neznámu sieť
  - sčasti neznámu sieť (modifikácia)



# Definície pre uľachčenie



$$L(0) = 0$$

$$L(1) = 2 \vee 3$$

$$L(2) = 0$$

$$L(3) = 2 \vee 3$$

$$L(4) = 1 \vee 0$$

$g$  - vytvorená siet' v R-ku, tj.  $\text{class}(g) = \text{"igraph"}$ .

$X(g)$  - funkcia, ktorá zo siete  $g$  vytvorí vektor ID-čiek všetkých vrcholov siete. ID každého vrcholu je vo vektore náhodne umiestnená a to práve jedenkrát. Výsledný vektorový výstup je označený ako  $X$ .

$L(x_i)$  - funkcia, ktorá pre vrchol s daným ID vráti label, ktorý sa medzi jeho susedmi vyskytuje s najvyššou frekvenciou. Ak existuje viacero labelov s najvyššou frekvenciou, tak sa vyberie jeden spomedzi nich rovnomerne náhodne.

# Popis algoritmu

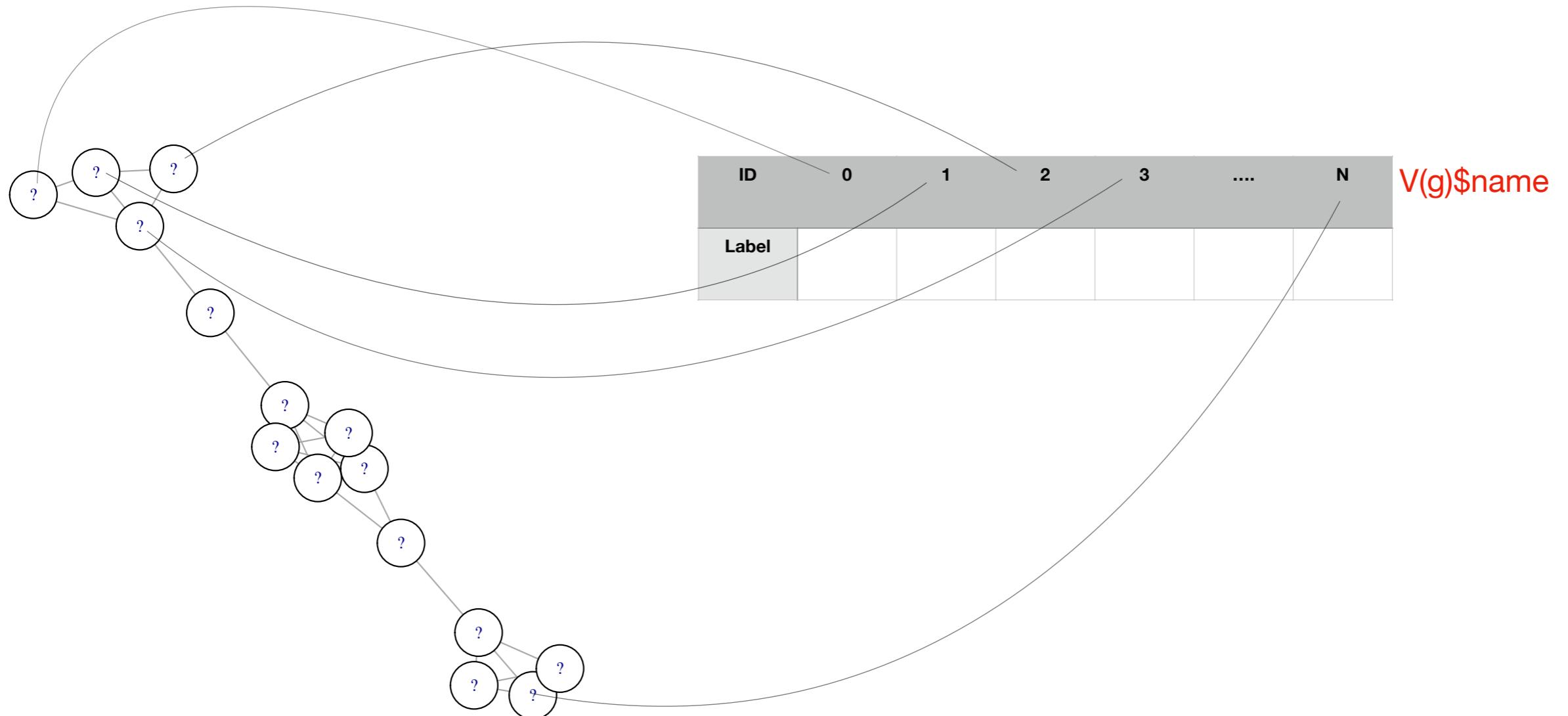
- I. Inicializácia unikatných ID-čiek pre každý vrchol siete.
- II. Inicializácia (číselných) rôznych labelov pre každý vrchol (node).
- III. Vytvoríme si  $X$ , tj. premiešaný vektor ID-čiek s využitím  $X(g)$ .
  - A. Iteračne, si pomocou každého ID  $x_i \in X$  vyhľadáme v sieti daný vrchol a k nemu prislúchajúcich susedov.
  - B. Danemu vrcholu zmeníme label na novú hodnotu získanú z  $L(x_i)$ .
- IV. Ak pre každý vrchol s ID  $x_i$  platí, že jeho label je zhodný s  $L(x_i)$  potom algoritmus končí a dostávame výsledné zhlukovanie, inak pokračujeme späť na III. Krok.

# Základná štruktúra

```
1 lpa <- function(g){  
2     g <- inicializaciaSiete(g)  
3     while(!jeSietOptimalna(g)){  
4         g <- robAlgoritmus(g)  
5     }  
6     return(g)  
7 }
```

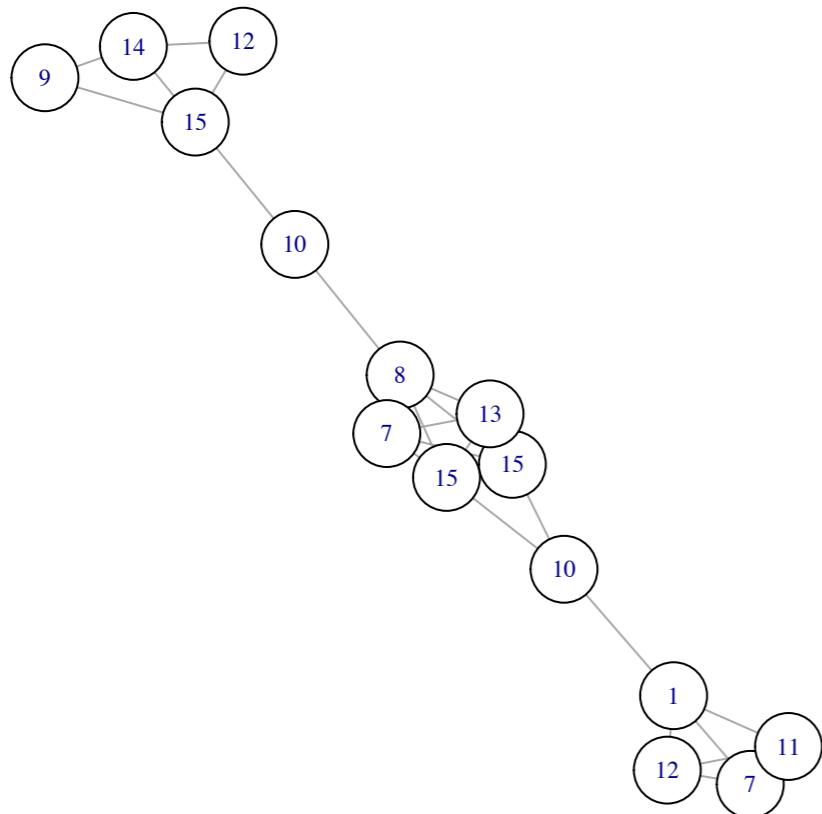
# Popis inicializácie 1.

Inicializácia unikatných ID-čiek pre každý vrchol siete.



