

Ανάλυση Κοινωνικών Δικτύων (Social Network Analysis)

3^η Εργαστηριακή Άσκηση

Συμεών Παπαβασιλείου (papavass@mail.ntua.gr)
Βασίλειος Καρυώτης (vassilis@netmode.ntua.gr)

09 Ιανουαρίου, 2017

OneMax Problem

- Βρείτε τη **δυαδική** ακολουθία (x_1, x_2, \dots, x_n) που μεγιστοποιεί το άθροισμα $x_1 + x_2 + \dots + x_n$, $n=20$, χρησιμοποιώντας τις συναρτήσεις του MATLAB, για γενετικούς αλγορίθμους.
- ga.m
- optimtool('ga')
- **Ποιότητα της λύσης:** πόσο κοντά είναι η λύση του γενετικού αλγορίθμου στην προφανή βέλτιστη λύση.
- Πληθυσμός από 10 έως 200 χρωμοσώματα με βήμα 10.
- Πιθανότητα διασταύρωσης από 0.3 έως 0.9 με βήμα 0.1.
- Πιθανότητα μετάλλαξης από 0.01 ως 0.2 με βήμα 0.01.
- Εξετάστε την ποιότητα της λύσης του γενετικού αλγορίθμου **για κάθε συνδυασμό** μεγέθους πληθυσμού, πιθανότητας μετάλλαξης και πιθανότητας διασταύρωσης που προκύπτει.

Application in Community Detection (1)

Social Network: $G=(V,E)$

Represented by an Adjacency Matrix: $A=[a_{ij}]$

Problem:

Find communities in the social network 

Find a partitioning of A into sub-matrices

Let $S = (I, J)$ be sub-matrix of A , where I is a subset of the rows $\{I_1, \dots, I_N\}$ of A , and J is a subset of the columns $\{J_1, \dots, J_N\}$ of A .

Let a_{iJ} denote the *mean value* of the i th row of the S , and a_{Ij} the mean of the j th column of S .

$$a_{iJ} = \frac{1}{|J|} \sum_{j \in J} a_{ij}, \text{ and } a_{Ij} = \frac{1}{|I|} \sum_{i \in I} a_{ij}$$

The *volume* v_S of a sub-matrix $S = (I, J)$ is the number of 1 entries a_{ij} such that $i \in I$ and $j \in J$, that is $v_S = \sum_{i \in I, j \in J} a_{ij}$.

Application in Community Detection (2)

Given a sub-matrix $S = (I, J)$, the *power mean of S of order r* , denoted as $\mathbf{M}(S)$ is defined as

$$\mathbf{M}(S) = \frac{\sum_{i \in I} (a_{iJ})^r}{|I|}$$

Definition of the fitness function

The *score* of S is defined as $Q(S) = \mathbf{M}(S) \times v_S$.

The *community score* of a partitioning $\{S_1, \dots, S_k\}$ of A is defined as

$$CS = \sum_i^k Q(S_i)$$

Εντοπισμός Κοινοτήτων σε Γράφους Κοινωνικών Δικτύων με Χρήση Γενετικών Αλγορίθμων (3)

Αναπαράσταση υποψήφιων λύσεων (χρωμοσωμάτων)

Κάθε χρωμόσωμα $b_{i,t}$ του πληθυσμού τη χρονική στιγμή t είναι ένα διάνυσμα με μήκος ίσο με το πλήθος των κόμβων του δικτύου, n . Αν $b_{i,t}(j) = d$ τότε οι κόμβοι j, d ανήκουν στην ίδια κοινότητα. Επιπλέον για να ισχύει $b_{i,t}(j) = d$, οι κόμβοι j, d θα πρέπει να συνδέονται στην αρχική τοπολογία. Τότε, βρίσκοντας τις συνδεδεμένες συνιστώσες (FindComponents.m) του χρωμοσώματος $b_{i,t}$, βρίσκουμε τις κοινότητες του δικτύου.

