

SVEUČILIŠTE U ZAGREBU  
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 1523

**Simulacija utjecaja okupljanja  
zaraženih gripom na širenje  
epidemije**

Alen Rakipović

Zagreb, siječanj 2011.

*Zahvaljujem Bruni Rahleu na velikoj pomoći pri izradi programskog dijela ovog rada.*

*Zahvaljujem Josipu Hucaljuku na korisnim savjetima.*

*Zahvaljujem Mili Šikiću na svemu.*

# SADRŽAJ

<b>1. Uvod</b>	<b>1</b>
<b>2. Širenje zaraze na kompleksnim mrežama</b>	<b>2</b>
2.1. Opis problema . . . . .	2
2.1.1. Osnovni algoritam . . . . .	2
2.1.2. Zaraza i oporavak . . . . .	3
2.1.3. Okupljanje . . . . .	4
2.2. Implementacijski detalji i strukture podataka programskog rješenja . .	6
<b>3. Rezultati</b>	<b>9</b>
3.1. Mreža astrofizičara . . . . .	9
3.2. Mreža znanstvenika . . . . .	14
3.3. Energetska mreža . . . . .	18
3.4. Mreža korisnika s www.youtube.com web stranice . . . . .	22
3.5. Grafički prikaz simulacije širenja zaraze . . . . .	26
<b>4. Paralelizacija</b>	<b>28</b>
<b>5. Zaključak</b>	<b>29</b>
<b>Literatura</b>	<b>30</b>

# 1. Uvod

Svjedoci smo ekspanzije raznoraznih zaraza kao što su razne vrste gripe i ostalih virusnih bolesti. Ovi izrazito opasni virusi postaju velika opasnost za moderno društvo. Znanstvenici sa svih strana svijeta i iz mnogih područja sve više surađuju, te pristupaju ovom problemu na razne načine ne bi li uspjeli otkriti na koji se način zaraza širi, te kako ju spriječiti.

Kompleksne mreže [2] [3], odnosno njezina topološka i statistička svojstava, su idealan medij za proučavanje zaraze. U zadnjih desetak godina se došlo do mnogih saznanja o svojstvima kompleksnih mreža. U ovom radu će razmatrati svojstvo podložnosti širenju epidemija (npr. svinjska gripa u ljudskom društvu, odnosno računalni virusi na Internetu).

Postoji mnoštvo različitih modela mreže. Klasični je SIR model[6] mreže i sva proučavanja i testiranja će biti obavljena na proširenom modelu. SIR model podrazumijeva neku populaciju čije su jedinice različite po raznim svojstvima. Također, bitna pretpostavka je da su susreti bilo koje dvije jedinice jednako vjerojatni. U SIR modelu populacija se dijeli na tri odjeljka: *zaraženi*, *oporavljeni*, te *podložni zarazi*. Ovdje je taj model proširen s još jednim stanjem—*otporni*. Nakon što se jedinka zarazi, ona može ozdraviti i steći dodatnu otpornost organizma na nove valove zaraze. Naš model će se zasnivati na dva vala zaraze između kojih će se odviti mutacija zaraženih jedinki. Uz to, u prvom valu će se organizirati tzv. okupljanja te će se pokušati zaključiti koliki je njihov utjecaj na širenje zaraze. Ovaj model je preuzet iz članka [5].

U drugom poglavlju je dan detaljan opis problema, pseudokodovi algoritama, te implementacijski detalji programskog rješenja. Treće poglavlje donosi rezultate simulacija za tri kompleksne mreže. Svi rezultati su prikazani grafovima. U četvrtom poglavlju se razmatraju mogućnosti paralelizacije algoritma koristeći GPU i CUDA SDK [13].

## 2. Širenje zaraze na kompleksnim mrežama

### 2.1. Opis problema

Problem širenja zaraze se poprilično jednostavno može modelirati pomoću teorije grafova. Jedna kompleksna mreža predstavlja jedan graf. Vrhovi grafa su tada čvorovi, tj. osobe, u mreži, a njegovi bridovi — mogućnost interakcije osoba u mreži. Nakon modela mreže, prelazimo na algoritam širenja zaraze.

#### 2.1.1. Osnovni algoritam

Pseudokod osnovnog algoritma dan je u nastavku.

```
begin
  inicijalneRadnje();
  while (ima_zarazenih)
    okupljanje();
    zaraza();
    oporavak();
  end
  mutacija();
  while (ima_zarazenih)
    zaraza();
    oporavak();
  end
end
```

**Pseudokod 1:** Osnovni algoritam širenja zaraze

Algoritam započinje inicijalnim radnjama kojima se definiraju sljedeće veličine:

- vjerojatnost\_zaraze,
- vjerojatnost\_oporavka,
- vjerojatnost\_mutacije,
- vjerojatnost\_organiziranja\_okupljanja,
- broj\_domacina\_okupljanja,
- broj\_uzvanika\_po\_domacinu,
- broj\_simulacija,
- ime\_datoteke,
- pocetni\_cvor,
- pocetni\_cvor2.

U inicijalne radnje pripadaju i učitavanje mreže iz datoteke (bilo *.gml* ili *.txt*), te zaraza početnog čvora (*pocetni\_cvor*).

Nakon toga počinje prvi od dva vala zaraze. On počinje priređivanjem okupljanja o kojima će više riječi biti u nastavku. U prvom valu se odvija i zaraza susjeda zaraženih čvorova s vjerojatnošću zaraze (*vjerojatnost\_zaraze*). Korak zaraze i okupljanja slijedi korak oporavka. U tom dijelu se zaraženi čvorovi oporavljaju s određenom vjerojatnošću (*vjerojatnost\_oporavka*). Ovaj dio se izvršava u petlji koja iterira sve dok se i posljednji zaraženi čvor ne oporavi.

Završetkom prvog vala zaraze, počinje faza mutacija. Mutacije se odvijaju na oporavljenim čvorovima na način da se od svih oporavljenih čvorova odabere određen postotak (*vjerojatnost\_mutacije*) koji postaju imuni na daljnju zarazu. Odabrani čvorovi se ne mogu zaraziti u drugom valu zaraze.

Drugi val zaraze započinje zarazom novog početnog čvora (*pocetni\_cvor2*). Početni čvor mora imati svojstvo udaljenosti od početnog čvora iz prvog vala (*pocetni\_cvor*) za barem četiri razine. U ovom valu se ne priređuju okupljanja, već se zaraza odvija iterativno na način da se prvo pokušaju zaraziti susjedi početno zaraženog čvora, zatim slijedi faza njihovog oporavka. Potom su susjedi susjeda podvrgnuti fazama zaraze i oporavka. Ovaj postupak se izvršava sve dok se svi zaraženi čvorovi ne oporave.

### **2.1.2. Zaraza i oporavak**

U nastavku je priložen pseudokod algoritma zaraze kada se ne organiziraju okupljanja, te nema faze mutacije.

```

begin
  pocetni = zarazi(id);
  staro_zarazeni := pocetni;
  while (!staro_zarazeni.empty()) do
    for i := 1 to broj_cvorova_grafa step 1 do
      if (lista_cvorova[i].podlozan_zarazi == true)
        if (rand() < (1 - (1 - vjerojatnost_zaraze)broj_zarazenih))
          novo_zarazeni := lista_cvorova[i];
        end
      end
    end
    for i := 1 to length(staro_zarazeni) step 1 do
      if (rand() < vjerojatnost_oporavka)
        staro_zarazeni.izbaci(i);
      end
    end
    prebaci_zarazene(staro_zarazeni, novo_zarazeni);
  end
end
end

```

### **Pseudokod 2:** Algoritam zaraze i oporavka

Algoritam počinje odabirom čvora od kojeg počinje zaraza. Taj čvor se ubacuje u listu *staro\_zarazeni*. Faze zaraze i oporavka se odvijaju sve dok ima čvorova u toj listi. U svakoj iteraciji petlje se za svaki čvor u grafu koji je podložen zarazi generira pseudoslučajni broj u intervalu  $[0 - 1]$ , te ako je on manji od vjerojatnosti da se taj čvor zarazi, čvor se ubacuje u listu *novo\_zarazeni*. Nakon toga se svi čvorovi iz liste *staro\_zarazeni* pokušaju oporaviti na identičan način. Na kraju se svi čvorovi iz liste *novo\_zarazeni* prebacuju u listu *staro\_zarazeni*.

### **2.1.3. Okupljanje**

U terminima epidemiologije “okupljanje” označava organizirano okupljanje zdravih i zaraženih jedinki u cilju zaraze zdravih jedinki. Ovim postupkom se želi postići zaraza u pogodnom trenutku, npr. kod zaraze vodenim kozicama se nastoji zaraziti čovjeka dok je još u dječjoj dobi ne bi li lakše prebolio taj oblik virusa.

U nastavku je priložen pseudokod algoritma okupljanja.

```

begin
  glavni_organizatori := izdvoji_dio_zarazenih();
  N := length(glavni_organizatori);
  for i := 1 to N step 1 do
    organizatori[i] := dohvati_dio_susjeda(glavni_organizatori[i]);
  end
  if (ima_visestruko_odabranih(organizatori) == true)
    spoji_okupljanje();
  end
  M := length(organizatori);
  for i := 1 to M step 1 do
    okupljanje[i] := organizatori[i] + dohvati_dio_susjeda(organizatori[i]);
  end
  K := length(okupljanje);
  for i := 1 to K step 1 do
    zaraza(okupljanje[i]);
  end
end

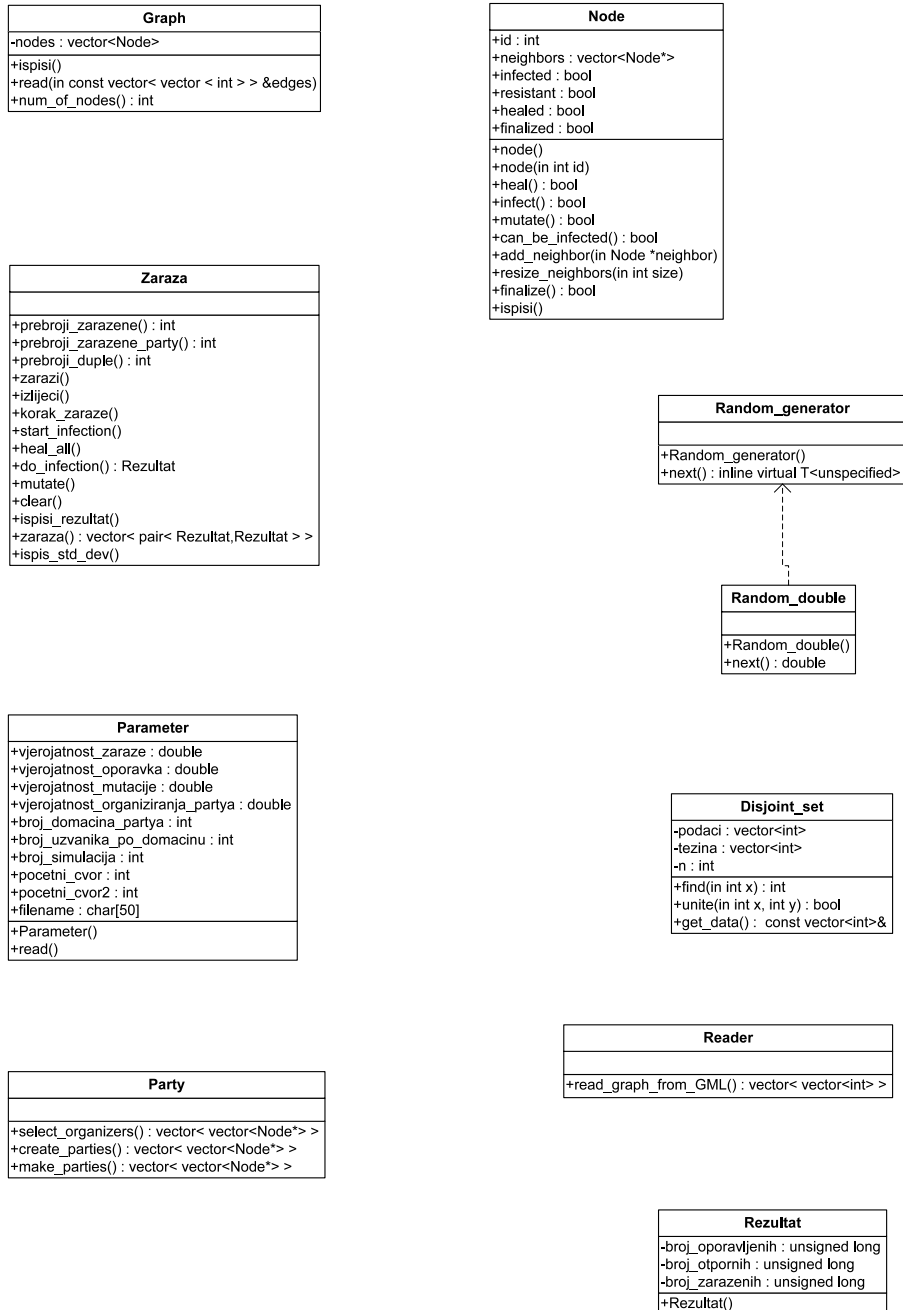
```