# Popis inicializácie 2.

Inicializácia (napr. číselných) rôznych labelov pre každý vrchol (node)

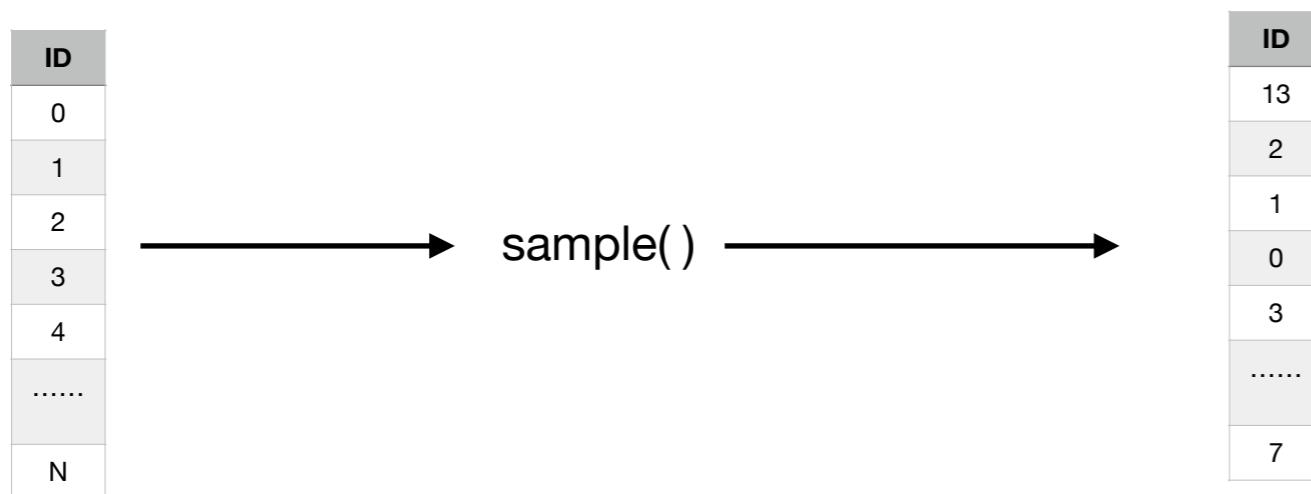


| ID    | 0 | 1  | 2  | 3  | .... | N | V(g)\$name  |
|-------|---|----|----|----|------|---|-------------|
| Label | 9 | 14 | 12 | 15 | ...  | 7 | V(g)\$label |

sample( )

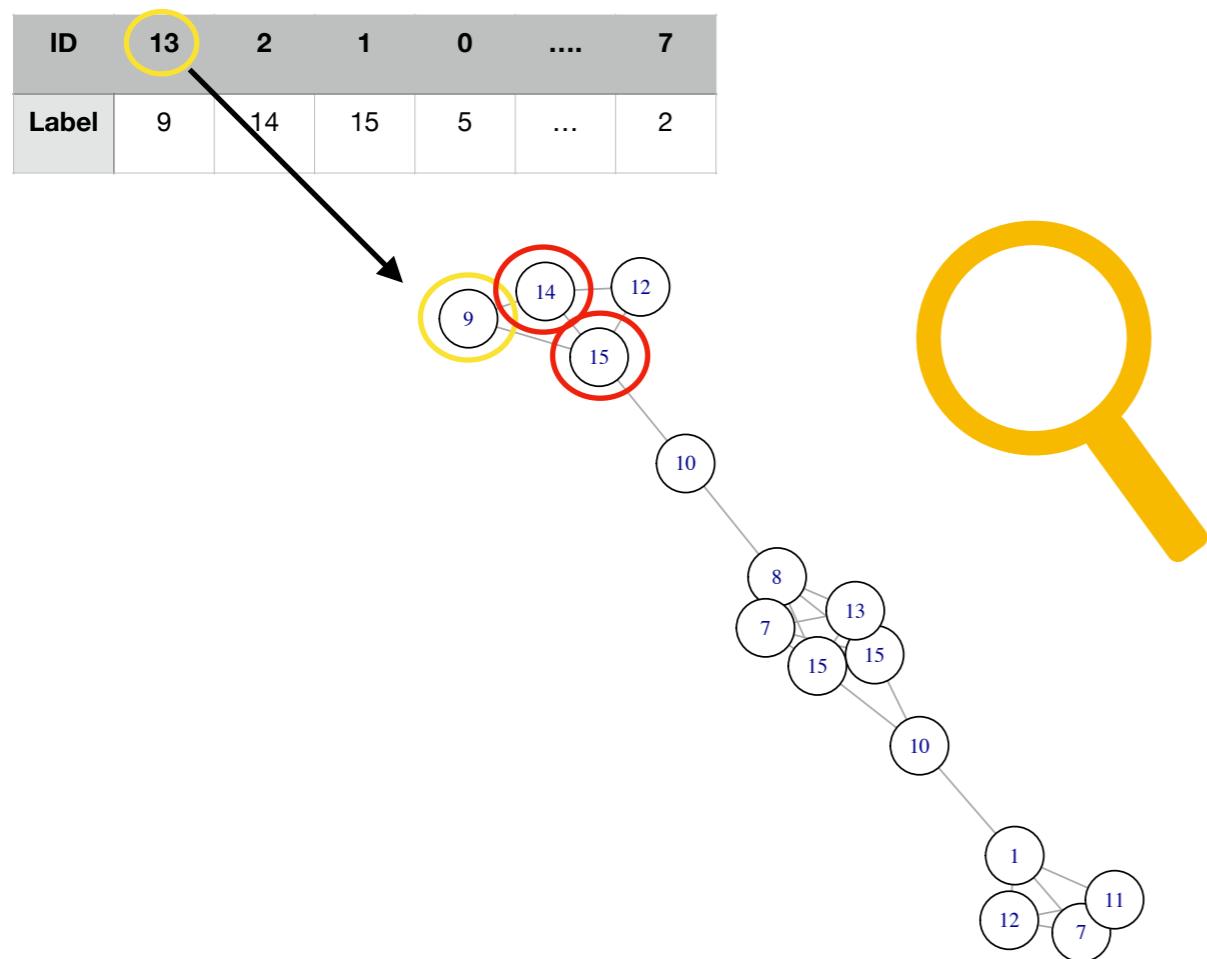
# Popis algoritmu 3.

A) Vytvoríme si  $X$ , tj. premiešaný vektor ID-čiek s využitím  $X(g)$ .



# Popis algoritmu 4.

B) Iteračne, si pomocou každého ID  $x_i \in X$  vyhľadáme v sieti daný vrchol a k nemu prislúchajúcich susedov.



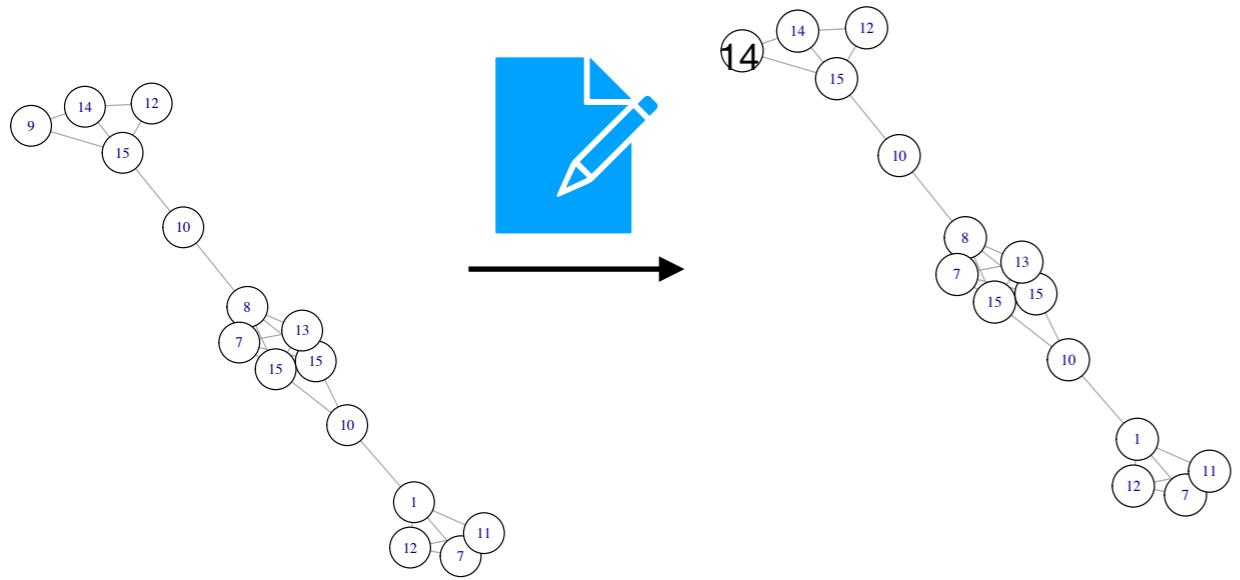
```

ID := '9'           IDX := 10
                    ↓
neighbors(g, ID / IDX)
                    ↓
list(
  name = c( 2, 1),
  label = c( 14, 15)
)
  
```

`ID := 9` → `which(ID == V(g)$name)` → `IDX := 10`

# Popis algoritmu 5.

c) Danému vrcholu zmeníme label na novú hodnotu získanú z  $L(x_i)$ .



