

The Board Game

Simulatore epidemiologico in C++

Michele Maestrelli, Nicolò Massari

1 Descrizione del progetto

Lo scopo della simulazione è quello di mostrare agli utenti del programma l'evoluzione nel tempo di un contagio epidemiologico basato sulle ipotesi del modello "SIR" (Suspected Infected Recovery). Il modello cerca di fornire una previsione teorica del contagio a partire dalla conoscenza di parametri noti quali l'indice (o probabilità) di contagio, indicato con B nel codice sorgente, e quello di guarigione, indicato con G . A seconda delle possibili combinazioni di B e G , sarà diversa l'entità dell'epidemia.

Per la simulazione è stata implementata una classe *Board*, contenente una griglia ($n \times n$, nelle opzioni non personalizzabili è 150×150) composta da celle di 4 diversi tipi:

- **State::S** = *Susceptible*, persone sane potenzialmente infettabili
- **State::I** = *Infectious*, persone infette
- **State::R** = *Recovered*, persone guarite o morte
- **State::B** = *Block*, ostacoli naturali o artificiali

I parametri che sono stati implementati nel programma sono:

- **B** = 0.40, la probabilità di infezione
- **G** = 0.25, la probabilità di guarigione
- **Range** = 1, il raggio d'azione del contagio
- **f** = 0.8, la frazione di celle che si possono muovere nella board
- **Block_parameters** = struttura di parametri che definisce la distribuzione della densità dei blocchi nella board

In seguito all'esecuzione del programma (si veda la SEZ. 2 per le istruzioni), l'utente è accolto in *The Board Game*, da un menu interattivo dal quale è invitato a scegliere una delle cinque modalità disponibili per l'evoluzione del contagio. Le prime tre sono esempi con parametri fissi che differiscono per i soli *Block_parameters*:

NOTA: Negli esempi successivi le celle sono così suddivise: blocchi (nere), suscettibili (gialle), contagiati (rosse), guariti/morti (arancioni).

"**r**" for *realistic*, così chiamato a causa dell'andamento omologo a quello del modello teorico SIR, ciò è dovuto alla logica con la quale sono stati scelti i parametri per la generazione dei blocchi in questo modello. Infatti questo modello è caratterizzato da un omogeneità nella distribuzione dei blocchi.

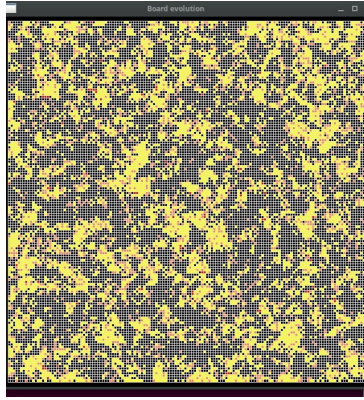


Figure 1: Output grafico dell'opzione "**r**"

"**u**" for *uncivilised* dove la differenza risiede nei parametri di generazione dei blocchi. I parametri che sono stati scelti rendono la generazione dei blocchi molto meno uniforme e quindi concentrata in punti ad alta densità. E' importante riflettere sul fatto che ogni parametro sia identico alla modalità realistica e l'unica differenza risiede nella distribuzione dei blocchi.

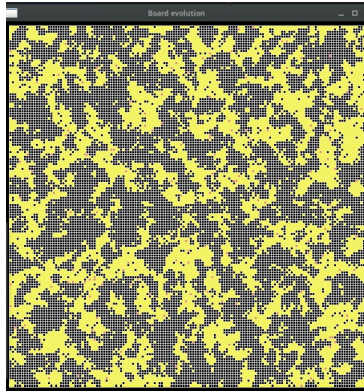


Figure 2: Output grafico dell'opzione "**u**"

”**n**” for *non-realistic*, mostra come il la pandemia si evolverebbe se non ci fossero blocchi o separazioni di alcun tipo, ovvero come se tutte le persone fossero a stretto contatto.