### Pseudokod 3: Algoritam okupljanja

Okupljanja se događaju u svakom koraku prvog vala zaraze nakon zaraze čvorova. Od svih trenutno zaraženih čvorova se odabire jedan dio (*vjerojatnost\_organiziranja\_okupljanja*) koji čine glavne organizatore okupljanja. Nakon toga se za svakog glavnog organizatora odabere nekoliko susjeda (*broj\_domacina\_okupljanja*). U slučaju da se među odabranim čvorovima u raznim skupinama pojavi više puta isti čvor, te skupine se spajaju u jedno zajedničko okupljanje. Nakon odabira skupina koje čine glavni organizatori i njihove obitelji (*broj\_domacina\_okupljanja + glavni\_organizator*), svaki od ovih čvorova poziva *broj\_domacina\_okupljanja* svojih susjeda na okupljanje. Ovime formiranje okupljanja završava, te se svi članovi skupine podvrgavaju djelovanju zaraze.



## 2.2. Implementacijski detalji i strukture podataka programskog rješenja



Slika 2.1: Dijagram razreda

Razred *Graph* je temeljni razred koji modelira strukture podataka u koju se pohranjuje mreža učitana iz datoteke. Sadrži listu svih čvorova, te metode za ispis i učitavanje bridova grafa. Što se tiče struktura podataka za pohranu grafa, korišteni su containeri iz C++ STL-a. Čvorovi su pohranjeni u container *vector* kao *vector<node> nodes*.

Razred *Node* opisuje jedan čvor grafa. Svaki čvor ima jedinstveni id, te oznaku stanja (*zaražen*, *oporavljen* ili *otporan*). Također, za svaki čvor moguće je pozivati metode koje će mu promijeniti stanje. Razred sadrži i metodu koja dodaje susjeda čvoru, te listu svih susjeda tog čvora. I ovdje je lista realizirana containerom *vector*, ali ovdje on sadrži pokazivače na čvorove (*vector<node\*> neighbors*). Ovime se smanjila memorijska i vremenska složenost programa.

Razred *Party* sadrži metode koje modeliraju okupljanje. Kada se želi organizirati okupljanje prvo se poziva metoda *make\_parties*. Ovdje se prvo izabere obitelj koja čini okupljanje te se članovi sprema u *vector<node \*> hosts*. Nakon toga se poziva metoda *select\_organizers* u kojoj se izabiru organizatori okupljanja. Organizatori se odabiru tako da se za svakog člana obitelji odabere još dio njegovih susjeda, te se oni spremaju u strukturu *vector <vector <node \*> > organizers*. Budući da su odabrani organizatori, kreće se na organiziranje okupljanja. To se odvija u metodi *create\_parties*. Za svakog od organizatora se odabere novih nekoliko susjeda te se sve zajedno spremi u *vector <vector <node \*> > parties* gdje vanjski *vector* zapravo spremište svih okupljanja, a unutrašnji sadržava čvorove svakog pojedinog okupljanja.

Razred *Zaraza* se sastoji samo od metoda koje implementiraju osnovni algoritam. Metoda *korak\_zaraze()* obavlja jednu zarazu, te oporavljanje čvorova grafa. Metoda *do\_infection()* obavlja jedan val zaraze, tj. ona poziva metodu *korak\_zaraze()* sve dok ima zaraženih čvorova. Metoda koja implementira algoritam dan Pseudokom 1 je *zaraza()*.

Razred *Random\_generator* je zapravo sučelje preko kojeg se pozivaju metode koje vraćaju pseudoslučajni broj.

Razred *Random\_double* implementira metode koje nasljeđuje od razreda *Random\_generator*. Metoda *next()* vraća pseudoslučajni broj u intervalu  $[0 - 1]$ .

Razred *Parameter* sadrži sve parametre potrebne za izvršavanje programa.

Razred *Reader* sadrži metodu za učitavanje mreže u *.gml[7]* formatu. Veze grafa se čuvaju u *vector< vector< int > >* strukturi podataka gdje vanjski *vector* označava da se pod tim rednim brojem nalazi neki čvor, a unutarnji čine id oznake njegovih susjeda.

Razred *Disjoint\_set* je pomoćni razred koji implementira uniju elemenata.

Razred *Rezultat* služi za pohranjivanje rezultata simulacija. Rezultati se pohranjuju u *vector< pair< Rezultat, Rezultat > > rezultati* pri čemu je svaki element *vector*-a čini par od broja zaraženih, mutiranih i oporavljenih u prvom valu zaraze, te istih varijabli u drugom valu zaraze.

## 3. Rezultati

Rezultati prikazuju ukupan broj zaraženih u prvom i u drugom valu zaraze. Na y-osi je prikazan broj zaraženih za odgovarajuću *vjerojatnost\_organiziranja\_okupljanja* koja je za svaki graf kretala u intervalu  $[0.01 - 0.2]$  s korakom od 0.01. Na grafovima je ta vjerojatnost označena  $k$ , te izražena u postocima. *Vjerojatnost\_zaraze*, te *vjerojatnost\_oporavka* su postavljene na 0.1. Pretpostavljena vrijednost veličina *broj\_domacina\_okupljanja* i *broj\_uzvanika\_po\_domacinu* je 2. U svrhu dobivanja realnog prikaza stanja nakon izvršenja algoritma (Pseudokod 1), taj se algoritam ponavlja određen broj puta (*broj\_simulacija*=1000). Ispod su prikazani rezultati simulacija na četiri različite mreže [12] [11]. Mreže su u *.gml* formatu, te su parsirane radi lakšeg manipuliranja podacima. Za svaku mrežu je prikazano dvadeset grafova — deset grafova prvog vala i deset grafova drugog vala za različite vjerojatnosti mutacije (interval  $[0.0 - 0.9]$  s korakom 0.1). Prvozaraženi čvor (*pocetni\_cvor*) prvog vala zaraze je uvijek s indeksom 0. Kod svakog grafa gdje se odvija okupljanje je prikazana jednadžba regresijskog pravca ili krivulje gdje varijabla  $k$  označava vjerojatnost organizacije okupljanja u postocima.

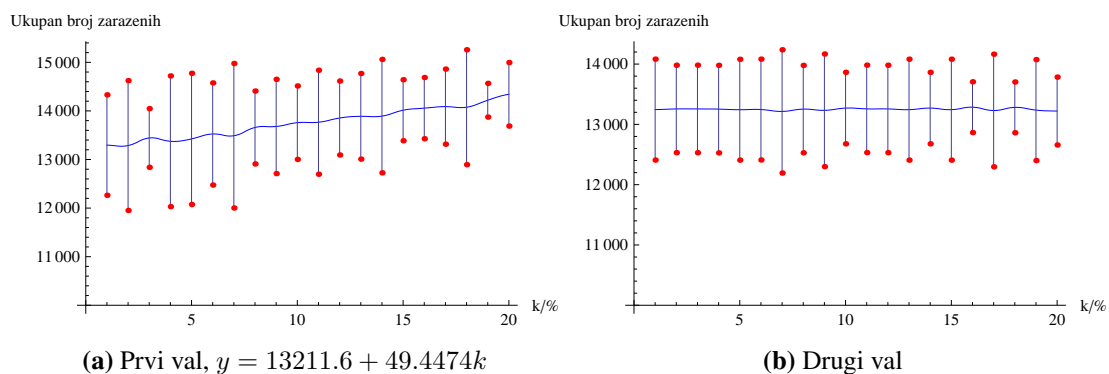
### 3.1. Mreža astrofizičara

Prva mreža je mreža koja predstavlja povezanosti astrofizičara u citiranju međusobnih znanstvenih radova.

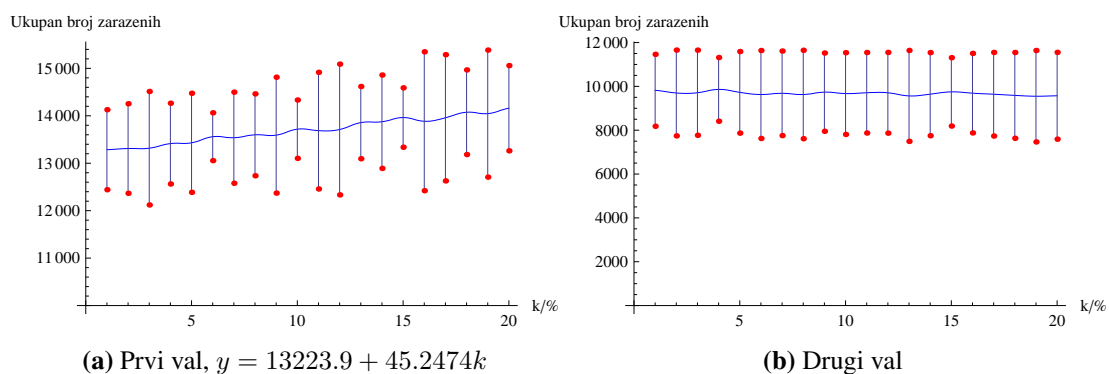
Mreža ima 16706 čvorova i 242502 veza. Prosječni stupanj čvora je 14.52. Početni čvor drugog vala (*pocetni\_cvor2*) ima id 782. Clustering koeficijent početnog čvora prvog vala je 0.0603175, a clustering koeficijent [1] početnog čvora drugog vala 0.887059. U tablici 3.1 su ispisane vrijednosti broja zaraženih u slučaju kada se ne organiziraju okupljanja.