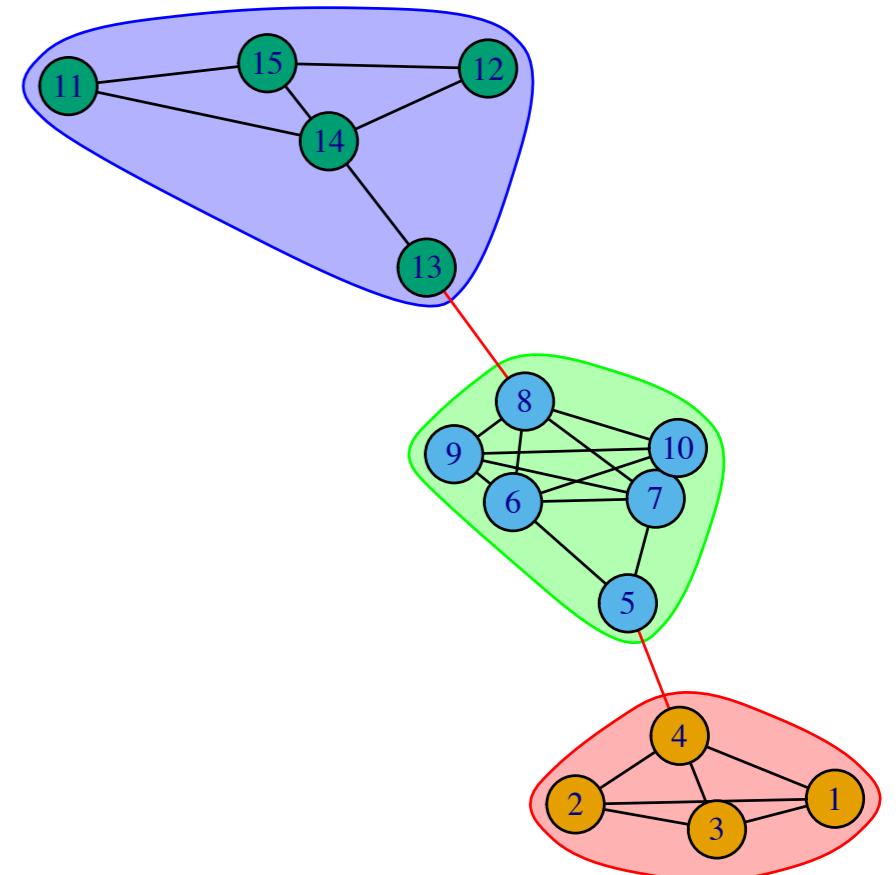
`V(g)$label[ IDX ] <- najviacFrekventovany( labels )`

|  |                   |   |                   |      |                       |
|--|-------------------|---|-------------------|------|-----------------------|
| <code>labels := [14,16,16,16,15,15]</code> | $\longrightarrow$ | <code>najviacFrekventovany( labels )</code> | $\longrightarrow$ | $16$ | $L(x_i) = 16$         |
| <code>labels := [14,15]</code>             | $\longrightarrow$ | <code>najviacFrekventovany( labels )</code> | $\longrightarrow$ | $15$ | $L(x_i) = 15 \mid 14$ |
| <code>labels := [14,15]</code>             | $\longrightarrow$ | <code>najviacFrekventovany( labels )</code> | $\longrightarrow$ | $14$ | $L(x_i) = 15 \mid 14$ |

# Popis testu optimalnosti 6.

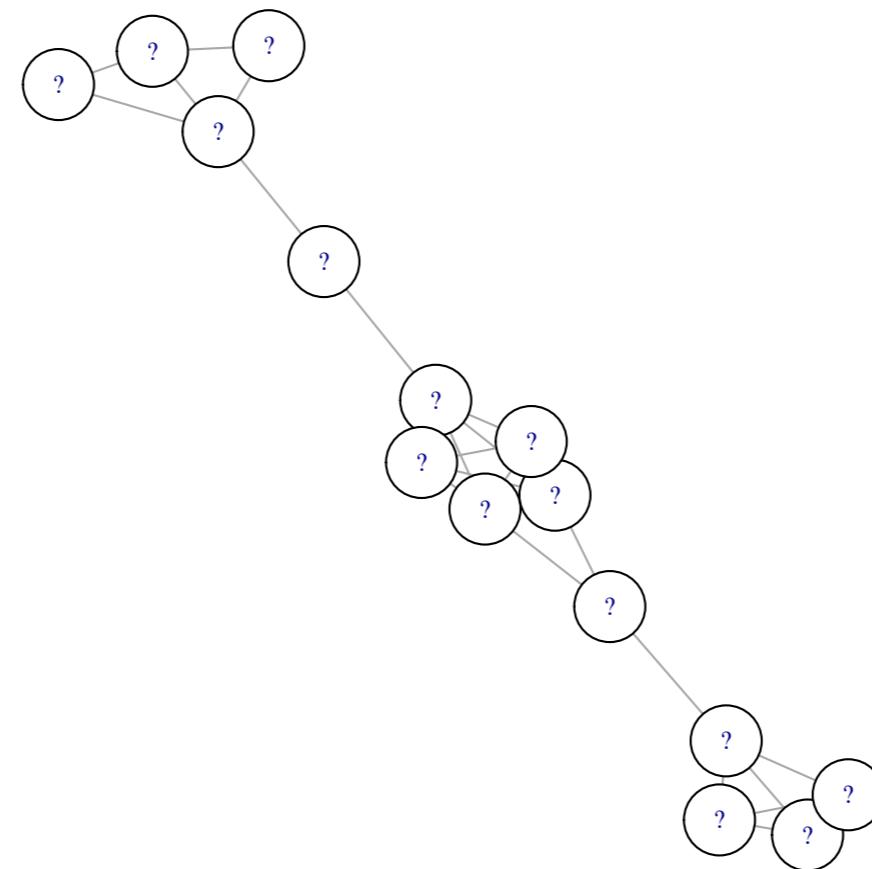
Ak pre každý vrchol s ID  $x_i$  platí, že jeho label je zhodný s  $L(x_i)$  potom algoritmus končí a dostávame výsledné zhľukovanie, inak opakujeme všetko od III. Kroku.

```
forEach id in V(g)$name  
.... neighbors(g,ID / IDX)$label ....  
If (.... najviacFrekventovany( labels ) ) ...
```