Αρχικοποίηση

Η αρχική γενιά του πληθυσμού θα πρέπει να αποτελείται από χρωμοσώματα τα οποία θα είναι τυχαία επιλεγμένα και διορθωμένα ώστε ο αριθμός που υπάρχει σε κάθε θέση να είναι γείτονας του αντίστοιχου κόμβου. (Αν δεν ισχύει θα πρέπει να αντικατασταθεί από ένα γείτονα του αντίστοιχου κόμβου.) Θεωρείστε πληθυσμό 300 χρωμοσωμάτων.

Συνάρτηση Fitness

Η συνάρτηση fitness υπολογίζεται όπως στις διαφάνειες 28 και 29 της αντίστοιχης διάλεξης του μαθήματος. Προσοχή για κάθε συνιστώσα (component) του γράφου θα πρέπει να βρίσκετε τον αντίστοιχο υπογράφο χρησιμοποιώντας τη συνάρτηση subgraph.m.

Εντοπισμός Κοινοτήτων σε Γράφους Κοινωνικών Δικτύων με Χρήση Γενετικών Αλγορίθμων (4)

Επιλογή (Selection)

Η επιλογή χρωμοσωμάτων από μία γενιά του πληθυσμού για την κατασκευή της επόμενης γενιάς γίνεται με τη μέθοδο της ρουλέτας (διαφάνειες μαθήματος και αλγόριθμος). Τροποποιήστε τον παραπάνω αλγόριθμο ώστε να εφαρμόσετε και ελιτισμό, δηλαδή τα x πρώτα χρωμοσώματα της νέας γενιάς θα είναι εκείνα τα χρωμοσώματα της προηγούμενης γενιάς που έχουν επιτύχει τη μέγιστη τιμή της συνάρτησης fitness.

Διασταύρωση (Crossover)

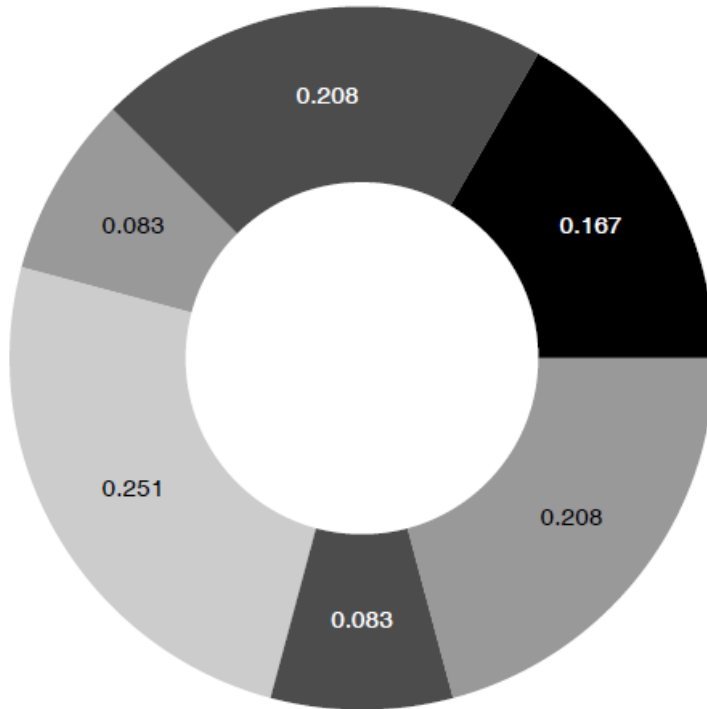
Η διασταύρωση πραγματοποιείται όπως περιγράφεται στον αλγόριθμο, μεταξύ διαδοχικών χρωμοσωμάτων της εκάστοτε γενιάς. Η πιθανότητα διασταύρωσης ανά ζεύγος είναι ίση με p_c .