**Tablica 3.1:** Tablica 1

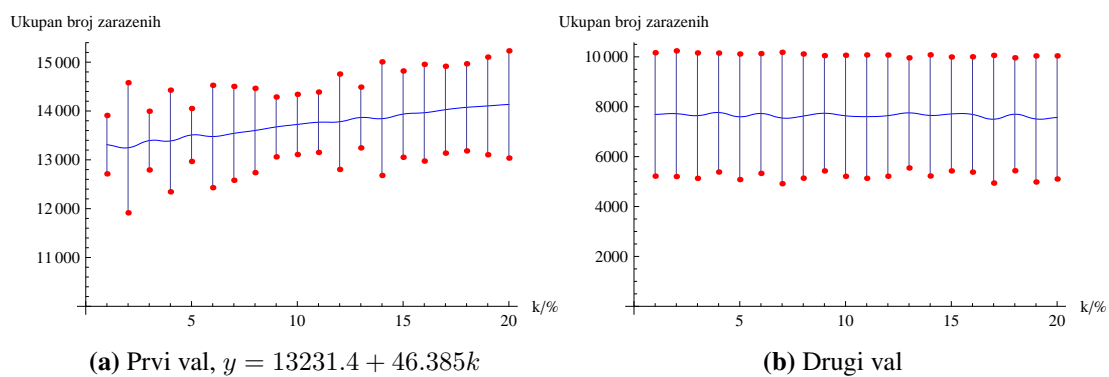
Vjerojatnost mutacije	Prvi val	Drugi val
$m = 0$	13232	13265
$m = 0.1$	13231	11580
$m = 0.2$	13245	9733
$m = 0.3$	13228	7732
$m = 0.4$	13229	5626
$m = 0.5$	13225	3527
$m = 0.6$	13239	1775
$m = 0.7$	13235	660
$m = 0.8$	13231	186
$m = 0.9$	13229	76



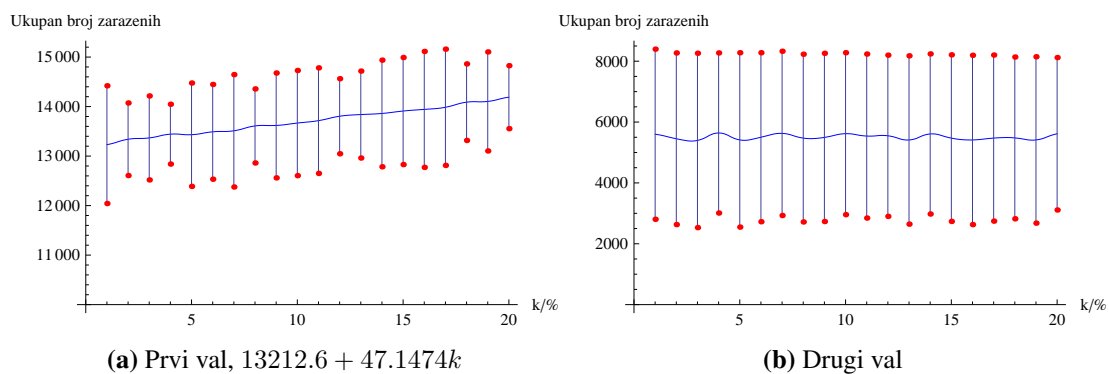
**Slika 3.1:** Mreža astrofizičara,  $m = 0.0$



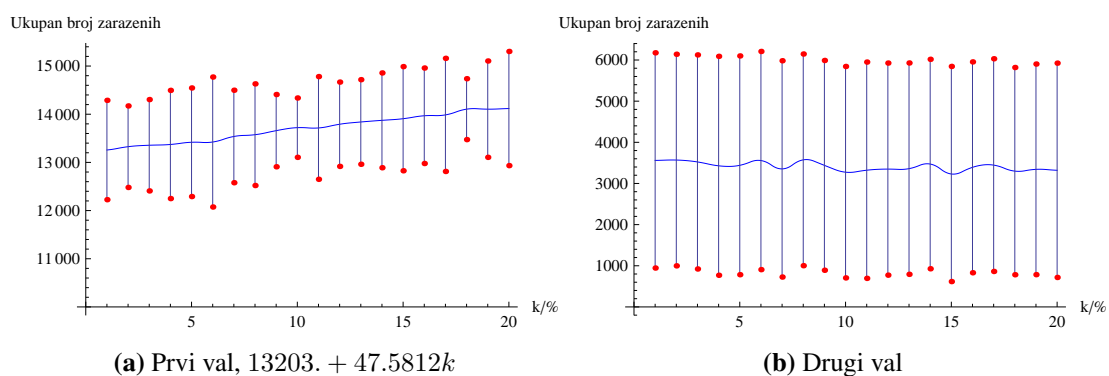
**Slika 3.2:** Mreža astrofizičara,  $m = 0.1$



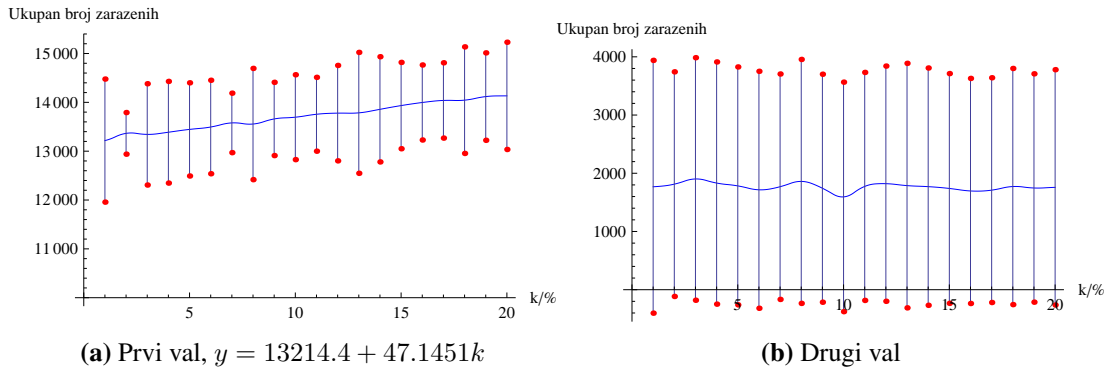
**Slika 3.3:** Mreža astrofizičara,  $m = 0.2$



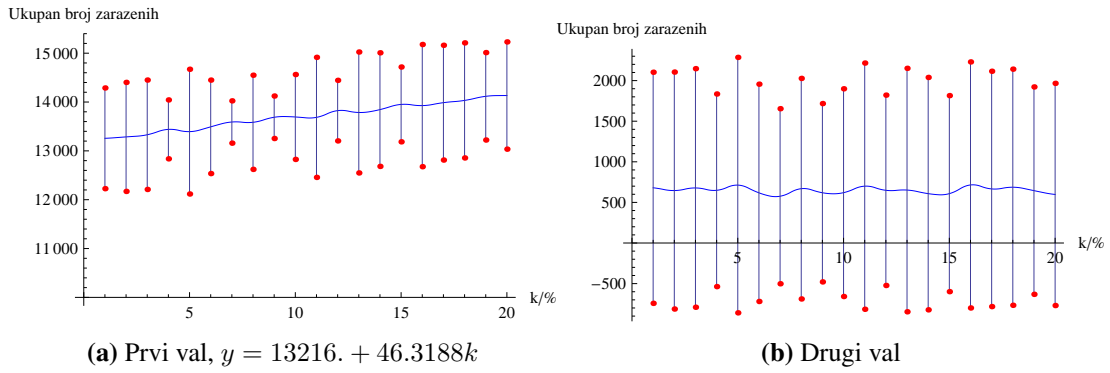
**Slika 3.4:** Mreža astrofizičara,  $m = 0.3$



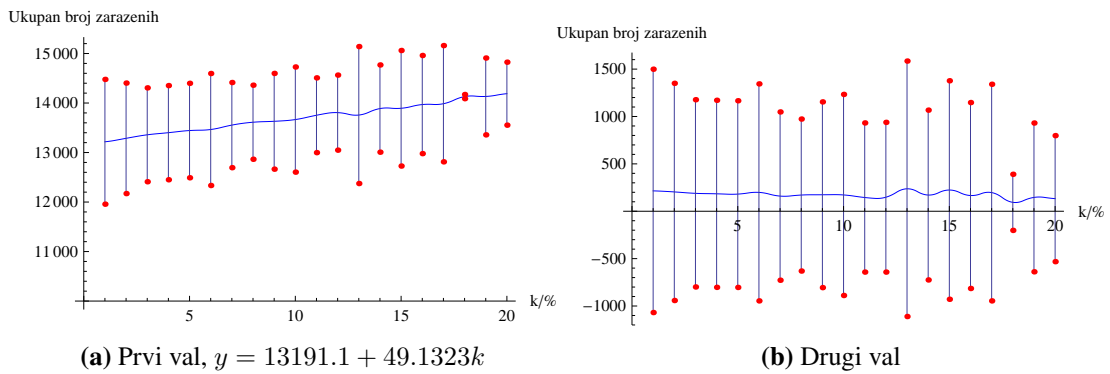
**Slika 3.5:** Mreža astrofizičara,  $m = 0.4$



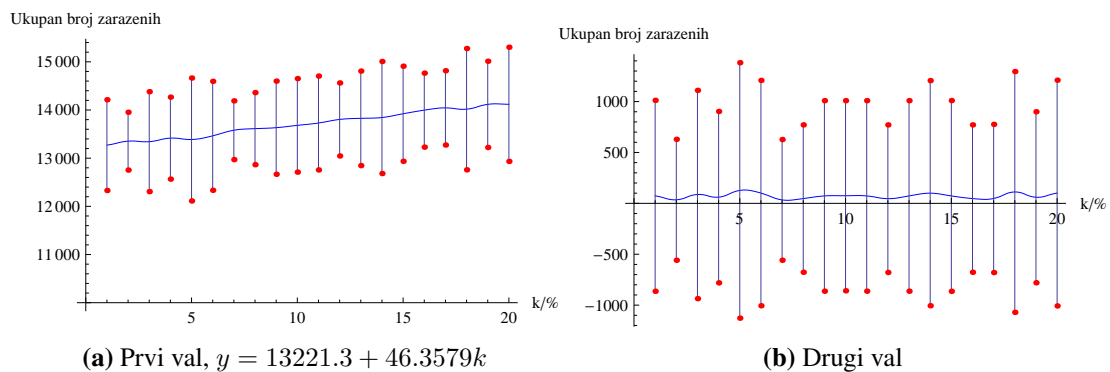
**Slika 3.6:** Mreža astrofizičara,  $m = 0.5$



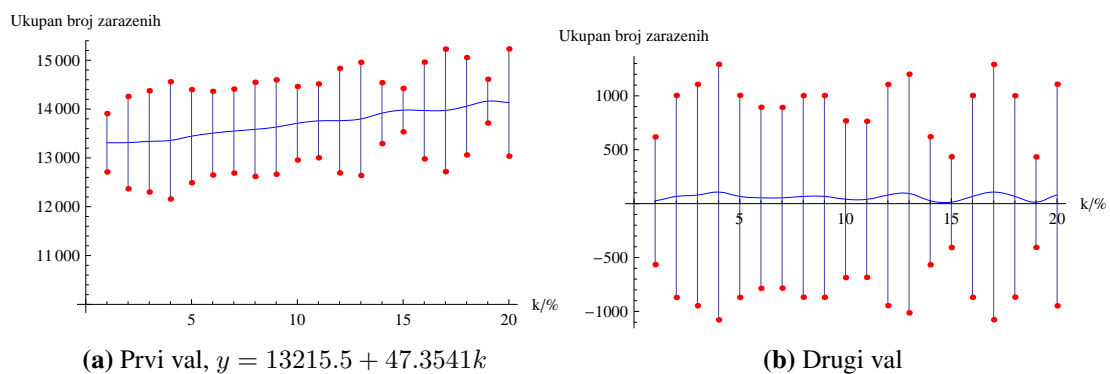
**Slika 3.7:** Mreža astrofizičara,  $m = 0.6$



**Slika 3.8:** Mreža astrofizičara,  $m = 0.7$



**Slika 3.9:** Mreža astrofizičara,  $m = 0.8$



**Slika 3.10:** Mreža astrofizičara,  $m = 0.9$

Kod mreža s visokim stupnjem povezanosti čvorova kao što je ova, okupljanja imaju linearan utjecaj na broj zaraženih kao što je vidljivo iz grafova prvog vala. Vjerojatnosti mutacije, a i odsustvo okupljanja, utječe na smanjenje broja zaraženih u drugom valu, te za vjerojatnost od 0.9 u drugom valu je vrlo mali zaraženih čvorova. Prosječna regresija iznosi  $13214.1 + 47.2117k$ .