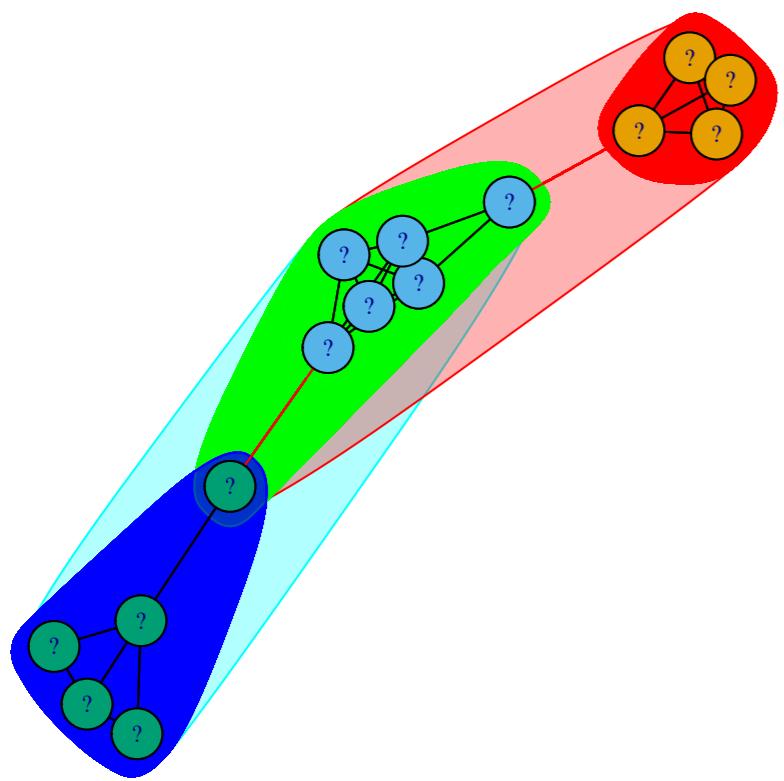


# Porovnanie algoritmov pre vygenerovanu siet'

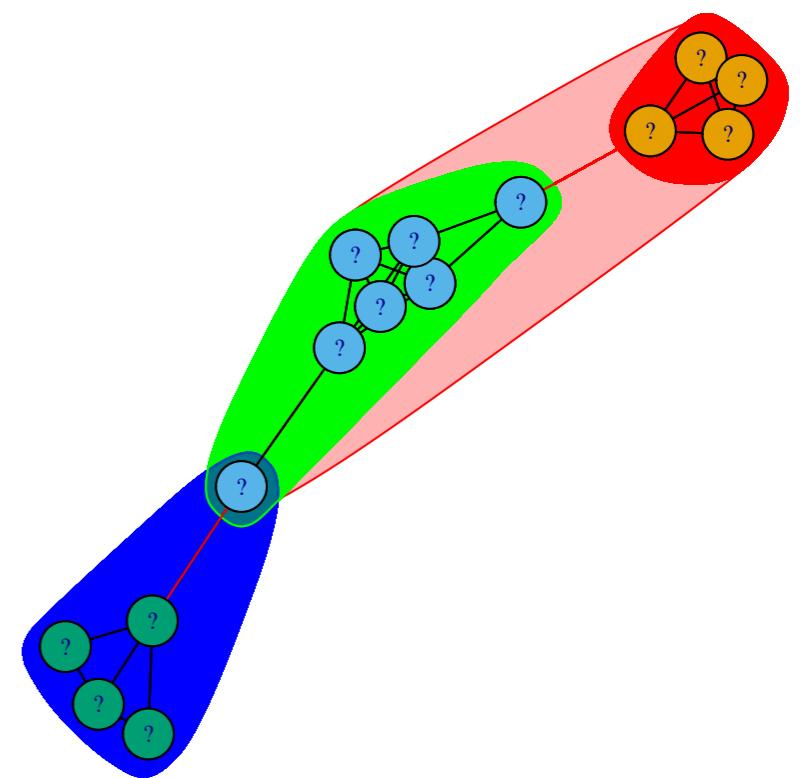
(1000 simulacii)