Μετάλλαξη (Mutation)

Η μετάλλαξη πραγματοποιείται σε κάθε θέση του χρωμοσώματος με πιθανότητα p_m , όπως στον αλγόριθμο. Θα πρέπει ωστόσο ο αλγόριθμος να τροποποιηθεί ώστε να δίνει ένα τυχαίο γείτονα του κόμβου που αντιστοιχεί σε κάθε θέση ως αποτέλεσμα της μετάλλαξης.

Selection – Algorithmic Description (5)

- Generalized roulette game: in a roulette game, the slots are not equally wide, i.e. the different outcomes can occur with different probabilities.



Select an individual – chromosome as in the scheme of the previous slide.

```
 $x := \text{Random}[0, 1];$   
 $i := 1$ 
```

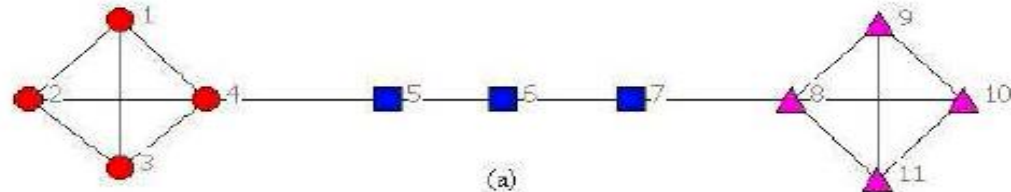
```
WHILE  $i < m$  &  $x < \sum_{j=1}^i f(b_{j,t}) / \sum_{j=1}^m f(b_{j,t})$  DO  
   $i := i + 1;$ 
```

```
select  $b_{i,t};$ 
```

Application in Community Detection (6)

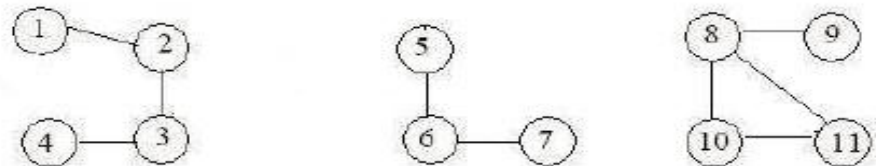
Representation:

- Each individual is a vector of size N (number of nodes), where the value j at position i means a link between nodes (i,j) .
- Such links should exist in the original network.
- Initialization via “safe” individuals i.e. check and correct if the links do not exist. If the value at position i is k and the link (i,k) does not exist, replace this link with a neighbor of i .
- Find communities represented by each individual via finding connected components.



| | | | | | | | | | | | |
|----------|---|---|---|---|---|---|---|----|---|----|----|
| Position | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
| Genotype | 2 | 1 | 2 | 3 | 6 | 5 | 6 | 10 | 8 | 11 | 8 |

(b)