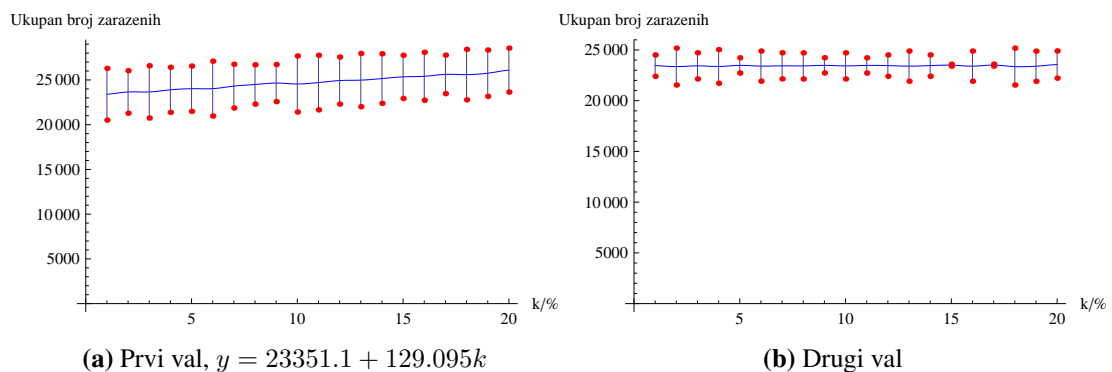
## 3.2. Mreža znanstvenika

Druga mreža predstavlja povezanosti znanstvenika u citiranju radova koji se tiču kondenziranih materija [14].

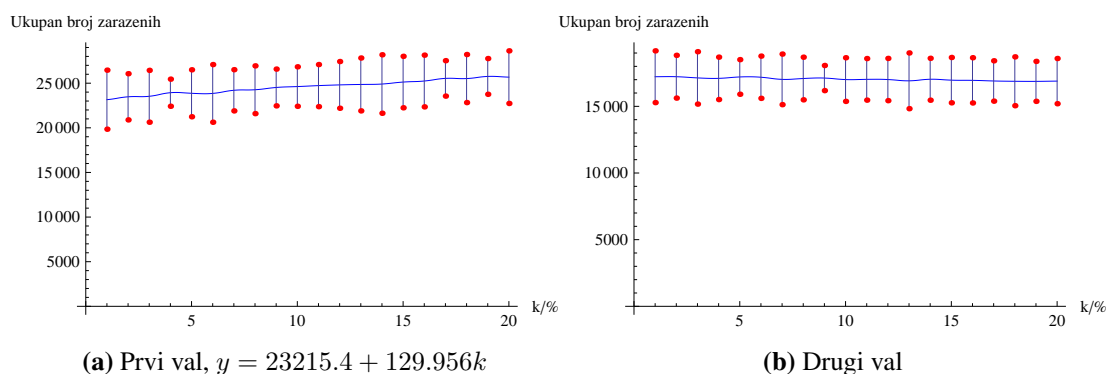
Mreža ima 31163 čvorova i 240058 veza. Prosječni stupanj čvora je 7.70. Početni čvor drugog vala (*pocetni\_cvor2*) ima id 782. Clustering koeficijent početnog čvora prvog vala je 0.0925926, a clustering koeficijent početnog čvora drugog vala 0.207207. U tablici 3.2 su ispisane vrijednosti broja zaraženih u slučaju kada se ne organiziraju okupljanja.

**Tablica 3.2:** Tablica 2

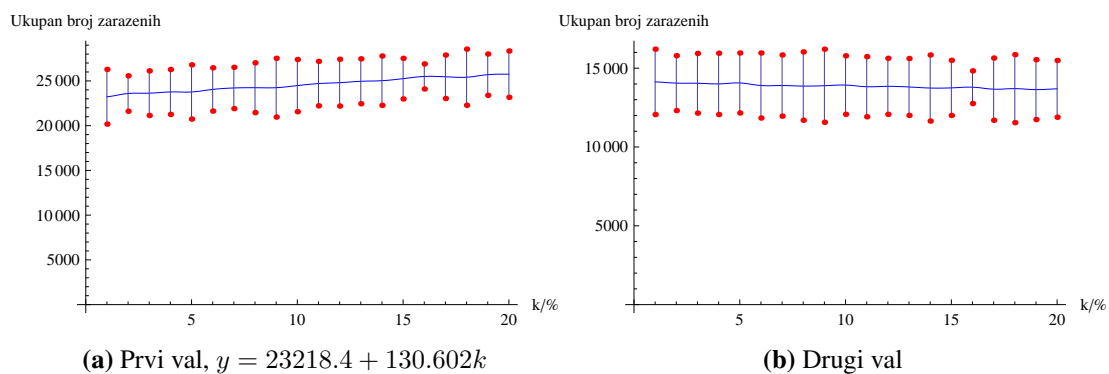
Vjerojatnost mutacije	Prvi val	Drugi val
$m = 0$	23253	23442
$m = 0.1$	23260	20317
$m = 0.2$	23243	17240
$m = 0.3$	23199	14122
$m = 0.4$	23183	11072
$m = 0.5$	23227	8027
$m = 0.6$	23209	5227
$m = 0.7$	23213	2655
$m = 0.8$	23202	766
$m = 0.9$	23235	275



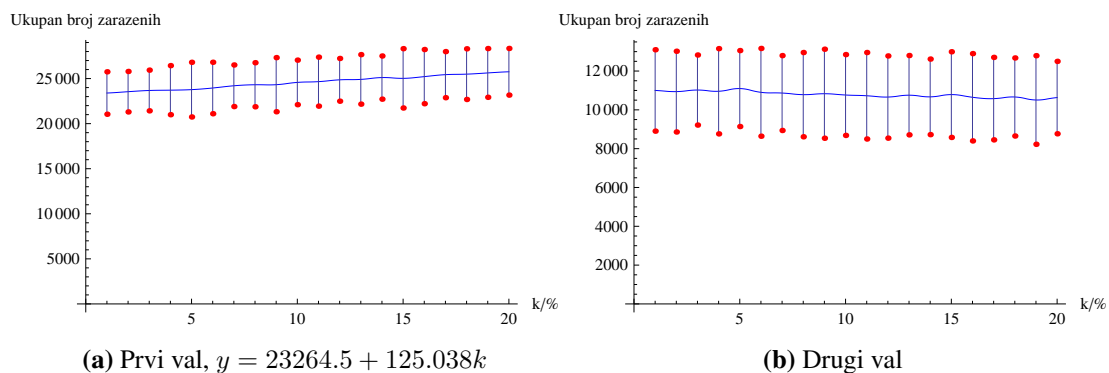
**Slika 3.11:** Mreža znanstvenika,  $m = 0.0$



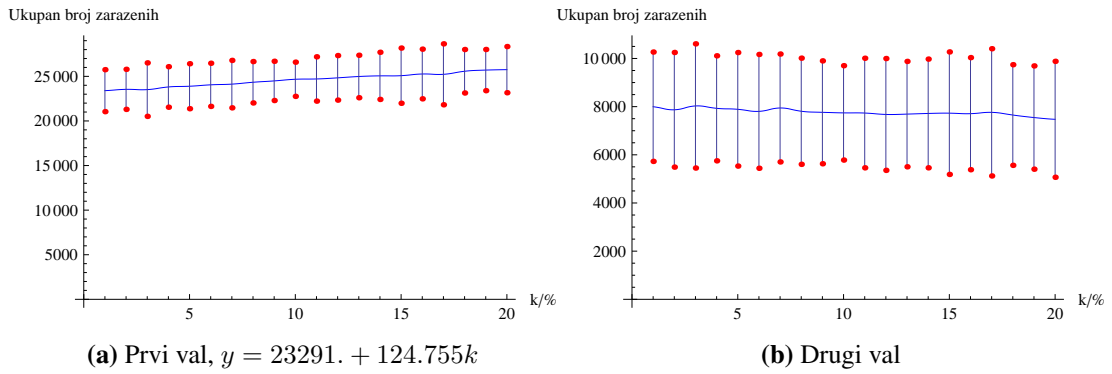
**Slika 3.12:** Mreža znanstvenika,  $m = 0.1$



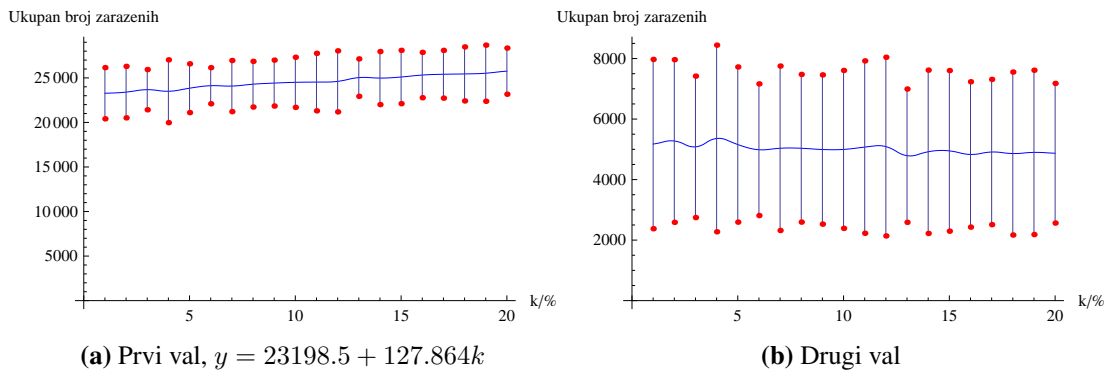
**Slika 3.13:** Mreža znanstvenika,  $m = 0.2$



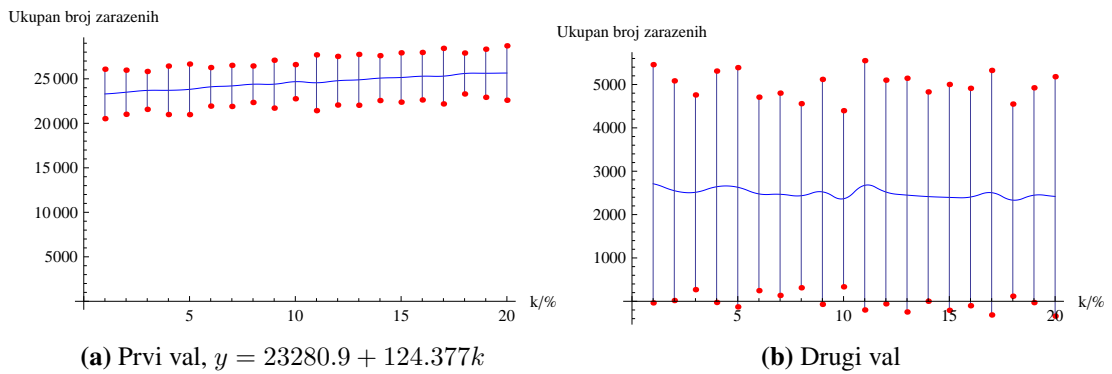
**Slika 3.14:** Mreža znanstvenika,  $m = 0.3$



