

UNIVERSITA' DEGLI STUDI DI BOLOGNA

Facoltà di scienze matematiche fisiche e naturali

CORSO DI LAUREA IN SCIENZE DI INTERNET

Relazione per il progetto di Sistemi Adattivi Complessi

AUTORE

SUSI STEFANO

DOCENTE

**OZALP BABAUGLU
STEFANO ARTECONI**

Capitolo 1 - Introduzione

SLACER è un protocollo che implementa il gioco del dilemma del prigioniero. Ogni individuo può decidere se attuare un comportamento egoistico o meno, ricevendo una determinata “utilità” in funzione del comportamento dell’avversario. Tale gioco porta a considerazioni paradossali se svolto per una sola volta, nel nostro caso però attraverso il simulatore PEERSIM riusciamo ad eseguire la simulazione per N cicli, quindi il nostro lavoro si è concentrato nel tentare di far “comprendere” ai nodi di attuare un comportamento tale da massimizzare l’utilità collettiva.

Il presente lavoro si è concentrato nell’implementare i metodi DROP end REWIRE dell’algoritmo SLACER . Tale algoritmo viene eseguito ad ogni passo della simulazione ed i metodi implementati sono eseguiti esclusivamente dal nodo che perde il confronto di utilità con un suo vicino scelto casualmente, tramite NEWSCAST.

Per quanto riguarda le funzionalità di distacco dei nodi della vista (ovvero il DROP) mi è sembrato sensato far staccare il nodo perdente da tutti quei nodi che hanno la stessa strategia o un utilità minore .

Logicamente un nodo non riesce ad imparare nulla da un suo simile che si comporta come lui (che è già risultato perdente) o da uno la cui utilità risulti inferiore.

Inoltre ho voluto inserire un minimo di “selezione“, ovvero ho escluso tutti i nodi che non hanno effettuato molte giocate, nel caso specifico un numero inferiore a 3, questo perché un nodo potrebbe risultare vincente una volta, solo per situazioni particolari che si sono verificate, di conseguenza la probabilità di una vittoria “fortunata” risulta molto più difficile su un numero maggiore di giocate .

Riporto un frammento di codice che descrive il comportamento precedentemente descritto :

```
if ((ySlacer.strategy == mySlacer.strategy) || (ySlacer.games < 3) || mySlacer.payoff < ySlacer.payoff){  
    mySlacer.removeNeighbor(y);  
    ySlacer.removeNeighbor(me);  
}
```

Ho fatto analoghe considerazioni per il metodo REWIRE, che permette di scegliere i NODI su cui agganciarsi una volta perso il gioco, logicamente l’implementazione di tale metodo è opposta al DROP, in quanto il nodo perdente preferirà agganciarsi con i nodi che attuano una strategia diversa dalla sua (che è appena stata sconfitta) .

Riporto successivamente un frammento di codice del metodo :

```
if(mySlacer.strategy != nSlacer.strategy && mySlacer.payoff < otherSlacer.payoff){  
    mySlacer.addNeighbor(n);  
    otherSlacer.addNeighbor(me);  
  
} else if (CommonState.r.nextDouble() < casualita){  
    mySlacer.addNeighbor(n);  
    otherSlacer.addNeighbor(me);  
}
```

Ho introdotto una variabile double “casualità”, che l’utente può comodamente impostare dal file di configurazione attraverso l’impostazione di un valore; tale parametro che nei test è impostato a 0,1(10%) rappresenta la percentuale di un possibile aggancio con un nodo che stia utilizzando la stessa strategia del nodo perdente oppure ha un’utilità che risulta minore rispetto al nodo perdente.

Capitolo 2 – I Parametri di Configurazione

Per effettuare i test dell'algoritmo ho utilizzato diverse configurazioni possibili.

Ho impostato le dimensioni della rete su tre possibili valori di network size :1000, 5000, 10000, inoltre ho fatto variare il valore di cache, ovvero il valore del numero massimo di vicini che ogni nodo può conoscere e quindi selezionare per effettuare il “gioco”. I valori del parametro di cache sono stati impostati su 20 e 40.