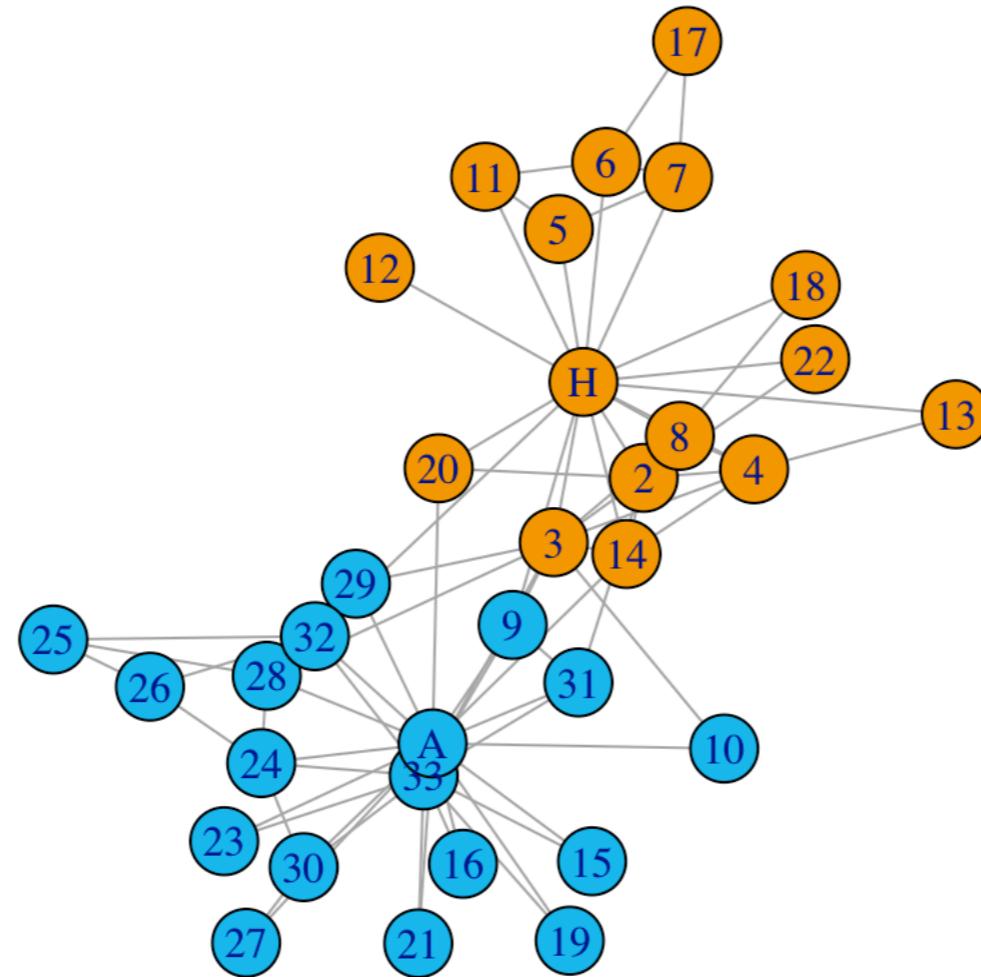
lpa



R lpa

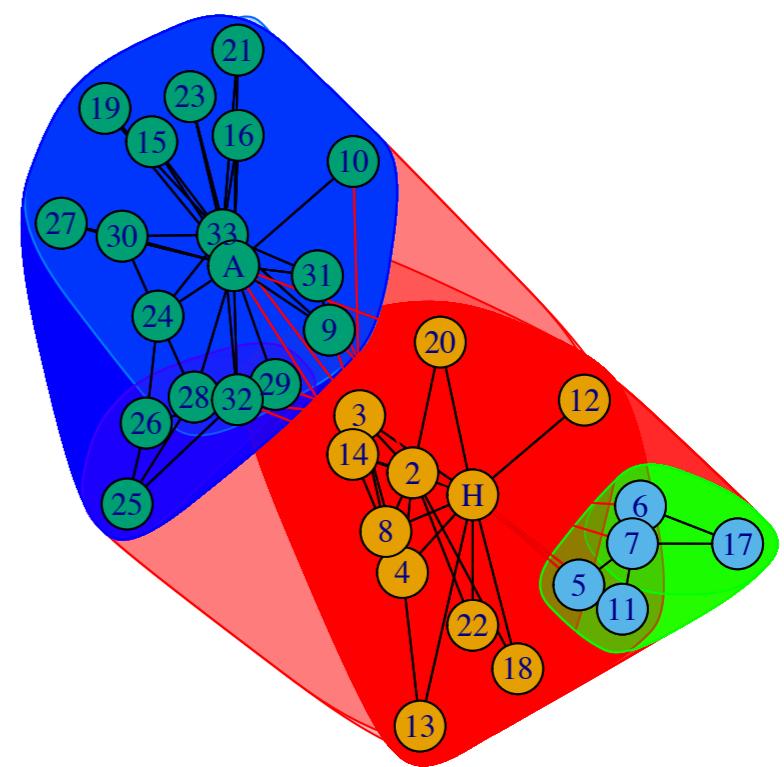
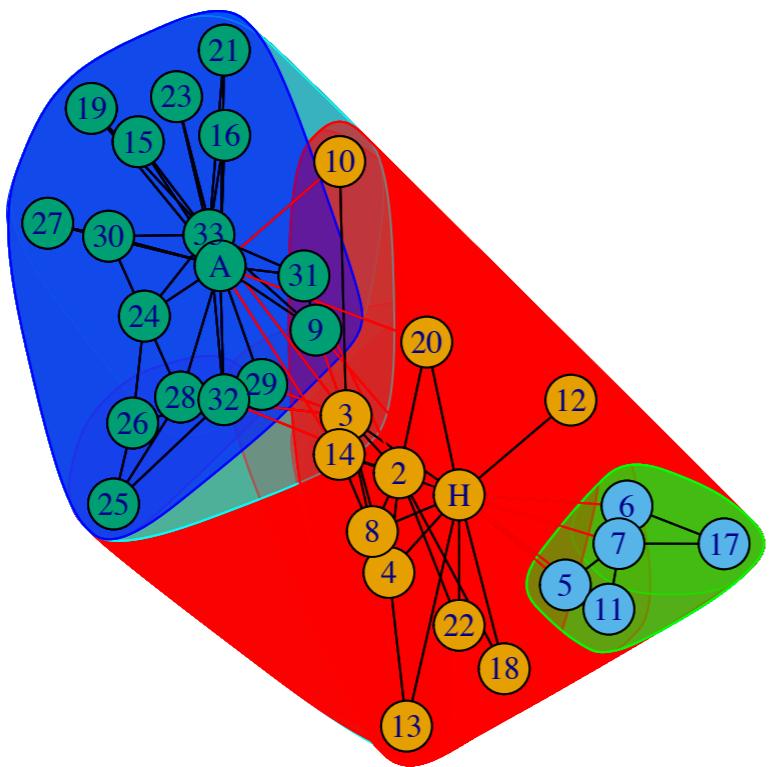


# Porovnanie algoritmov pre karate siet'



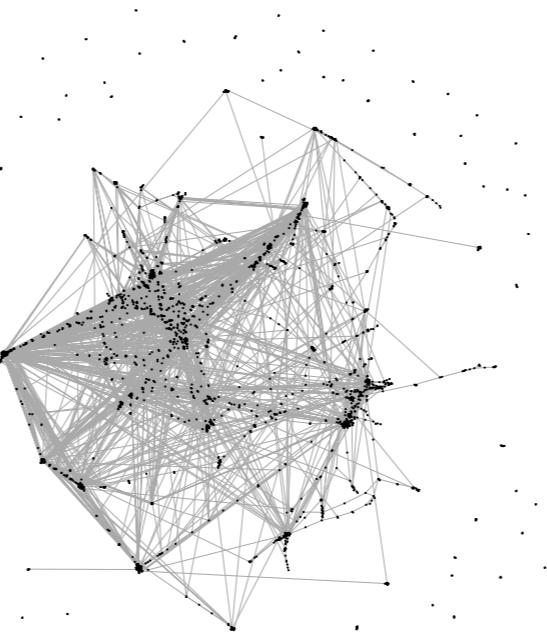
lpa

R lpa

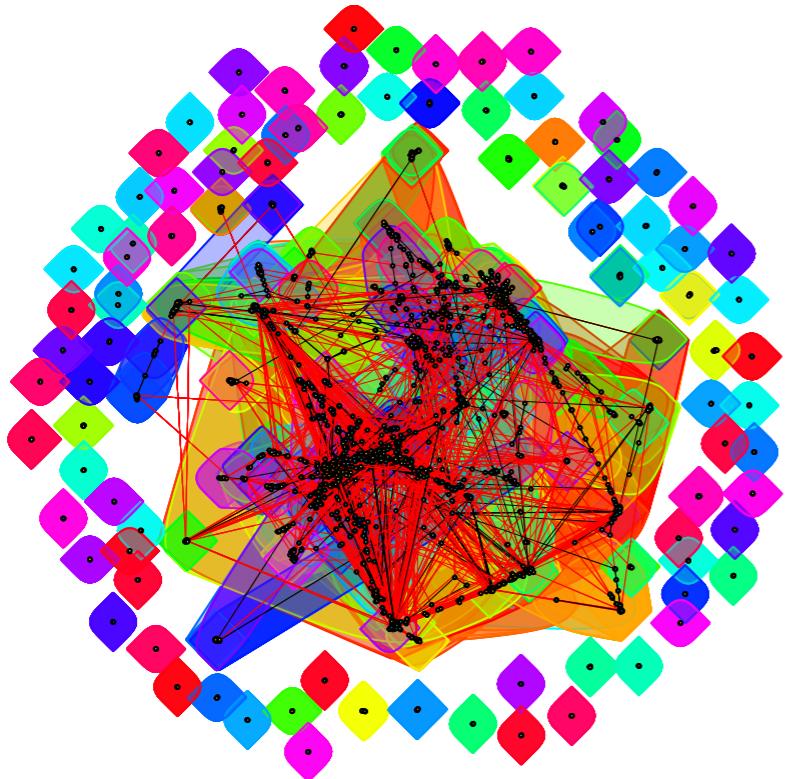


# Porovnanie algoritmov pre yeast siet'

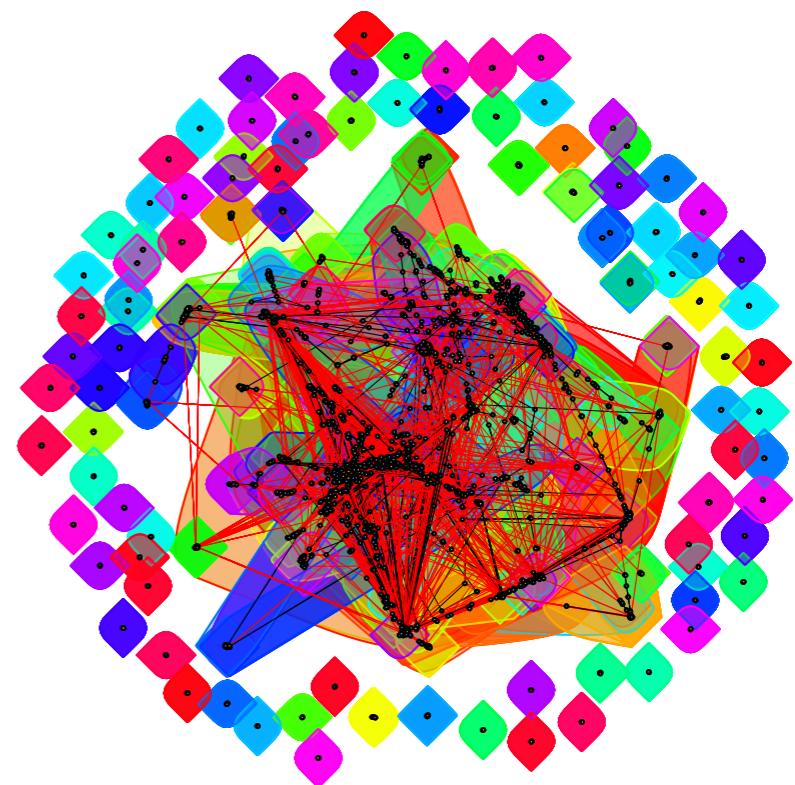
(10 simulácií)