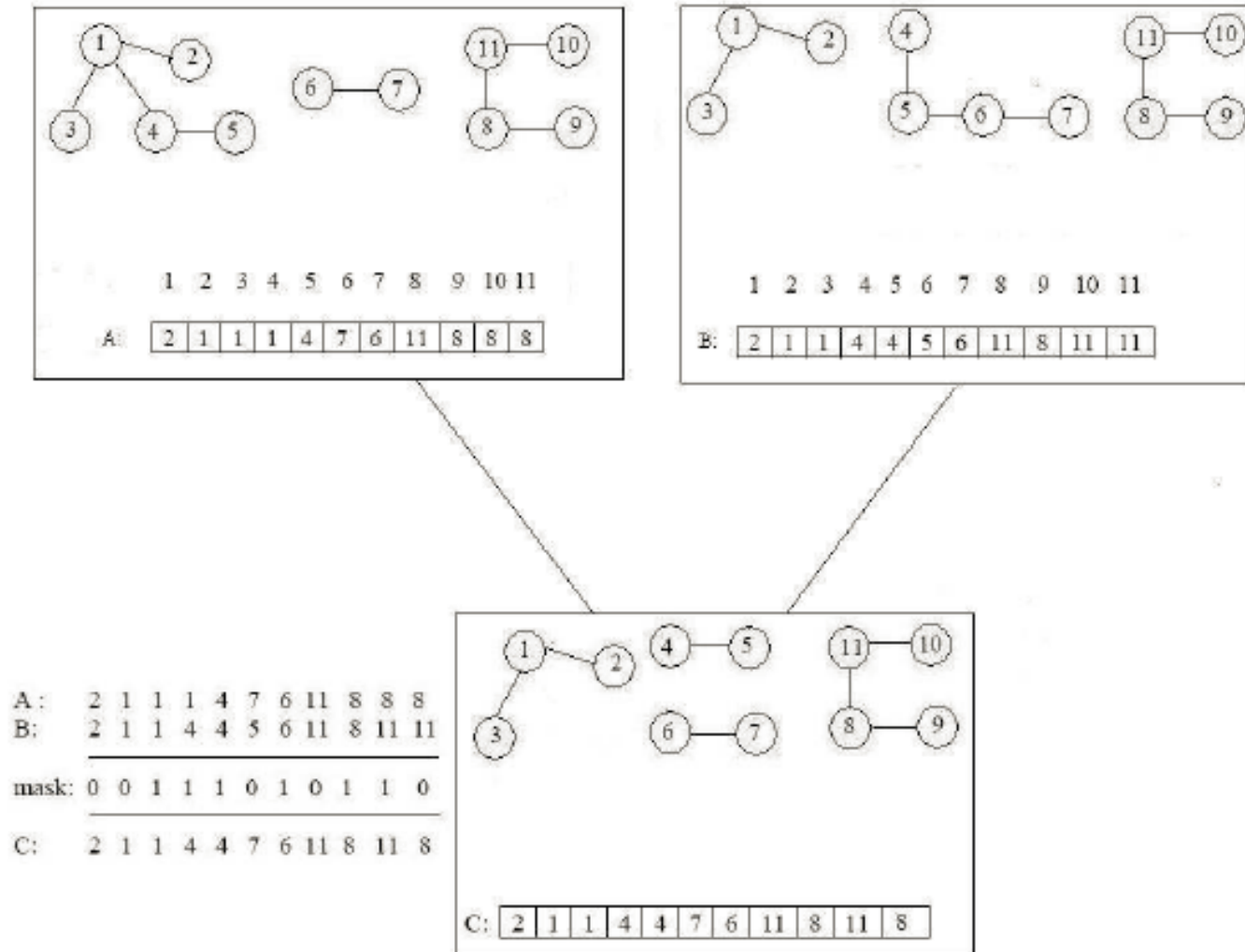
(c)

Mutation

- At position i only with the neighbors of i .

Application in Community Detection (7)

Uniform Crossover: exchange randomly genes of safe parents.



Εντοπισμός Κοινοτήτων σε Γράφους Κοινωνικών Δικτύων με Χρήση Γενετικών Αλγορίθμων (8)

Κριτήριο Τερματισμού (Stopping Criterion)

Ως κριτήριο τερματισμού ορίστε ένα μέγιστο πλήθος επαναλήψεων εφαρμογής του αλγορίθμου (γενεών πληθυσμού), ίσο με 30, ή ότι η βέλτιστη τιμή της συνάρτησης fitness δεν έχει μεταβληθεί για τις τελευταίες 5 επαναλήψεις.

Εφαρμόστε τον παραπάνω γενετικό αλγόριθμο στις πραγματικές τοπολογίες της δεύτερης εργαστηριακής άσκησης που δίνονται και στον Πίνακα 1. Οι παράμετροι που θα πρέπει να εξεταστούν δίνονται στον παρακάτω πίνακα. Προσοχή θα πρέπει να εξεταστούν όλοι οι συνδυασμοί.

| Παράμετρος | Τιμές |
|-------------------|---------------------|
| Διασταύρωση p_c | 0.7:0.9 με βήμα 0.1 |
| Μετάλλαξη p_m | 0.1, 0.2 |
| Ελιτισμός x | 1:3 με βήμα 1 |