La cache è anche conosciuta come vista cioè è una collezione di vicini a cui un nodo è logicamente collegato.

Per quanto riguarda l'overlay, ovvero la topologia di rete di partenza, ho scelto due possibili scenari alternativi WireKOut, WireStar.

Riporto i parametri passati nel file di configurazione:

```
protocol.1.caso 0.1  
protocol.1.reprprob 0.2  
protocol.1.mutation 0.005  
protocol.1.tagmutation 0.01  
init.2.quantity 0.3
```

Il primo parametro (“caso”), è un valore inserito nel mio algoritmo ed indica la percentuale con cui un nodo può agganciarsi ad un altro (anch'egli perdente) in fase di REWIRE.

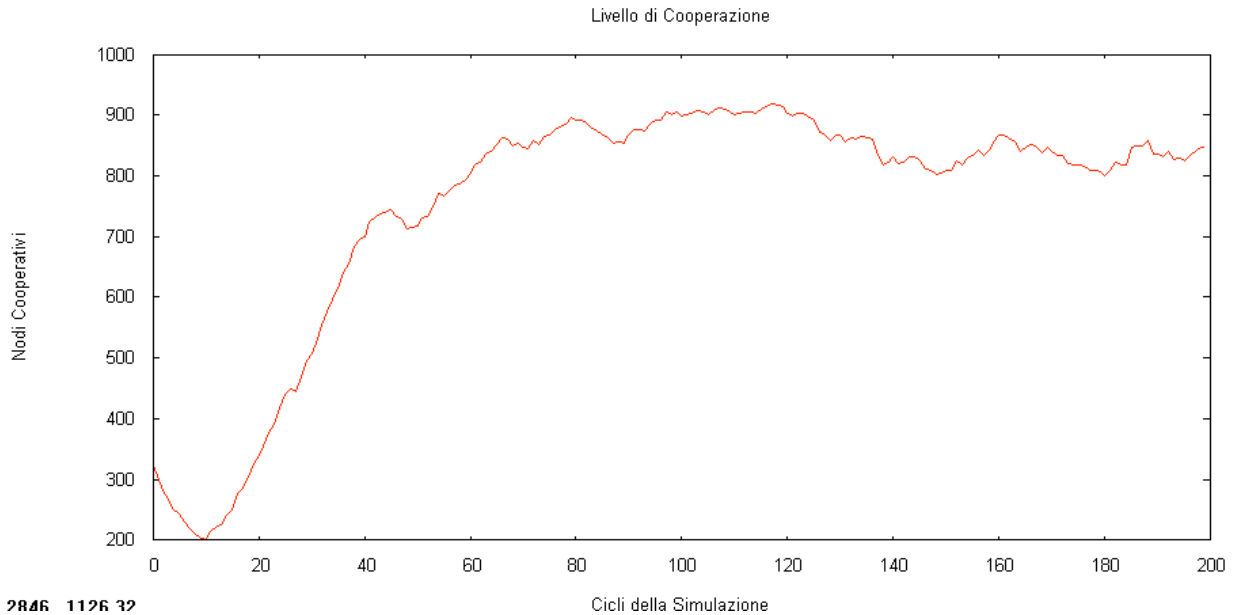
Tutti gli altri parametri protocol sono tipici dell'algoritmo SLACER.

L'ultimo invece rappresenta la percentuale di nodi Cooperativi iniziale.