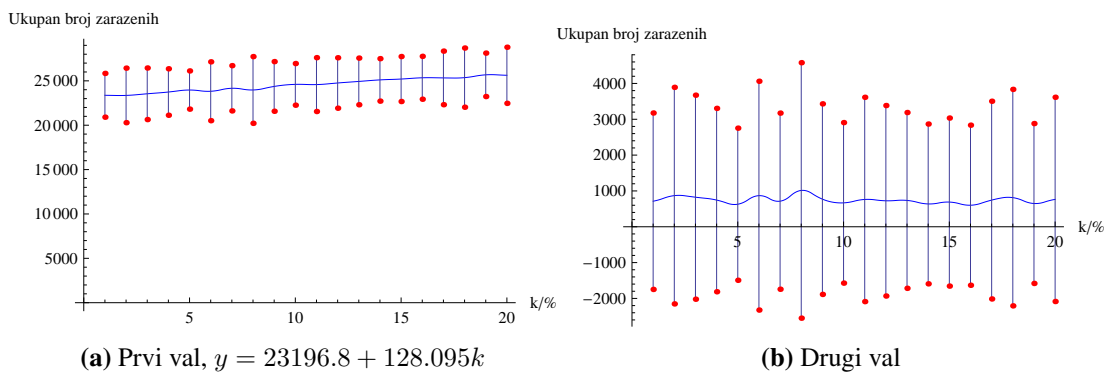
**Slika 3.15:** Mreža znanstvenika,  $m = 0.4$



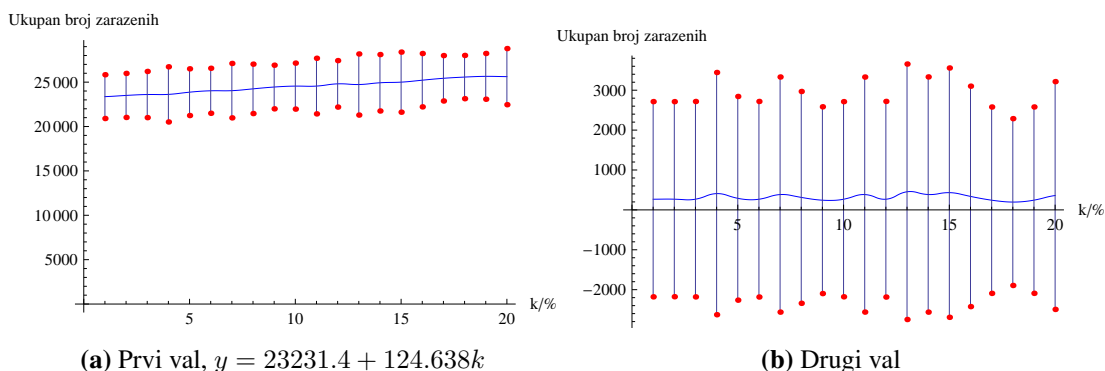
**Slika 3.16:** Mreža znanstvenika,  $m = 0.5$



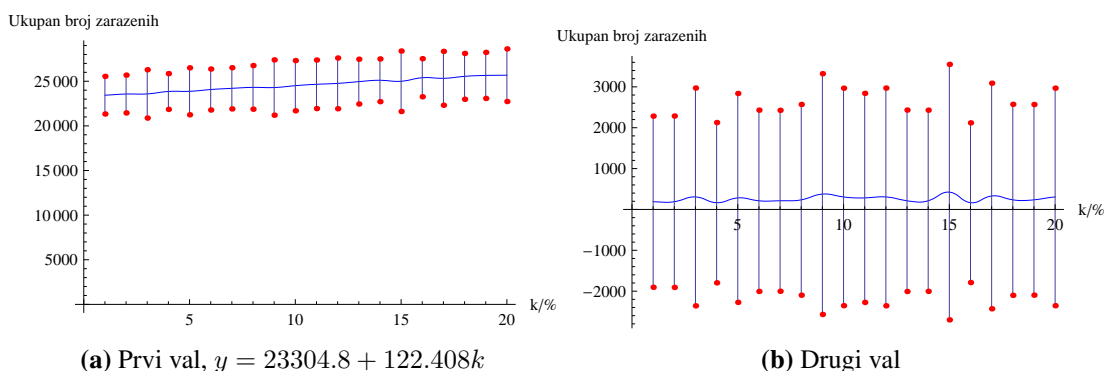
**Slika 3.17:** Mreža znanstvenika,  $m = 0.6$



**Slika 3.18:** Mreža znanstvenika,  $m = 0.7$



**Slika 3.19:** Mreža znanstvenika,  $m = 0.8$



**Slika 3.20:** Mreža znanstvenika,  $m = 0.9$

Ova mreža ima srednje visok stupanj povezanosti čvorova, no ponašanje algoritma zaraze je slično kao i na prethodnoj mreži. Iz grafova prvog vala se primjećuje linearan utjecaj okupljanja na ukupan broj zaraženih čvorova u tom valu. U drugom valu broj zaraženih opada s povećanjem koeficijenta mutacije, a također se gubi i svojstvo linearnog rasta zbog odsustva okupljanja. Prosječna regresija iznosi  $23255.3 + 126.683k$ .

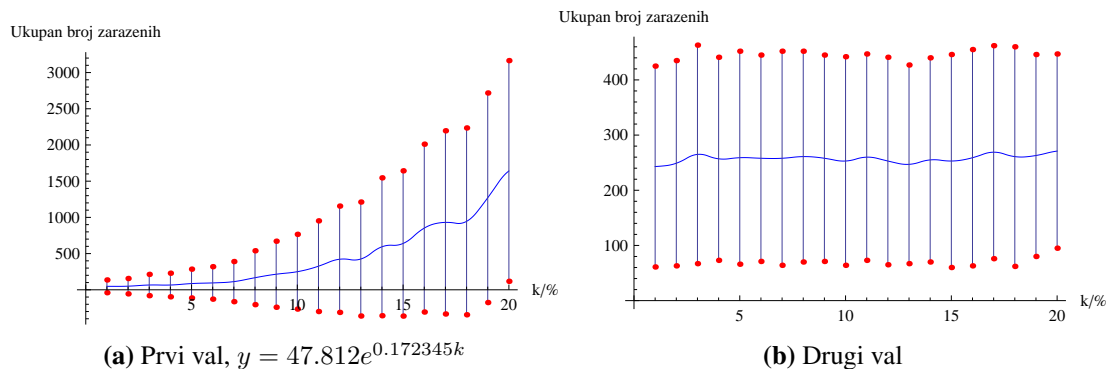
### 3.3. Energetska mreža

Treća mreža predstavlja topološku mrežu energetskog sustava Zapadnih država Sjedinjenih Američkih Država.

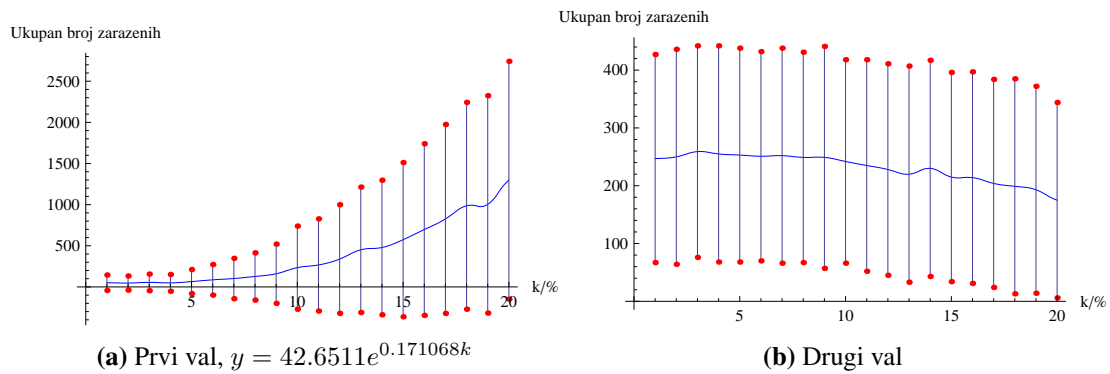
Mreža ima 4941 čvorova i 13188 veza. Prosječni stupanj čvora je 2.67. Početni čvor drugog vala (*pocetni\_cvor2*) ima id 782. Clustering koeficijent početnog čvora prvog vala je 0, a clustering koeficijent početnog čvora drugog vala 0.0643275. U tablici 3.3 su ispisane vrijednosti broja zaraženih u slučaju kada se ne organiziraju okupljanja.

**Tablica 3.3:** Tablica 3

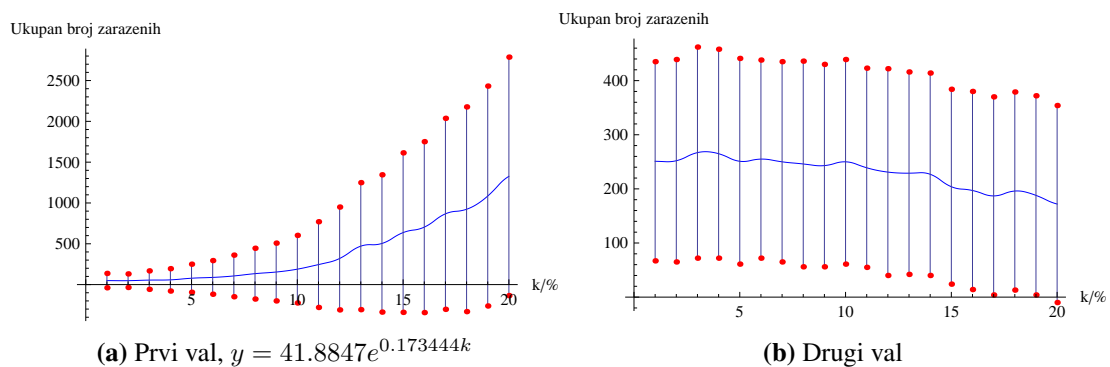
Vjerojatnost mutacije	Prvi val	Drugi val
$m = 0$	48	255
$m = 0.1$	49	257
$m = 0.2$	48	260
$m = 0.3$	49	259
$m = 0.4$	47	255
$m = 0.5$	48	258
$m = 0.6$	48	251
$m = 0.7$	49	251
$m = 0.8$	48	256
$m = 0.9$	47	256



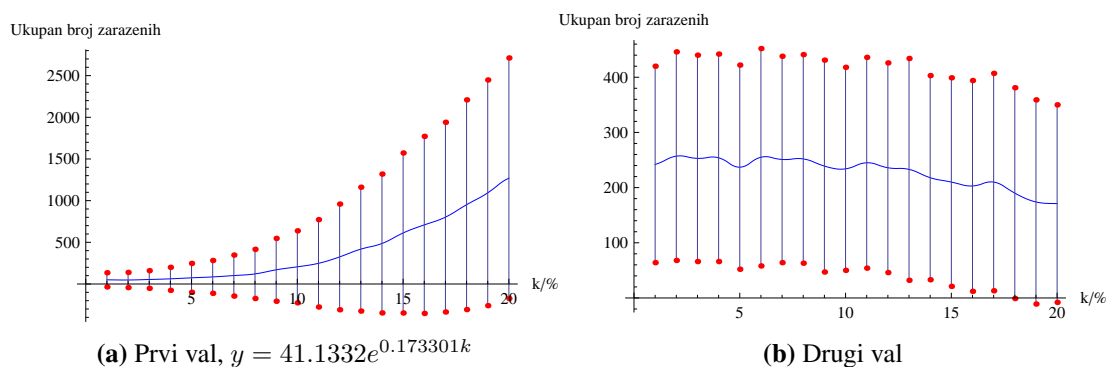
**Slika 3.21:** Energetska mreža,  $m = 0.0$



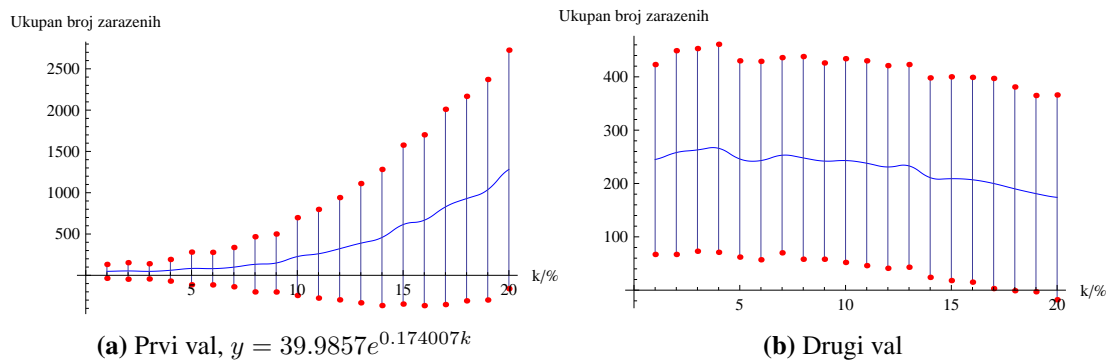
Slika 3.22: Energetska mreža,  $m = 0.1$