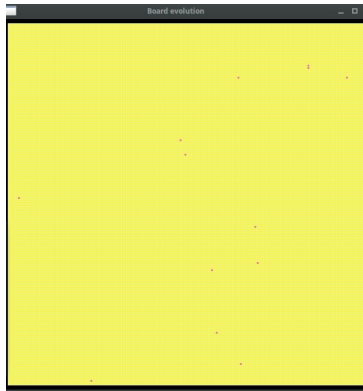


Figure 3: Output grafico dell'opzione ”n”

“**p**” for *personalised*, permette all’utente di scegliere i principali parametri, senza entrare nei dettagli dei parametri legati alla generazione dei blocchi (neri).

```
Welcome in The Board Game!
In order to continue please select one of the following options:
--> r <-- r for realistic, our best model!
--> u <-- u for uncivilised, fewer blocks and lower moving fraction.
--> n <-- n for nonrealistic, zero or close to zero blocks.
--> p <-- p for personalised, choose your own parameters!
--> pa <-- pa for personalised advanced, play with how the blocks are distributed! (we strongly recommend to read instructions before choosing this option).
Please press one the following keys r, u, n, p, pa...
p

Please, insert the probability of infection of your virus (from 0 to 1):
0.7

Please, insert the probability of healing from your virus (from 0 to 1):
0.65

Please, insert the fraction of moving people in your model (from 0 to 1):
0.9

Please, insert the range of infection of your virus (from 0 to (n - 1) / 2):
7

Please, insert the number of initial infected in your model (integer from 0 to n * n * 0.1):
3

Please, insert the percentage of blocks in your model (integer from 0 to 9):
4
```

Figure 4: Opzione ”p” sul terminal

“**pa**” for *personalised advanced*, permette di scegliere tutti i parametri, inclusa la dimensione del lato della board (n, negli esempi già fatti è 150) e tutte le 8 variabili che compongono il Block.parameters.

```

Welcome in the Board Game!
In order to continue please select one of the following options:
--> r <-- r for realistic, our best model!
--> u <-- u for uncivilised, fewer blocks and lower moving fraction.
--> n <-- n for nonrealistic, zero or close to zero blocks.
--> p <-- p for personalised, choose your own parameters!
--> pa <-- pa for personalised advanced, play with how the blocks are distributed! (we strongly recommend to read instructions before choosing this option).
Please press one the following keys r, u, n, p, pa...
pa

Please, insert the dimension (the length) of your board (from 1 to 200):
175

Please, insert the probability of infection of your virus (from 0 to 1):
0.8

Please, insert the probability of healing from your virus (from 0 to 1):
0.3

Please, insert the fraction of moving people in your model (from 0 to 1):
0.7

Please, insert the range of infection of your virus (from 0 to (n - 1) / 2):
3

Please, insert the number of initial infected in your model (integer from 0 to n * n * 0.1):
2

Please, insert the percentage of blocks in your model (integer from 0 to 9):
7

Please, insert the first number to define the interval for the generation of blocks in your model (integer from 0 to 8):
parameter_1 = 1

Please, insert the second number to define the interval for the generation of blocks in your model (integer from parameter_1 to 8):
parameter_2 = 3

Please, insert the third number to define the interval for the generation of blocks in your model (integer from parameter_2 to 8):
parameter_3 = 5

Please, insert the probability of generating a block between 0 and parameter_1 (decimal from 0.01 to 1):
0.2

Please, insert the probability of generating a block between parameter_1 and parameter_2 (decimal from 0.01 to 1):

```

Figure 5: Opzione "pa" sul terminal

Al programma è stata data inoltre un'impronta di carattere randomico. Infatti ogni simulazione sarà diversa da quella successiva, oltre che per la randomicità dell'infezione anche per la casuale distribuzione dei blocchi, che ad ogni esecuzione vengono generati.

Durante la simulazione del contagio, oltre alle finestre grafiche l'utente visualizza un ulteriore menu in cui è presente una lista di comandi eseguibili da tastiera per modificare lo sviluppo del contagio in corso. Questi comandi, che si attivano premendo i tasti corrispondenti, sono:

- **Q** (*Quarantine*), introduce la quarantena per la popolazione. La probabilità di essere infettati dal virus non cambia, ciò che cambia è la frazione f delle persone che si muovono nella griglia. Infatti si può notare dal menu che la frazione passa dal valore iniziale di 0.8 ad un valore di 0.1.
- **L** (*Less*), modifica il *delay* (l'intervallo di tempo tra un'iterazione e la successiva), minimizzandolo, ovvero lasciando che il tempo di iterazione sia solo dipendente dalla velocità di esecuzione del programma.