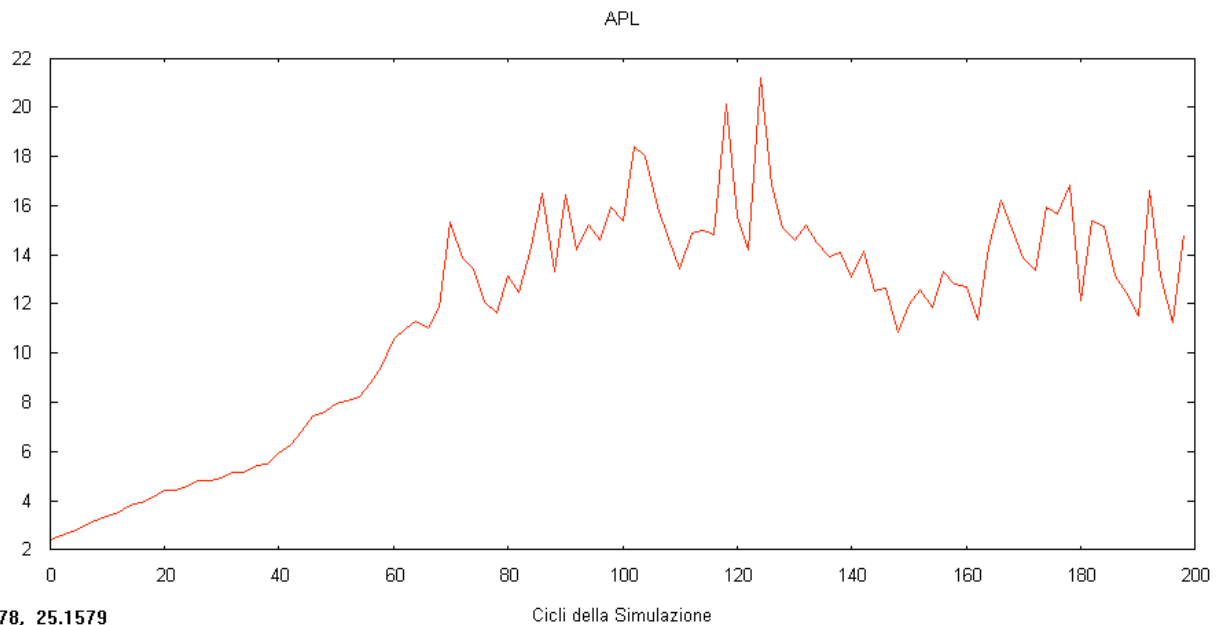
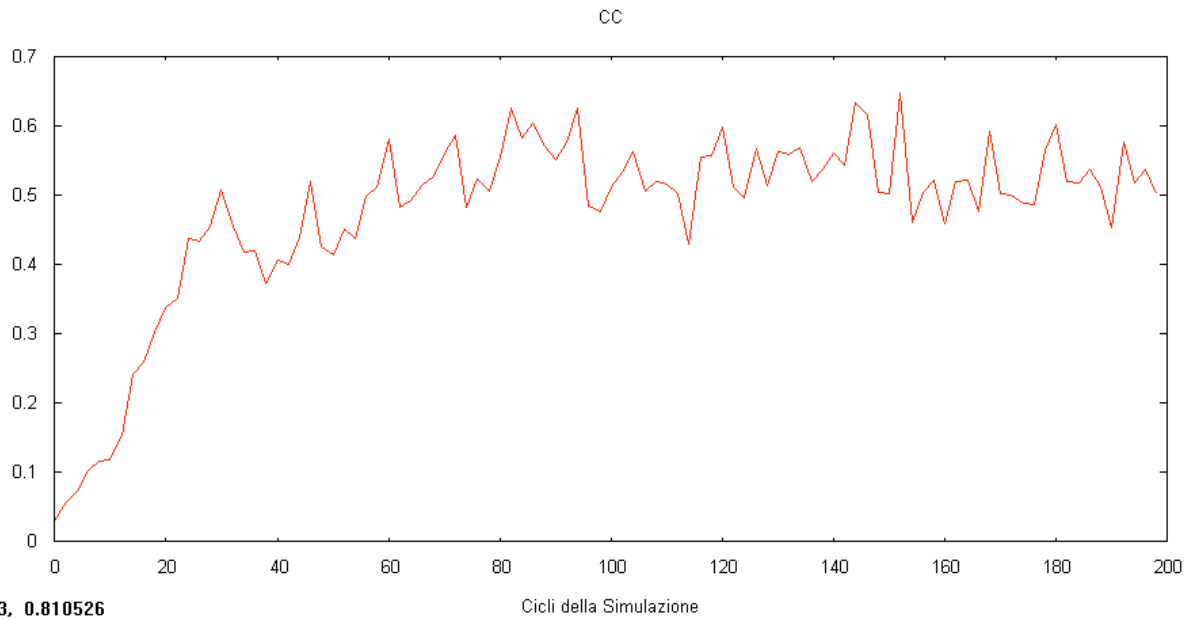
Dopo diversi tentativi empirici ho deciso di mantenere fissi tali parametri, successivamente dimostrerò che il cambiamento di alcuni di essi non influenza in maniera netta l'evoluzione delle rete.