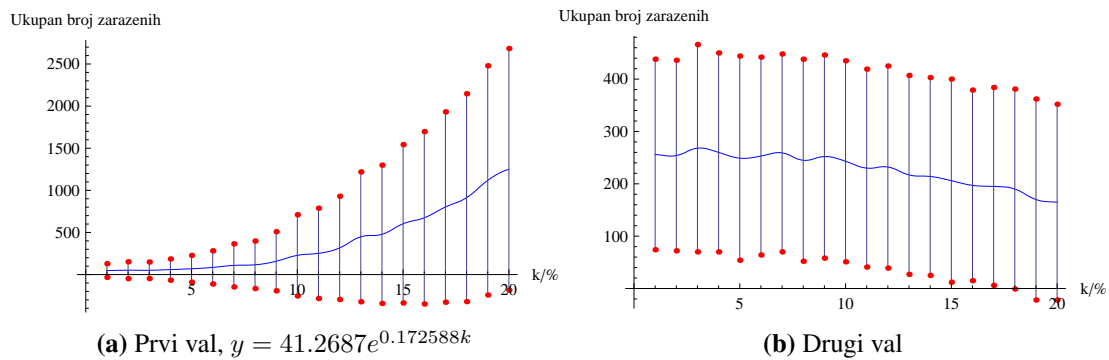
Slika 3.23: Energetska mreža,  $m = 0.2$



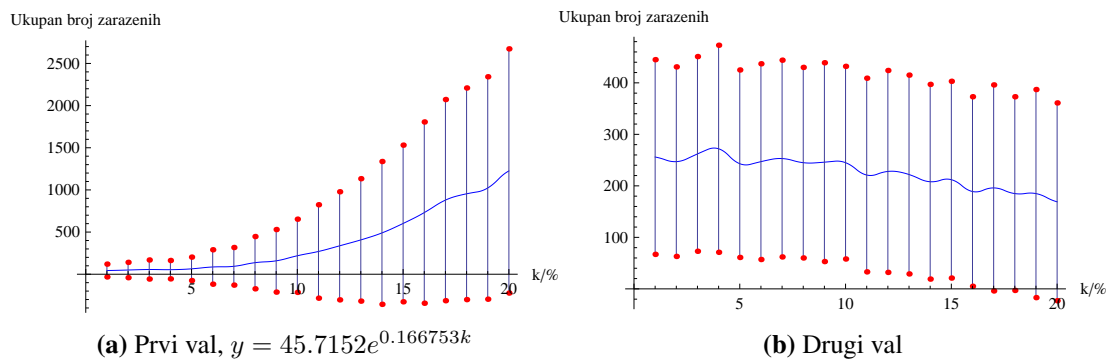
Slika 3.24: Energetska mreža,  $m = 0.3$



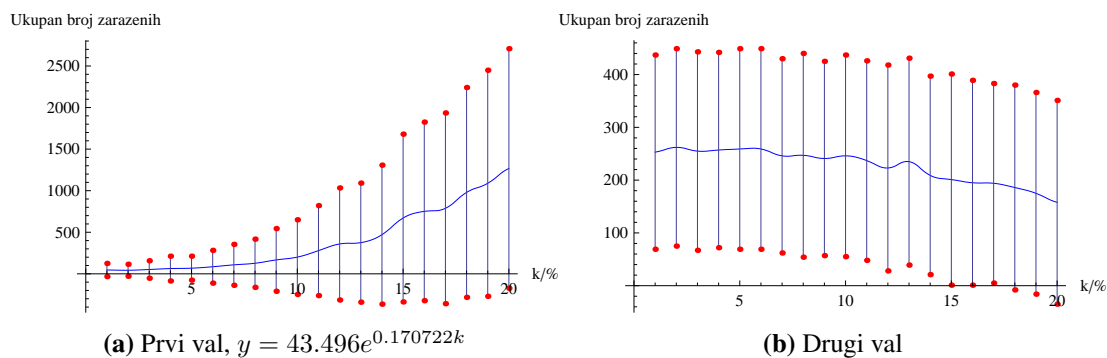
Slika 3.25: Energetska mreža,  $m = 0.4$



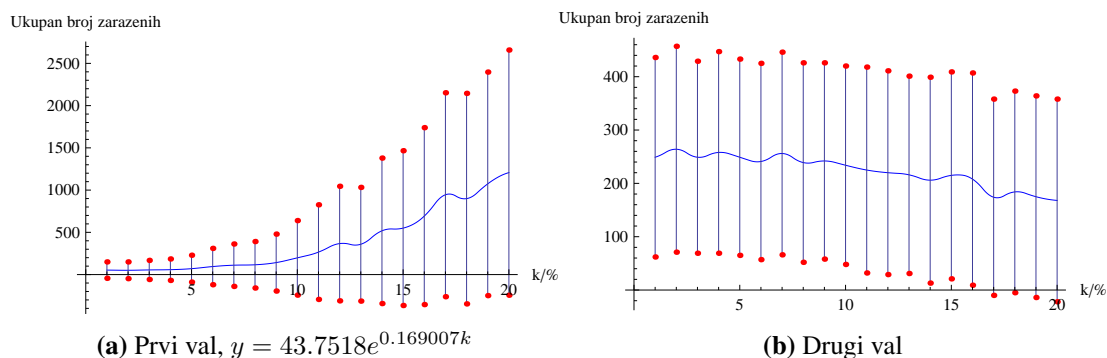
Slika 3.26: Energetska mreža,  $m = 0.5$



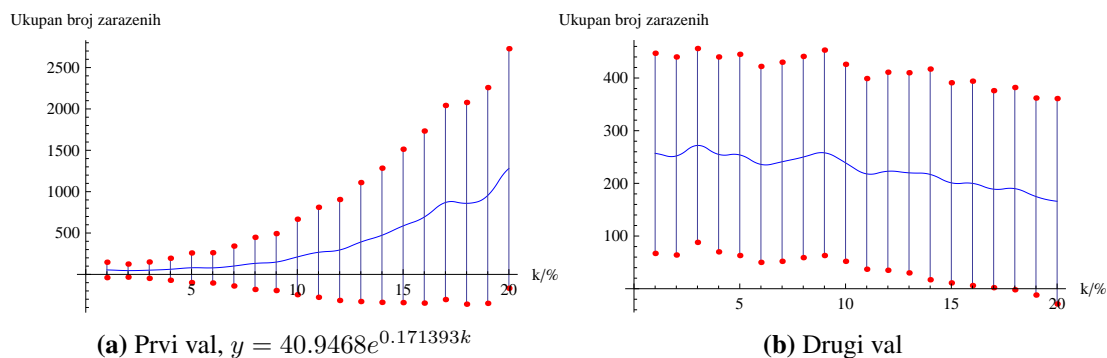
Slika 3.27: Energetska mreža,  $m = 0.6$



**Slika 3.28:** Energetska mreža,  $m = 0.7$



**Slika 3.29:** Energetska mreža,  $m = 0.8$



**Slika 3.30:** Energetska mreža,  $m = 0.9$

Za razliku od prethodne dvije mreže, ova ima nizak stupanj povezanosti čvorova, te je ovdje primjetan eksponencijalan rast broja zaraženih s povećanjem vjerojatnosti organizacije okupljanja u prvom valu. U drugom valu se broj zaraženih kreće na sličan način kao kod mreža s visokim stupnjem povezanosti—pad zaraženih zbog povećanja

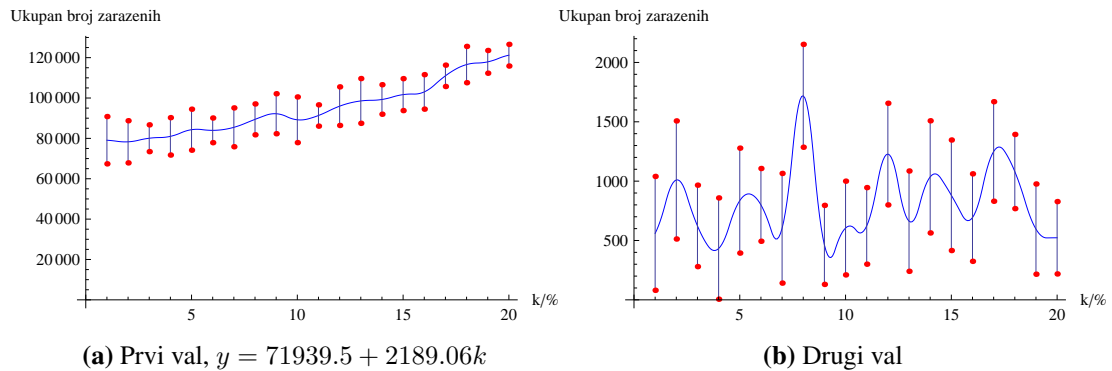


vjerojatnosti mutacije, te se gubi svojstvo rasta broja zaraženih zbog odsustva okupljanja.

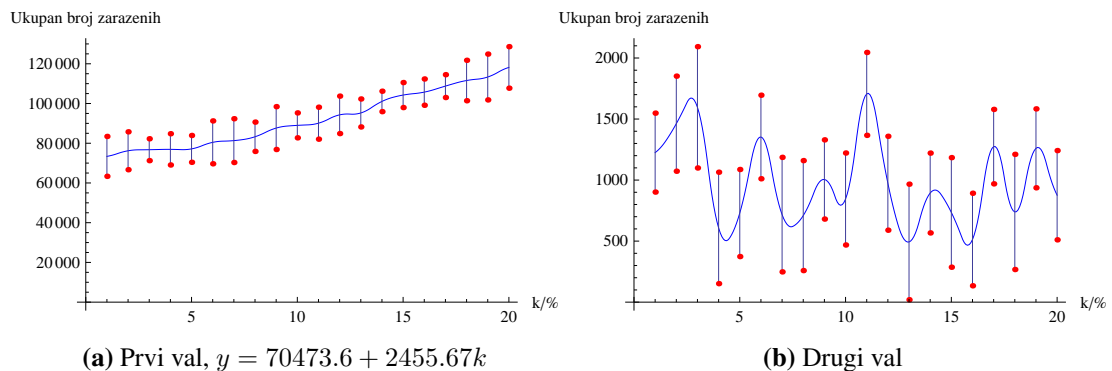
### 3.4. Mreža korisnika s [www.youtube.com](http://www.youtube.com) web stranice

Ovo je mreža koja modelira interakcije korisnika [www.youtube.com](http://www.youtube.com) web stranice. [10]

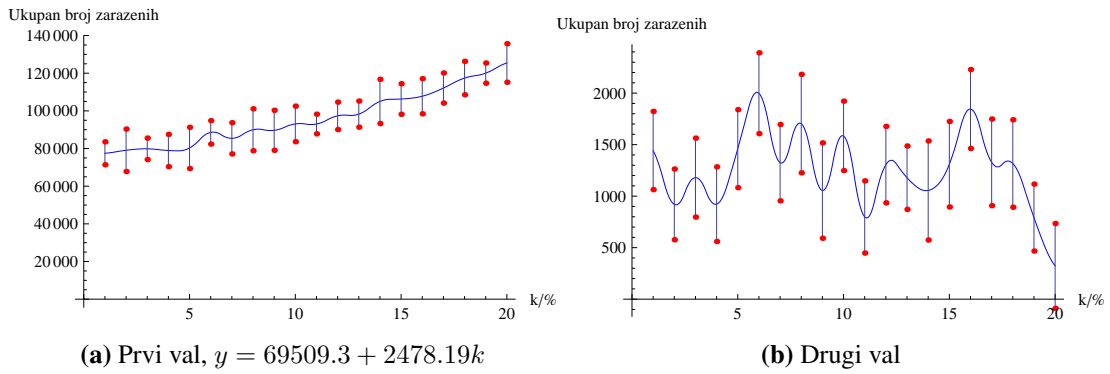
Mreža ima 1157827 čvorova i 9890764 veza. Prosječni stupanj čvora je 8.54. Početni čvor drugog vala (*pocetni\_cvor2*) ima id 555555. Zbog veličine mreže vrijeme izvođenja je drastično poraslo pa je vjerojatnost oporavka povećana na 0.2.