In ogni caso, l'evoluzione del contagio prosegue fino a quando l'utente non sceglie di chiudere manualmente la simulazione, interrompendo quindi l'esecuzione del programma. Questa scelta è stata fatta per dare all'utente il tempo necessario di osservare e leggere i grafici, senza chiusure improvvise del programma.

In generale il programma crea un modello di simulazione pandemica in funzione

di variabili dipendenti dal virus (B, G, range), di variabili dipendenti dalla popolazione (f) e della conformazione geografica del luogo (ottenibile con uno studio opportuno delle variabili di Block_parameters). Con l'idea di distribuire un modello di simulazione pandemica semplice in formato di gioco ed esperienza ricreativa.

Inoltre, la presenza dei due grafici dinamici, uno teorico e l'altro relativo all'effettiva simulazione lanciata, permette di stabilire in modo diretto e visuale il confronto tra un modello basato solamente sui parametri di contagio e guarigione (cioè il modello teorico SIR) e il modello programmato che tiene conto di molti altri fattori e approssima al suo meglio un contagio reale (cioè la simulazione nella Board).

2 Istruzioni per l'esecuzione (Linux users)

Eeguire le seguenti linee di comandi:

(Cloning della *repository git*)

```
$ git clone https://gitlab.com/zLocoo/board_game.git
```

(Esecuzione del programma)

```
$ cd board_game/board_game_code
```

```
$ rm -rf build
```

```
$ mkdir build
```

```
$ cd build
```

```
$ cmake ../.
```

```
$ make $ ./board_game_main
```

(Esecuzione del *testing*)

```
$ cd board_game/board_game_code
```

```
$ rm -rf build
```

```
$ mkdir build
```

```
$ cd build
```

```
$ cmake ../.
```

```
$ make $ ./testing
```

NOTA: le linee di comandi sono precedute dal \$, il dollaro non è compreso nel comando da eseguire.

3 Descrizione dell'output

Output da terminale: il programma interagisce con l'utente dando il benvenuto e chiedendo la modalità d'uso, permettendo di scegliere tra tre diverse opzioni.

Output grafico: subito dopo la selezione della modalità, il programma genera tre finestre. Una più grande che mostra la diffusione del contagio, tramite celle, della popolazione e due più piccole contenenti gli andamenti del contagio.

Le celle della board sono così suddivise:

- Celle nere: rappresentano i blocchi (descritti nella SEZ. 1)
- Celle gialle: persone suscettibili al contagio (parametro S)
- Celle arancioni: persone contagiate (parametro I)
- Celle rosse: persone immunizzate (guarite o decedute; parametro R)

Per il grafico del modello SIR (teorico):

- Curva verde: suscettibili (S)
- Curva rossa: contagiati (I)
- Curva blu: immunizzati (guariti o deceduti; R)

Per il grafico rappresentante la board:

- Curva arancione: suscettibili (S)
- Curva rossa: contagiati (I)
- Curva gialla: immunizzati (R)

Le finestre sono state programmate con l'utilizzo della libreria SFML. Le finestre si possono muovere, chiudere e minimizzare.

Inoltre, sono stampati a schermo tutti i dati relativi alla simulazione in corso.