lpa

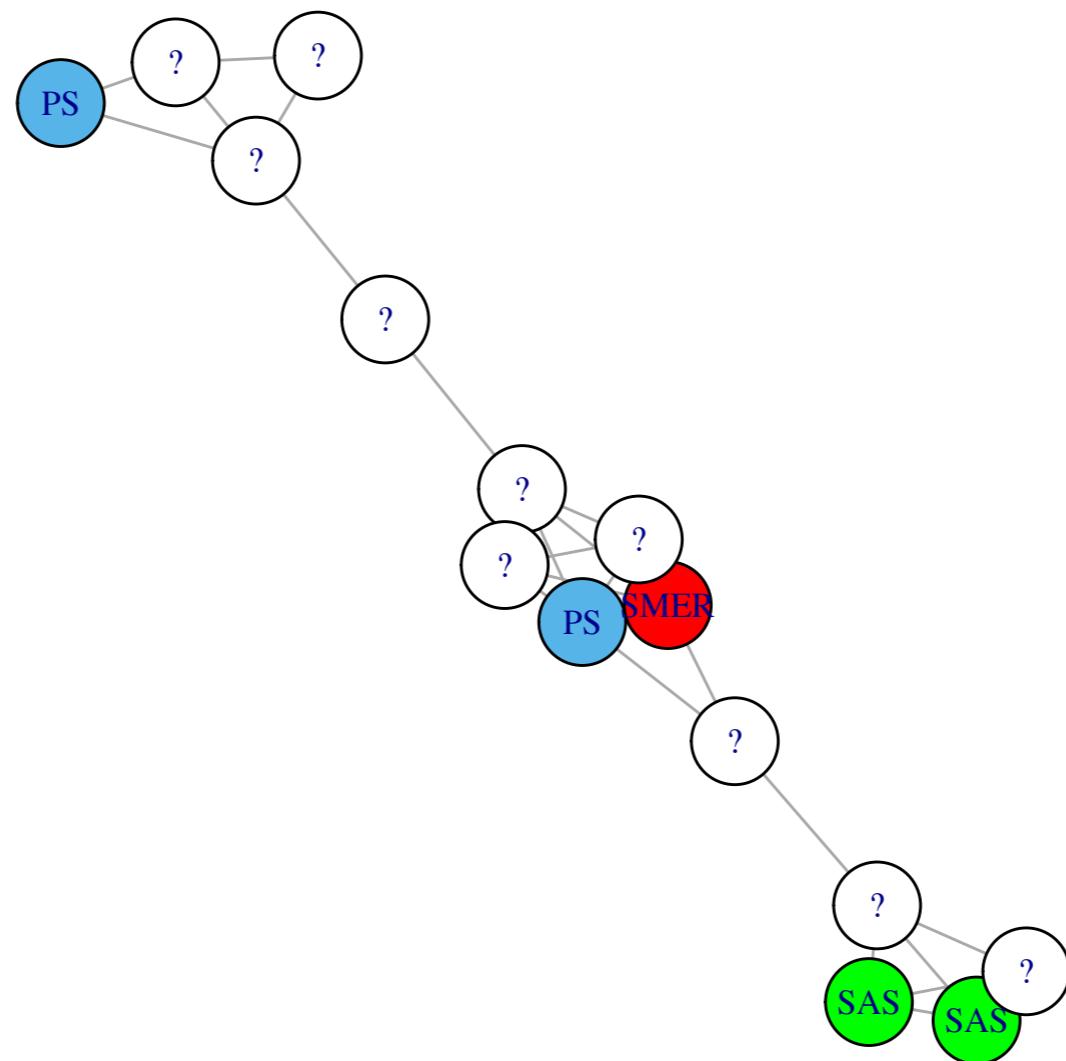


R lpa

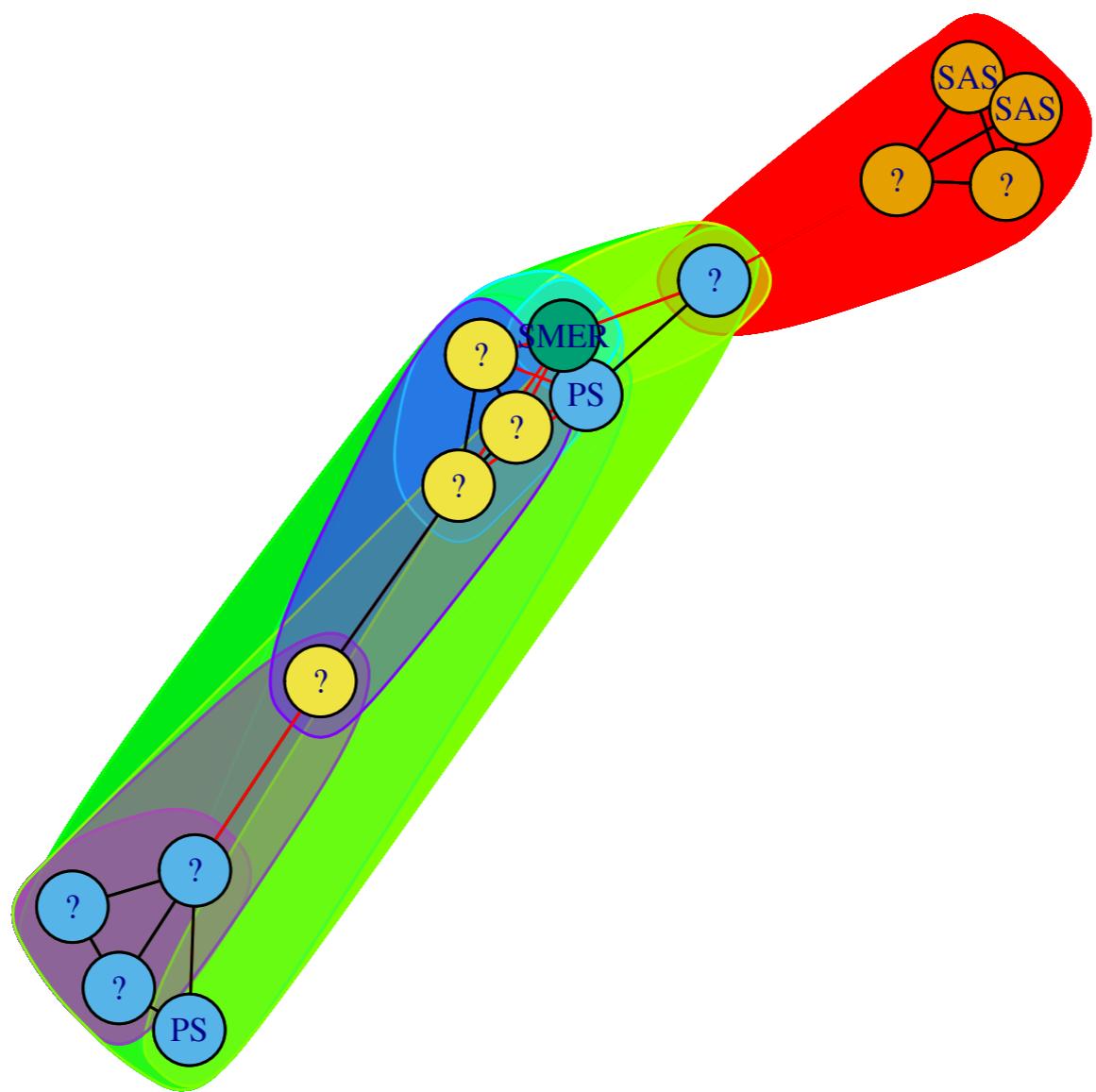


# Modifikovaný algoritmus pre sčasti neznámu siet'

(100 sim)