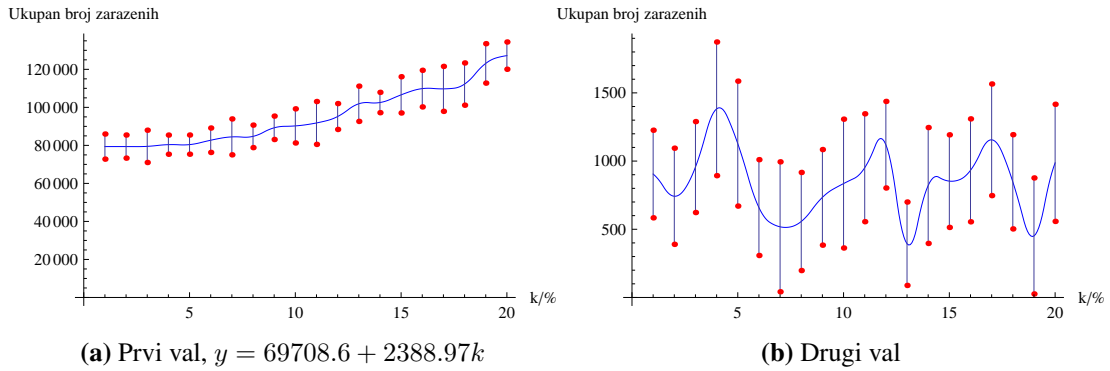
Slika 3.31: Youtube korisnici,  $m = 0.0$



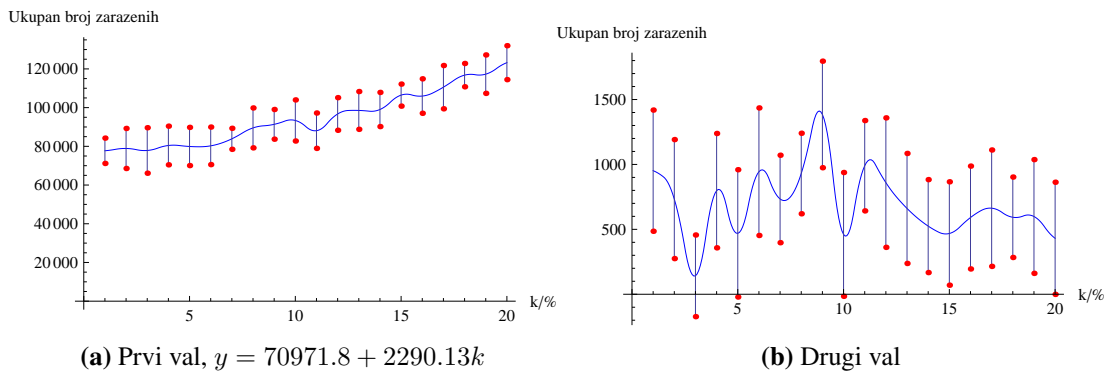
Slika 3.32: Youtube korisnici,  $m = 0.1$



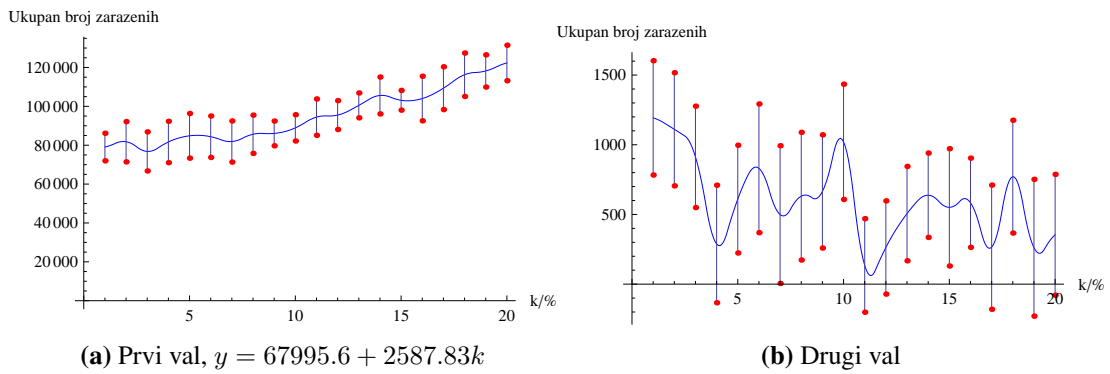
Slika 3.33: Youtube korisnici,  $m = 0.2$



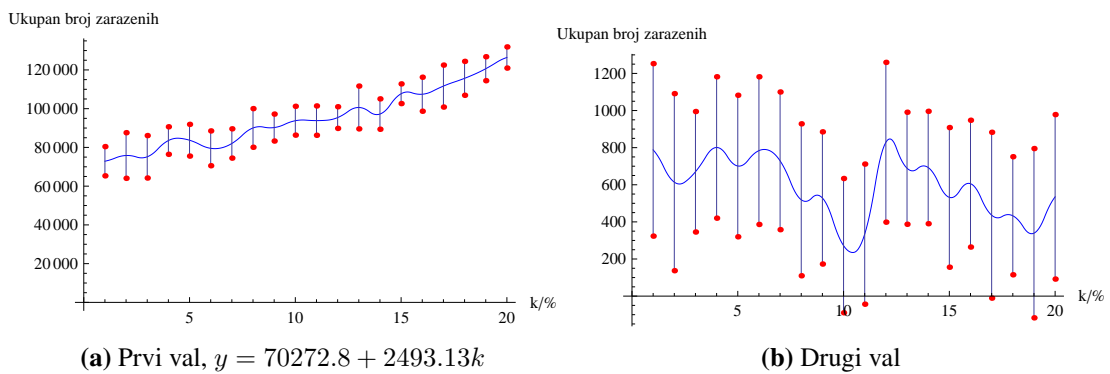
Slika 3.34: Youtube korisnici,  $m = 0.3$



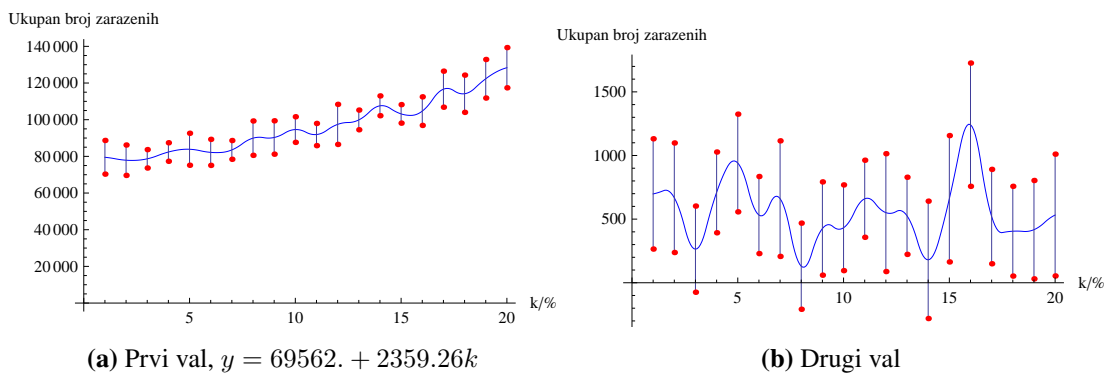
Slika 3.35: Youtube korisnici,  $m = 0.4$



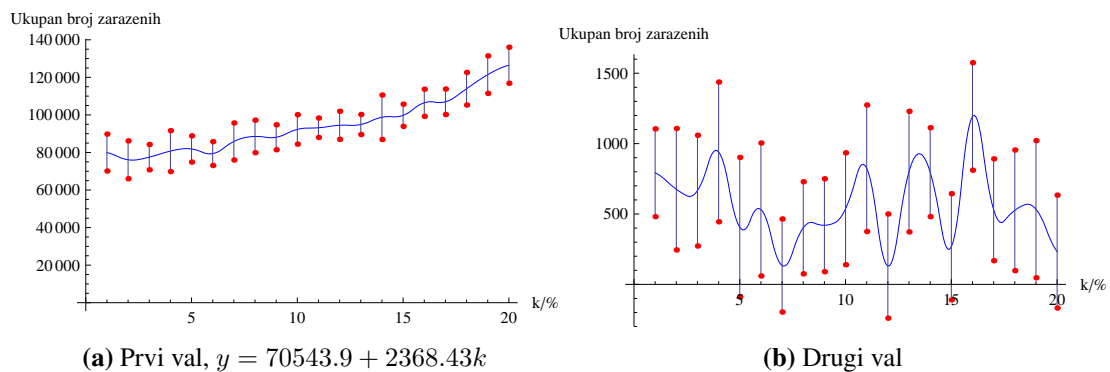
**Slika 3.36:** Youtube korisnici,  $m = 0.5$



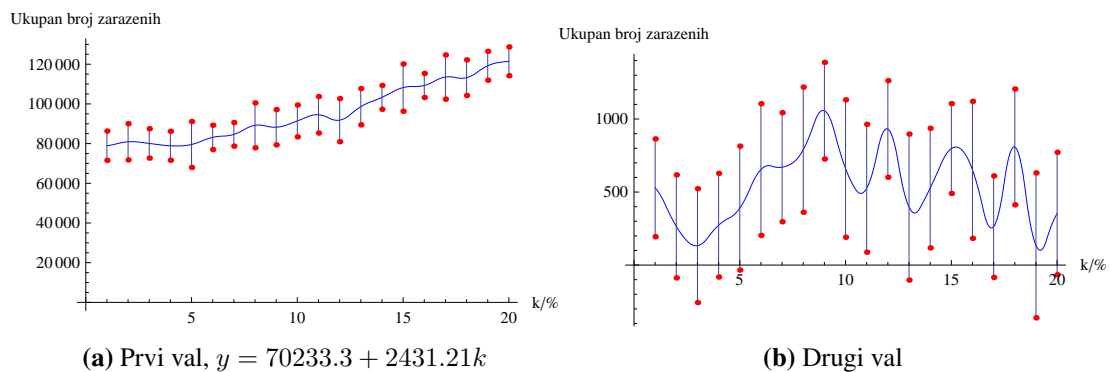
**Slika 3.37:** Youtube korisnici,  $m = 0.6$