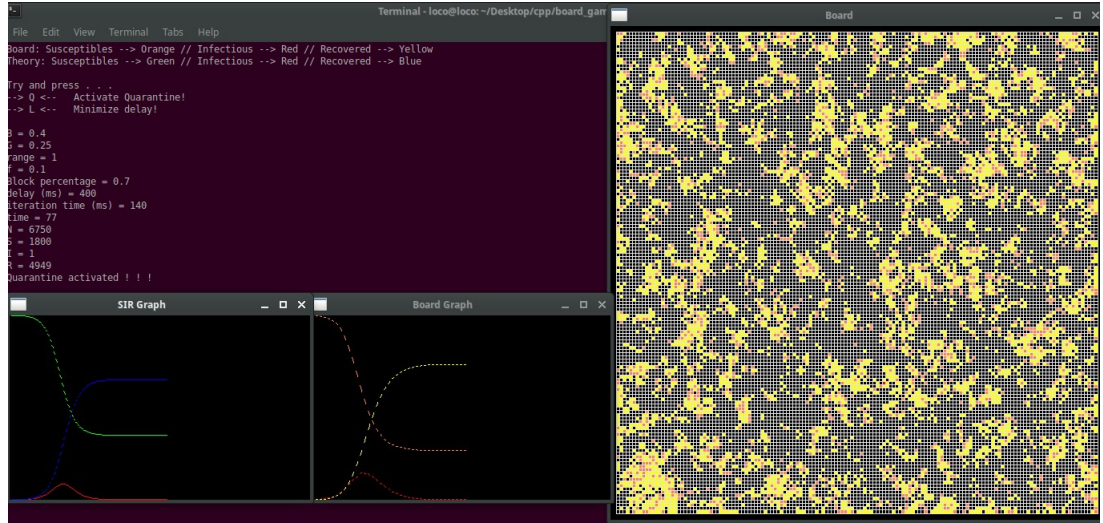


Figure 6: Esempio di output completo

4 Interpretazione dei risultati

Nella (quasi) totalità dei casi di un contagio “realistico”, osservando i due grafici, si può notare come la simulazione da noi sviluppata approssimi ragionevolmente l’andamento delle curve predetto dal SIR.

Invece, nei casi “u” e “n” si nota appunto come ciò che accade nella board non rispecchia il modello. Questo perché i blocchi (neri), che rappresentano le barriere fisiche, sono generati molto meno uniformemente (caso “u”) o addirittura sono completamente inesistenti (caso “n”).

Per chiarezza si osservino anche i confronti nelle figure a seguire.

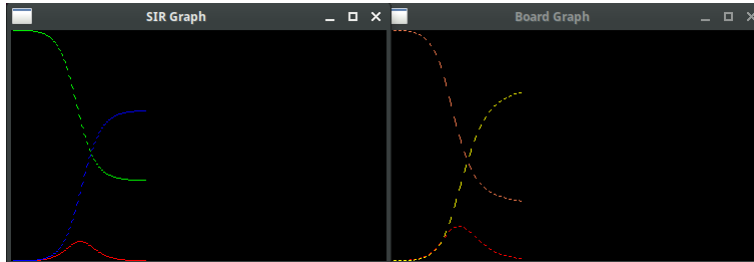


Figure 7: Confronto tra SIR e simulazione nel caso *realistic*

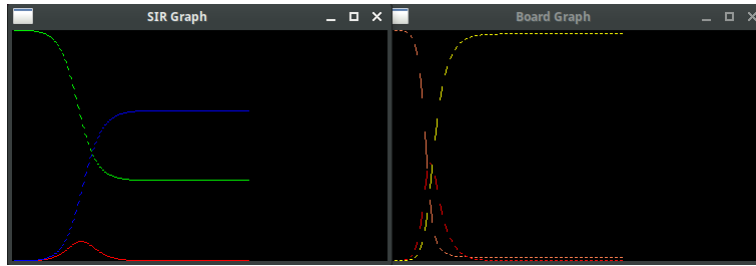


Figure 8: Confronto tra SIR e simulazione nel caso *uncivilized*

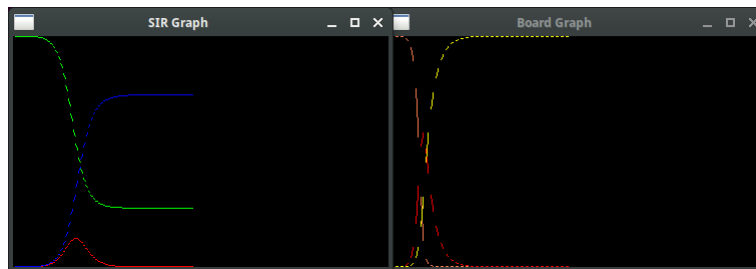


Figure 9: Confronto tra SIR e simulazione nel caso *non-realistic*

5 Strategie di test

Il testing del programma è focalizzato su due parti principali:

- Processo di infezione della Board
- Calcolo del modello SIR

Per il processo di infezione della board sono state testate tutte le funzioni utilizzate per arrivare all'evoluzione, in più sono stati aggiunti asserts e exceptions per assicurarsi che il programma non possa fare cose assurde e infattibili.

Per quanto riguarda il calcolo invece è stato effettuato un testing della correttezza del risultato ottenuto, dal momento che il modello SIR è basato su formule e quindi di carattere puramente matematico.