Εντοπισμός Κοινοτήτων σε Γράφους Κοινωνικών Δικτύων με Χρήση Γενετικών Αλγορίθμων (9)

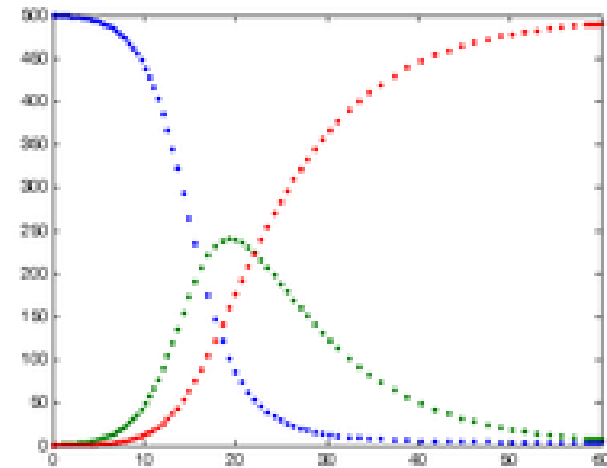
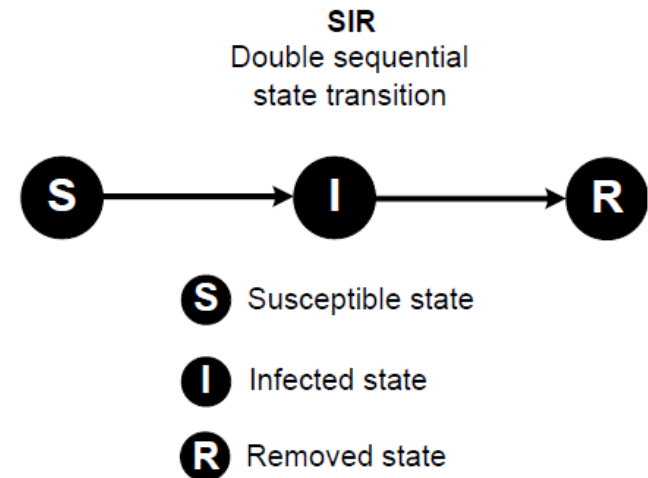
- Συγκρίνετε το αποτέλεσμα του γενετικού αλγορίθμου (το καλύτερο για τις διάφορες επιλογές παραμέτρων) με εκείνο των αλγορίθμων εντοπισμού κοινοτήτων της εργαστηριακής άσκησης 2 (δίνονται και στον Πίνακα 2), υπολογίζοντας το **modularity** χρησιμοποιώντας τη συνάρτηση **QFModul.m**.
- Θα πρέπει να έχετε πρωτίστως μετατρέψει το αποτέλεσμα του γενετικού αλγορίθμου σε κατάλληλη μορφή για να θεωρηθεί ως είσοδος στην **QFModul.m**, με χρήση της **FindComponents.m** πάνω στο χρωμόσωμα που επιτυγχάνει τη βέλτιστη τιμή της συνάρτησης **fitness**.
- Θα έχει γίνει αυτό ούτως ή άλλως για τον υπολογισμό της συνάρτησης **fitness**.

Στην παρούσα άσκηση θα πρέπει να παραδοθεί και ο κώδικας.

SIR Model

- **SIR: Susceptible-Infected-Removed**
 - Also Kermack-McKendrick model
- Models specific & single threats
- 2 state transitions for nodes
 - S to I to R
- System of ODEs necessary
 - γ is the mean recovery (healing) rate

$$\begin{aligned}\frac{dS(t)}{dt} &= -\frac{\beta S(t)I(t)}{N}; \\ \frac{dI(t)}{dt} &= \frac{\beta S(t)I(t)}{N} - \gamma I(t); \\ \frac{dR(t)}{dt} &= \gamma I(t); \\ N &= S(t) + I(t) + R(t),\end{aligned}$$