**Slika 3.38:** Youtube korisnici,  $m = 0.7$



**Slika 3.39:** Youtube korisnici,  $m = 0.8$

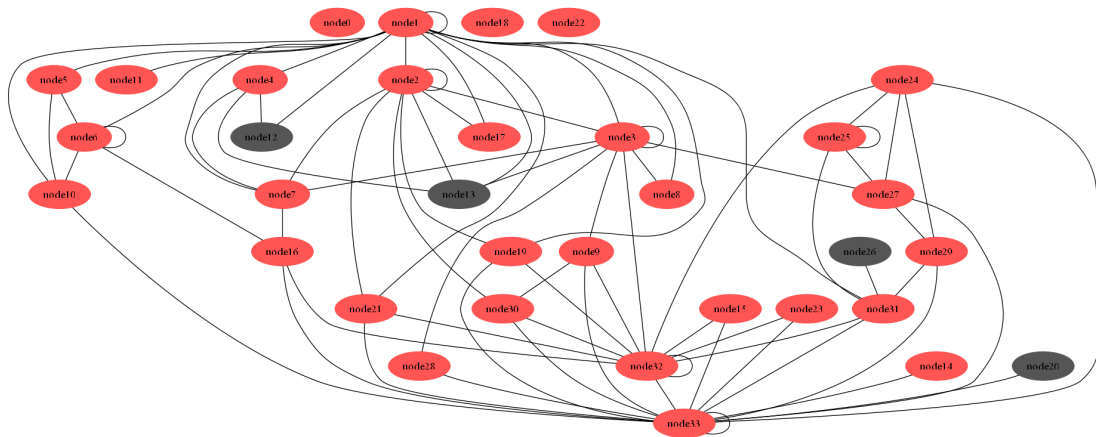


**Slika 3.40:** Youtube korisnici,  $m = 0.9$

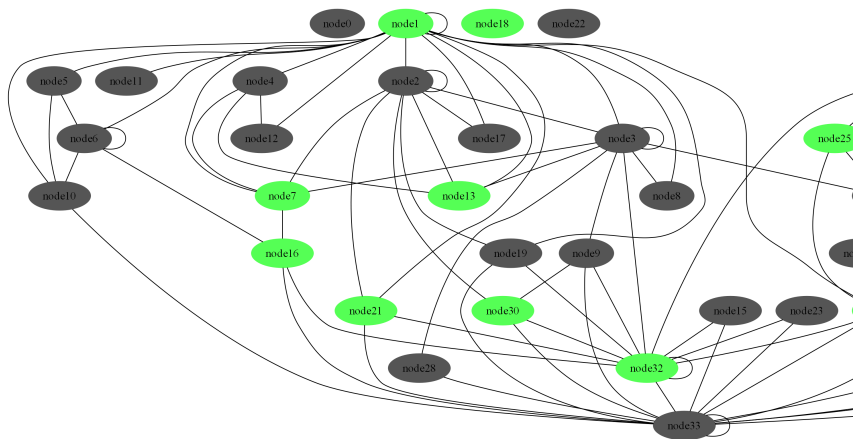
Ovdje se nastojalo provjeriti što se događa pri simulaciji širenja zaraze, okupljanja i mutacije u mrežama s vrlo velikim brojem čvorova. U prvom valu broj zaraženih se povećava linearno, dok su u drugom valu vidljive brojne oscilacije zbog visoke vjerojatnosti oporavka, a malog broja podložnih čvorova.

### 3.5. Grafički prikaz simulacije širenja zaraze

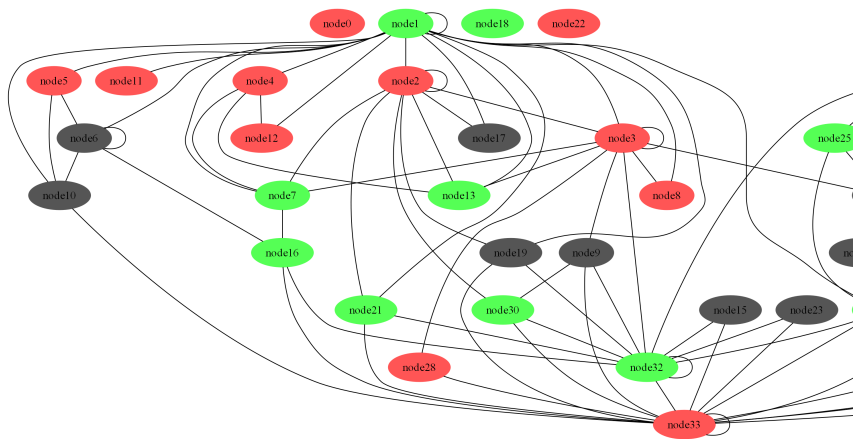
Na slikama 3.41, 3.43 i 3.42 je grafički [4] prikazan algoritam dan Pseudokodom 1. Slika 3.41 prikazuje stanje čvorova nakon prvog vala zaraze. Slika 3.42 prikazuje stanje nakon mutacije, te slika 3.43 prikazuje stanje nakon drugog vala zaraze. Crvenom bojom su prikazani zaraženi čvorovi, sivom podložni zarazi, a zelenom mutirani čvorovi. Mreža je svega 34 čvora i prikazuje povezanost članova nekog karate kluba [12].



Slika 3.41: Stanje čvorova nakon prvog vala.



Slika 3.42: Stanje čvorova nakon faze mutacije.



Slika 3.43: Stanje čvorova nakon drugog vala.

## 4. Paralelizacija

Paralelno programiranje je posebna grana računarstva u kojoj se koriste posebne metode za razdvajanje algoritama ili problema na njegove osnovne dijelove i izvode se istovremeno na tzv. *cluster*-ima. Cluster je skup usko povezanih računala koja rade zajedno tako da se mogu gledati kao jedno računalo. Ovim načinom izvođenja programa moguće je smanjiti vrijeme koje je potrebno za pronalaženje rješenja. Razmatrana je mogućnost paralelizacije programa upotrebom GPU i CUDA SDK.

Zbog kompleksnosti struktura podataka, te samog algoritma, nije moguće potpuno iskoristiti mogućnosti paralelizacije korištenjem CUDA SDK, stoga se postupak paralelizacije ostvario modifikacijom osnovnog algoritma na način da se umjesto standardnog (ANSI C++) načina generiranja pseudoslučajnih brojeva pri modeliranju koeficijenta zaraze, oporavka i mutacije svakog čvora, koristio algoritam *MersenneTwister* [9] koji je pogodan za paralelizaciju pomoću CUDA SDK. Usporedba je napravljena na službenom primjeru NVIDIA-e i C++ implementaciji [8]. Algoritam je dobio nezantno ubrzanje na CUDA SDK tehnologiji.

## 5. Zaključak

Proučavanje širenja zaraze, te raznih modela mreža i njihovih svojstava je od iznimne važnosti za epidemiologiju. U velikom broju zaraznih bolesti postoje dva vala zaraze. Nakon prvog vala, virus često mutira i to zna dovesti do katastrofalnih posljedica za populaciju. Ovdje se nastojalo modelirati takva dva vala zaraze, te proučiti utjecaj organizacije okupljanja u prvom valu zaraze na ukupan broj zaraženih u prvom i drugom valu. Simulacije su iskazale očekivane rezultate. Kod mreža s visokim stupnjem povezanosti čvorova, utjecaj organizacije okupljanja ima blago linearan utjecaj na ukupan broj zaraženih što je i logično jer je vjerojatnost sastanka dvije jedinice dovoljno visoka. Eksponencijalni rast ukupnog broja zaraženih u prvom valu je karakterističan za mreže s niskim stupnjem povezanosti gdje okupljanja uvelike pomažu širenje zaraze. Nakon prvog vala mreže s visokim stupnjem povezanosti, a također i one s niskim stupnjem povezanosti, iskazuju slično ponašanje. Mutacijom zaraženih jedinki, oni postaju otporni u drugom valu, te se broj zaraženih smanjuje. Ukoliko se dobivena svojstva iskoriste na pravilan način, moguće je uvelike smanjiti ukupan broj zaraženih jedinki te tako oslabiti širenje virusa.



# LITERATURA

- [1] *Clustering coefficient*, 2010. URL [http://en.wikipedia.org/wiki/Clustering\\_coefficient](http://en.wikipedia.org/wiki/Clustering_coefficient).
- [2] *Complex Networks*, 2010. URL [http://en.wikipedia.org/wiki/Complex\\_network](http://en.wikipedia.org/wiki/Complex_network).
- [3] Nino Antulov-Fantulin. *Utjecaj zaraze na svojstva kompleksne mreže*, 2008.
- [4] Emden Gansner Yifan Hu-Stephen North Yehuda Koren Vladimir Alexiev David Dobkin Eleftherios Koutsofios Bruce Lilly Glen Low John Mocenigo Jeroen Scheerder Daniel Richard G. Gordon Woodhull Arif Bilgin, John Ellson. *GraphViz picture format*, 1993. URL <http://www.graphviz.org/>.
- [5] Barthelemy Marc Valleron Alain-Jacques Vespignani Alessandro Colizza Vittoria, Barrat Alain. Modeling the worldwide spread of pandemic influenza: Baseline case and containment interventions. *Journal*, 2007.
- [6] Lang Moore David Smith. *The SIR Model for Spread of Disease*, 2000. URL <http://www.math.duke.edu/education/ccp/materials/diffcalc/sir/index.html>.
- [7] Michael Himsolt. *GML format datoteka*, 2010. URL <http://www.infosun.fim.unipassau.de/Graphlet/GML/gml-tr.html>.
- [8] Charles Karney. *Implementacija Mersenne twister algoritma u C++*, 2010. URL <http://www.math.sci.hiroshima-u.ac.jp/~m-mat/MT/VERSIONS/C-LANG/c-lang.html>.
- [9] Takuji Nishimura Makoto Matsumoto. *Mersenne twister algoritam*, 2010. URL [http://en.wikipedia.org/wiki/Mersenne\\_twister](http://en.wikipedia.org/wiki/Mersenne_twister).
- [10] Alan Mislove, Massimiliano Marcon, Krishna P. Gummadi, Peter Druschel, i Bobby Bhattacharjee. Measurement and Analysis of Online Social Networks. U

*Proceedings of the 5th ACM/Usenix Internet Measurement Conference (IMC'07)*,  
San Diego, CA, October 2007.

- [11] M. E. J. Newman. *Proc. Natl. Acad. Sci. USA* 98, 404-409, 2001.
- [12] Mark Newman. *GML datoteke mreža*, 2010. URL <http://www-personal.umich.edu/~mejn/netdata/>.
- [13] NVIDIA. *CUDA SDK*, 2010. URL [http://www.nvidia.com/object/cuda\\_home\\_new.html](http://www.nvidia.com/object/cuda_home_new.html).
- [14] Vespignani A. Satorras R. *Epidemics and immunization in scale-free networks*, 2002. URL [ArXiv:cond-mat/0205260v1](http://arxiv.org/abs/cond-mat/0205260v1).

## **Simulacija utjecaja okupljanja zaraženih gripom na širenje epidemije**

### **Sažetak**

Širenje zaraze se vrlo dobro može modelirati na kompleksnim mrežama. Ovdje je odabran prošireni SIR model mreže, te su simulirana dva vala zaraze između kojih nastupa faza mutacije. Okupljanja su organizirani susreti zaraženih i zdravih jedinki u svrhu zaraze koja se odvijaju u prvom valu zaraze. Njihov utjecaj na ukupan broj zaraženih ima linearan rast kod mreža s visokim stupnjem povezanosti čvorova, a eksponencijalan rast kod mreža s niskim stupnjem povezanosti čvorova. Faza mutacije utječe na smanjenje ukupnog broja zaraženih u drugom valu zaraze.

**Ključne riječi:** kompleksne mreže, okupljanja, mutacije

## **Simulation of the impact organizing parties on propagation of viruses at spreading the infection**

### **Abstract**

The spread of infection can be very well modeled using complex networks. The model presented here is the extended SIR network model on which two waves of infection have been simulated, with a mutation phase between them. Meetings are organized gatherings of healthy and infected individuals with the purpose of infection occurring during the first wave. While the impact on the total number of infected cells has a linear increase in networks with a high degree of connection between nodes, networks with a low degree of connection are characterized with exponential growth. The mutation phase causes the reduction of the total number of infected cells in the second wave.

**Keywords:** complex networks, parties, mutations