Capitolo 3 – Risultati delle Simulazioni

Risultati Network-size 1000, Cache 20, Overlay WireKOut.

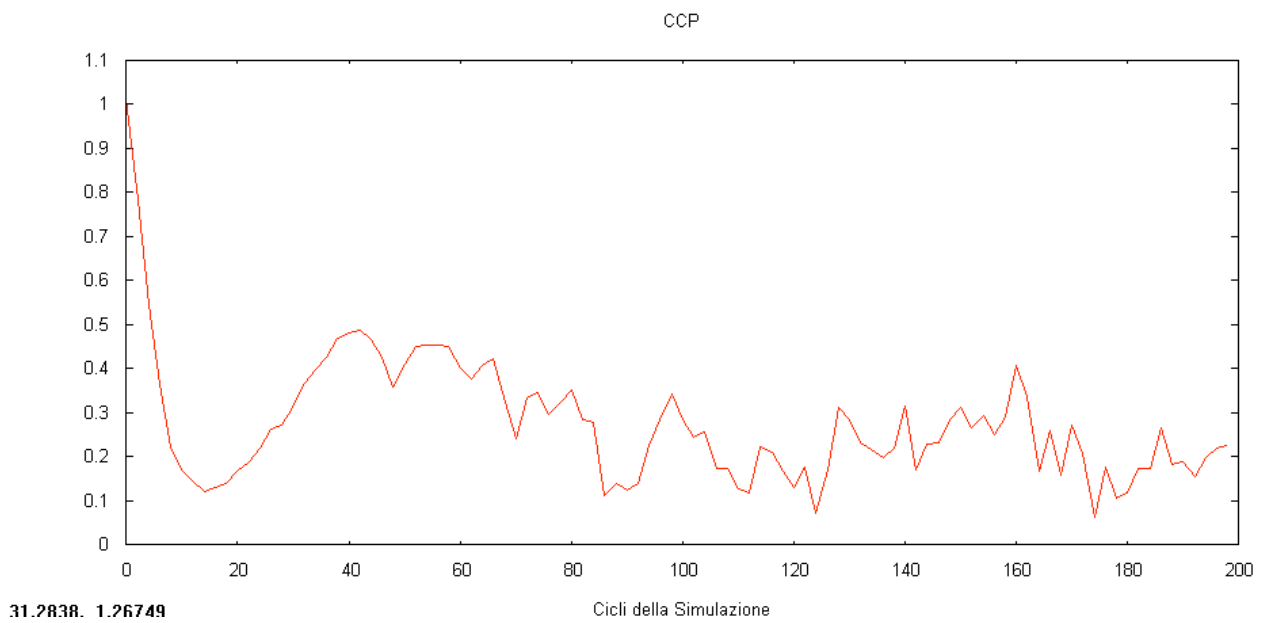
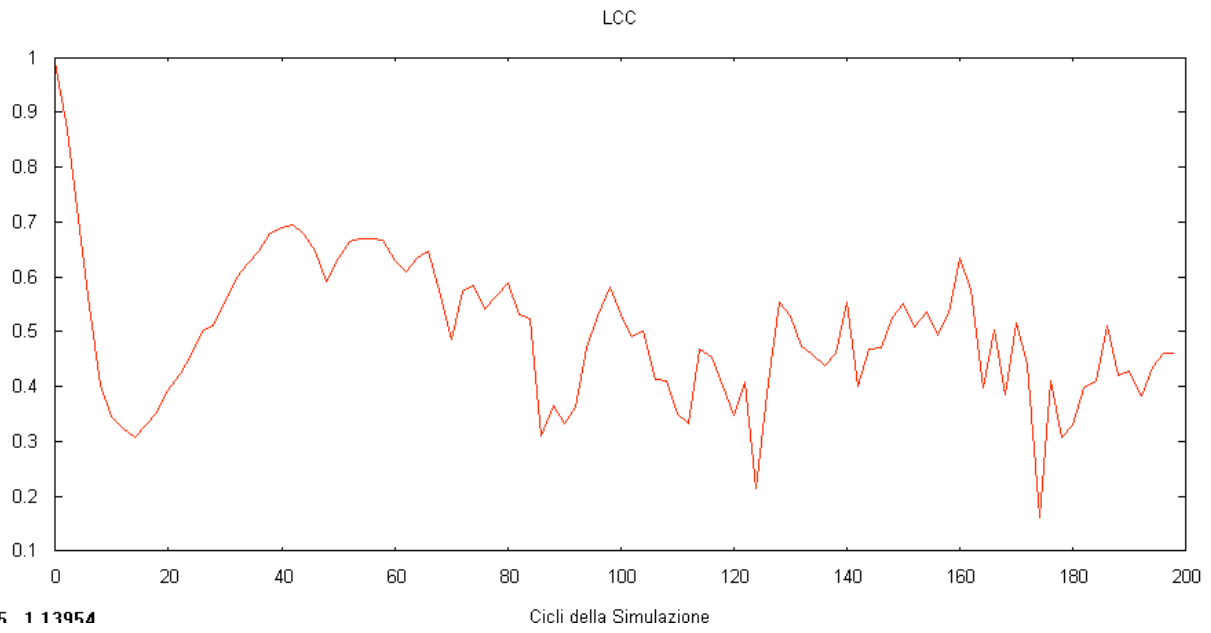