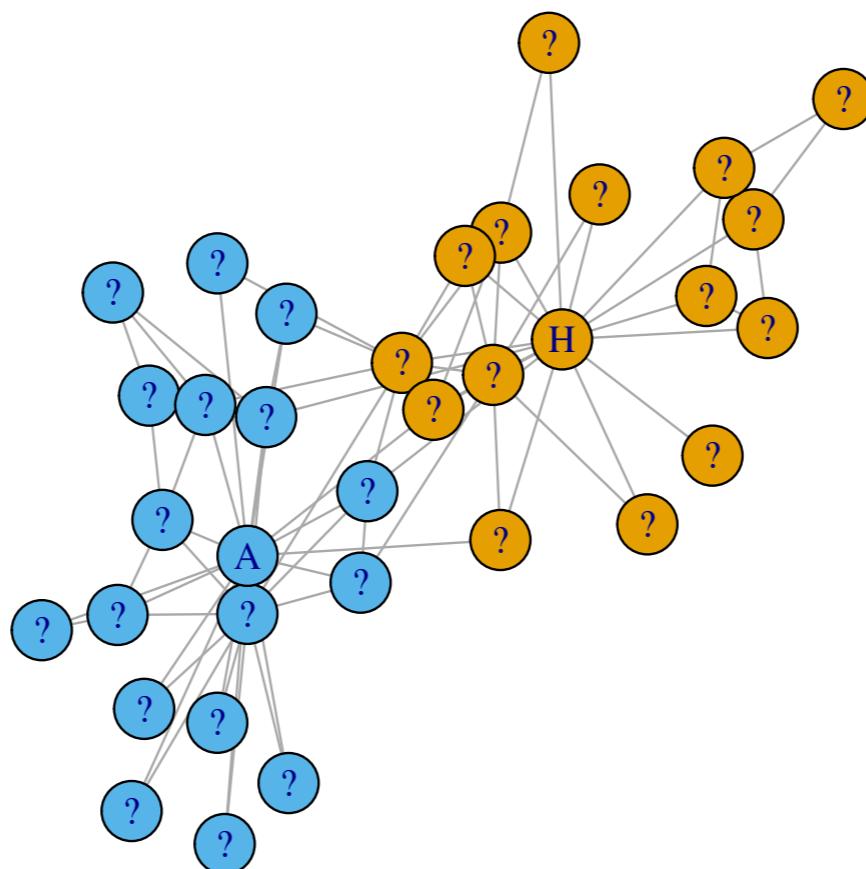


Lpal

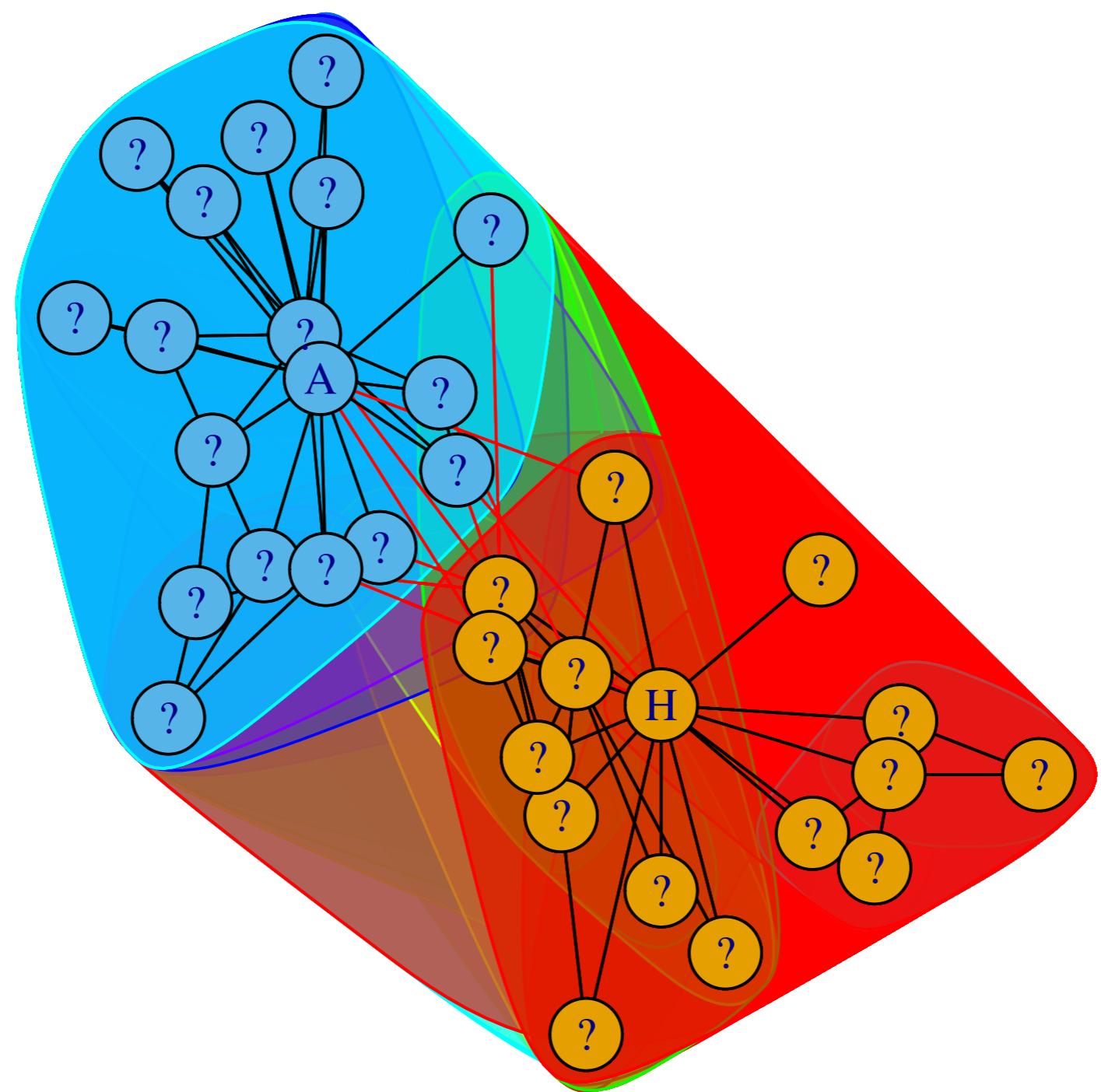


# Modifikovaný algoritmus pre sčasti neznámu siet'

(100 sim)