- Nonlinear ODEs with no generic solution

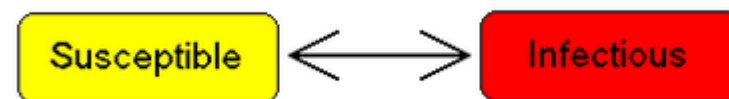
Epidemics: SIR Model

File: SIR.m

```
%-----  
% User Section 1: Definition of model parameters  
%-----  
%  
param.beta = 1e-3; % set the parameter 'beta' of the model  
param.r = 1e-1;    % set the parameter 'r' of the model  
  
%-----  
% User Section 2: Definition of initial conditions  
%-----  
%  
initial.S = 499;    % set the initial value of 'S'  
initial.I = 1;      % set the initial value of 'I'  
initial.R = 0;      % set the initial value of 'R'  
  
%-----  
% User Section 3: Definition of the simulation time  
%-----  
end_time = 100;  
  
%-----  
% User Section 4: Definition of the ODE system  
%-----  
function deriv = ode_system (t, x, param)  
% Function to calculate derivatives of the SIR model  
%  
% Input:  
%     t:      Time (not used in this example because there is no explicit  
%            time dependence).  
%     x:      Vector of the current values of all variables in the same  
%            order as you defined the initial values: S, I, R  
%     param:  Used to pass parameter values.  
% Output:  
%     deriv:  Column vector of derivatives, must be the same order as the  
%            input vector x.
```

SIS Model

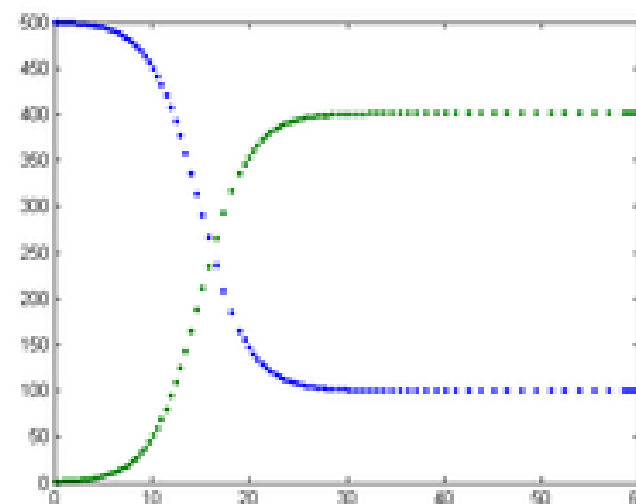
- SIS: Susceptible-Infected-Susceptible
- Examples are common cold & flu
- Macroscopic malware modeling
- No immunity after recovery



$$\frac{dS}{dt} = -\frac{\beta SI}{N} + \gamma I$$

$$\frac{dI}{dt} = \frac{\beta SI}{N} - \gamma I$$

$$\frac{dS}{dt} + \frac{dI}{dt} = 0 \Rightarrow S(t) + I(t) = N$$



- The dynamics of infectious is ruled by a logistic equation:

$$\frac{\beta N}{\gamma} \leq 1 \Rightarrow \lim_{t \rightarrow +\infty} I(t) = 0$$

$$I(t) = \frac{I_{\infty}}{1 + V e^{-\chi(t-t_0)}} \quad I_{\infty} = \chi N / \beta \quad \chi = \beta - \gamma$$

$$\frac{\beta N}{\gamma} > 1 \Rightarrow \lim_{t \rightarrow +\infty} I(t) = \frac{\beta N - \gamma}{\beta}$$

$$V = I_{\infty} / I_0 - 1$$

Epidemics: SIR Model

File: SIS.m, func_SIS.m, beta.m

```
[T,Y] = ode45('func_SIS',[0 10],[0.2 0],options);
```

Based on the following set of equations

$$\begin{aligned}\frac{dI}{dt} &= (\beta(t)N - a)I - \beta(t)I^2 \\ \beta(t) &= 2 - 1.8\cos(5t)\end{aligned}$$

Modified (periodic) contact rate – e.g. flu/common cold