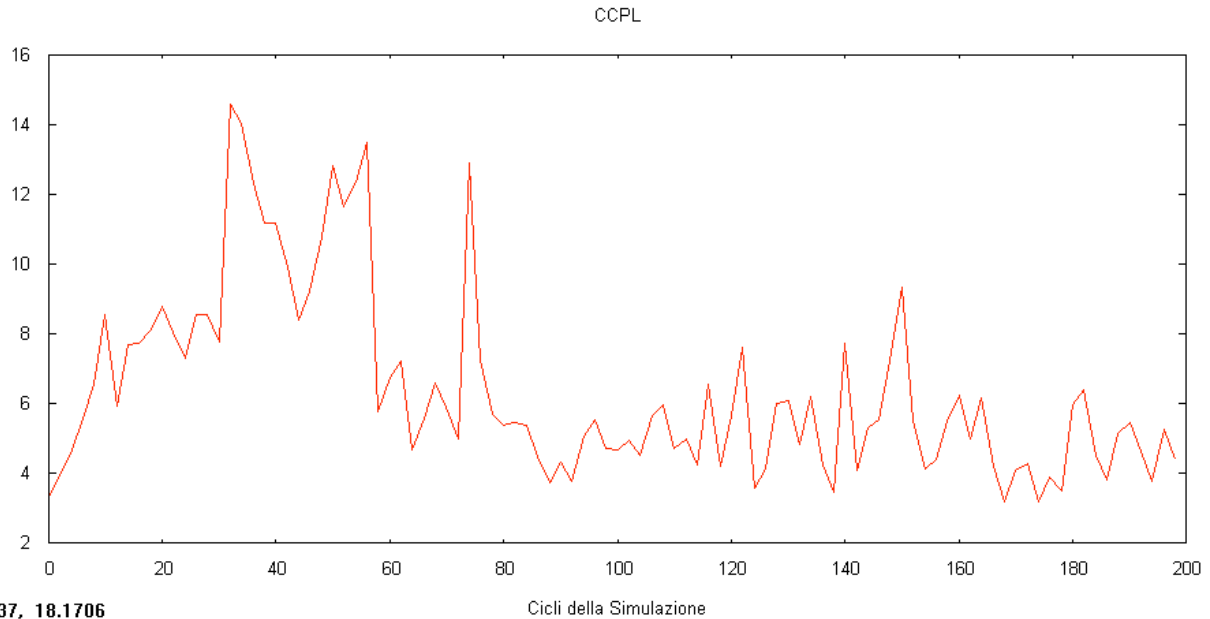
La figura illustra il livello di cooperazione: è interessante notare l'andamento dei nodi cooperativi, che partono come stabilito da un livello del 30 % , attuano un comportamento egoistico (non cooperativo) nei primi cicli della simulazione, per poi “capire” che quello attuato non sia il comportamento che ottimizza la loro utilità . Si arriva ad un livello di alta cooperazione dopo 40-50 cicli , il valore finale oscilla tra il 90 % e l'85 % . Logicamente non si giungerà mai ad una completa stabilità perché i nodi muteranno la loro strategia in funzione dei parametri M e TM del file di configurazione