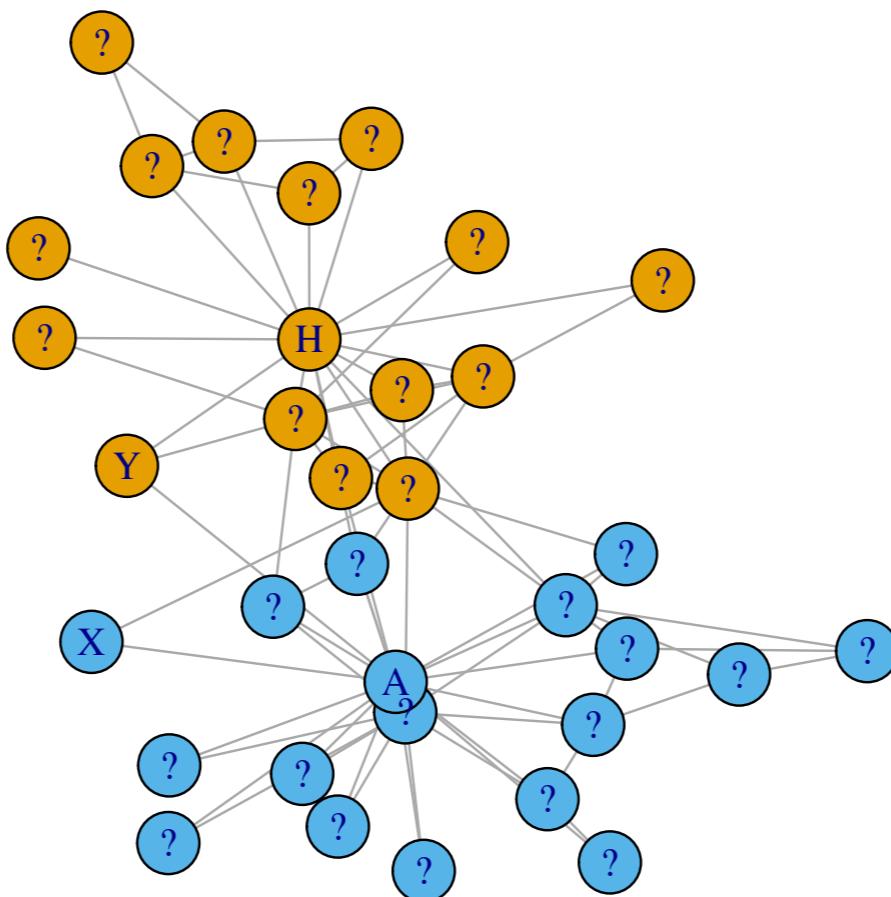


Lpal

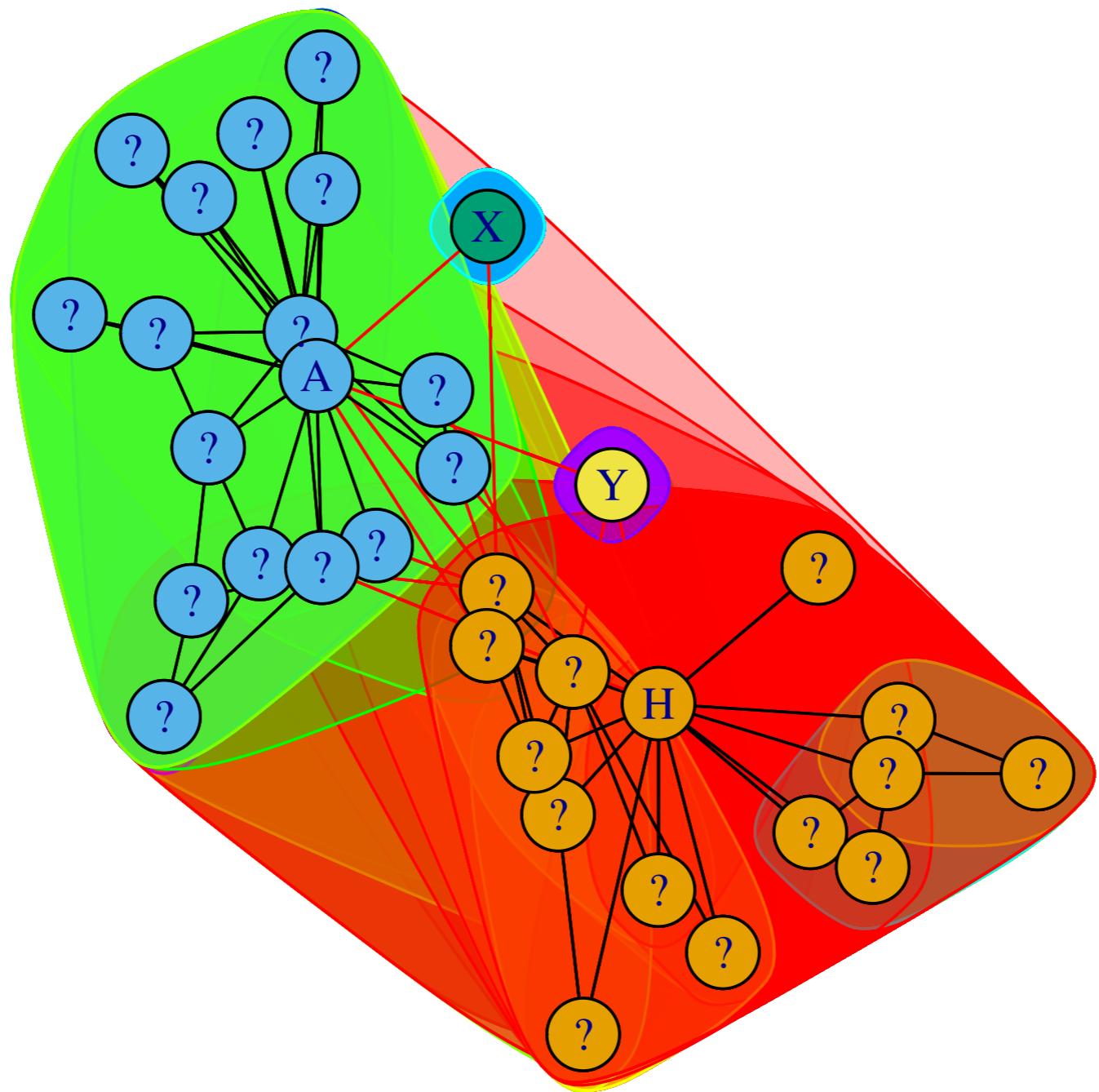


# Modifikovaný algoritmus pre sčasti neznámu siet'

(100 sim)

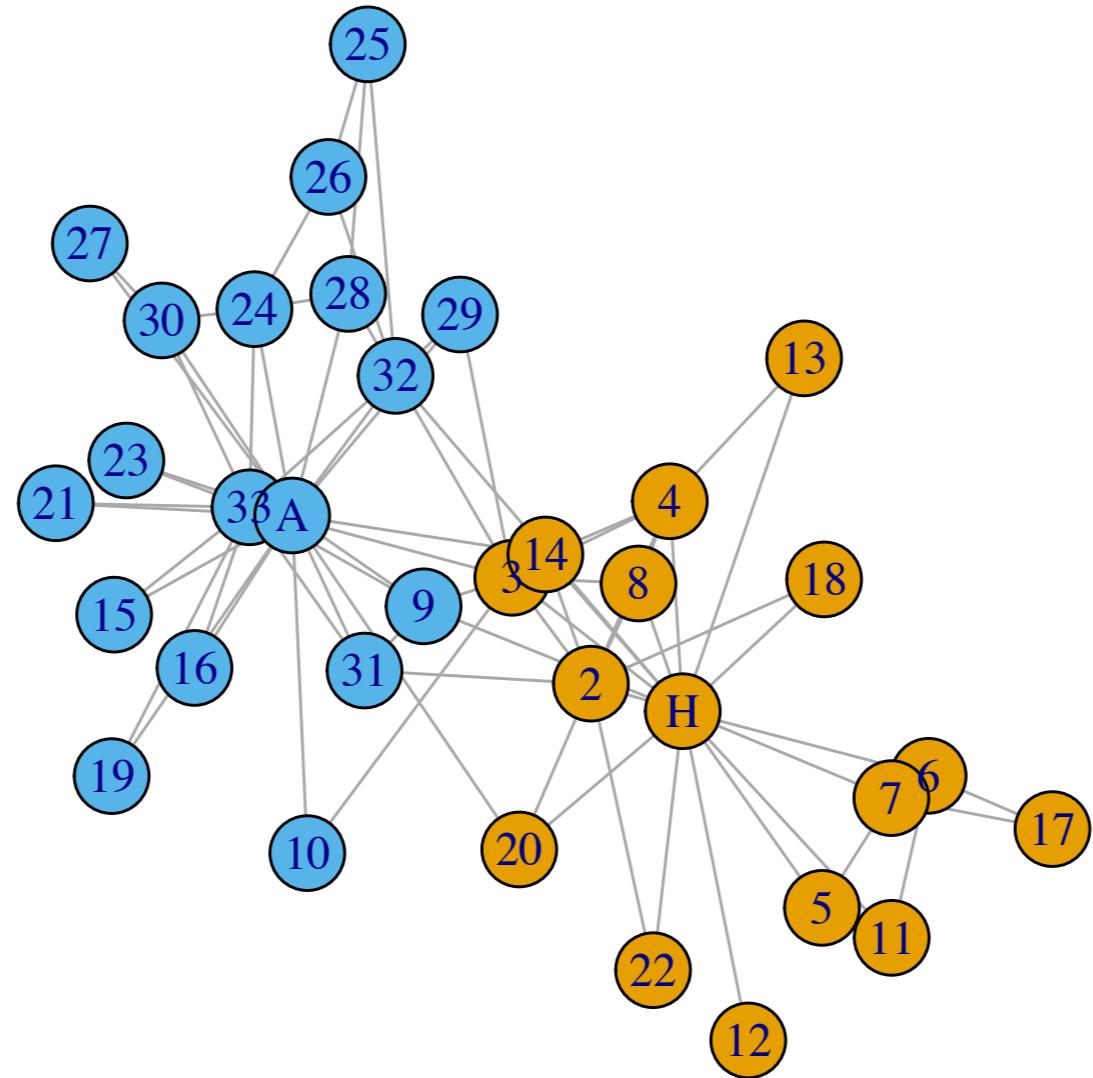


Lpal



# Modifikovaný algoritmus pre sčasti neznámu siet'

(n sim)



Lpal