Le due precedenti immagini illustrano il CC e l' APL che ci forniscono una stima sulle proprietà topologiche della rete. In questa simulazione il CC si attesta su valori pari al 0.5; il CC è il rapporto tra i collegamenti esistenti tra un nodo ed i suoi vicini e tutti i possibili collegamenti .

L'APL, che ci indica una stima della lunghezza media di tutti i cammini possibili, si attesta su valori molto alti. Logicamente sarebbe preferibile avere un valore di APL basso (4-5), in questo caso il valore a fine simulazione si aggira su 14 il che indica che molti nodi sono connessi tra loro attraverso cammini molto lunghi.

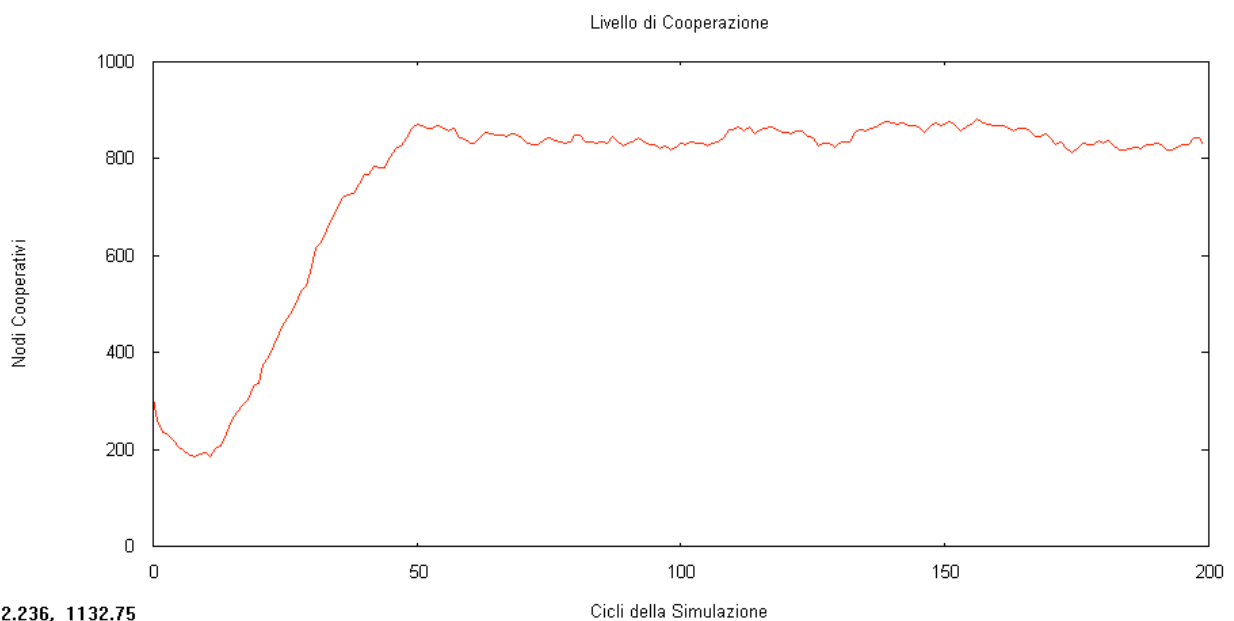




Risulta interessante confrontare il CCPL con l'APL, a fronte di un valore elevato di APL abbiamo un valore molto basso di CCPL (4), questo porta a pensare che i nodi cooperativi siano collocati in zone molto vicine tra loro. Il valore di LCC pari al 0.5 indica che vi sia una grande zona all'interno della rete formata da nodi cooperativi.

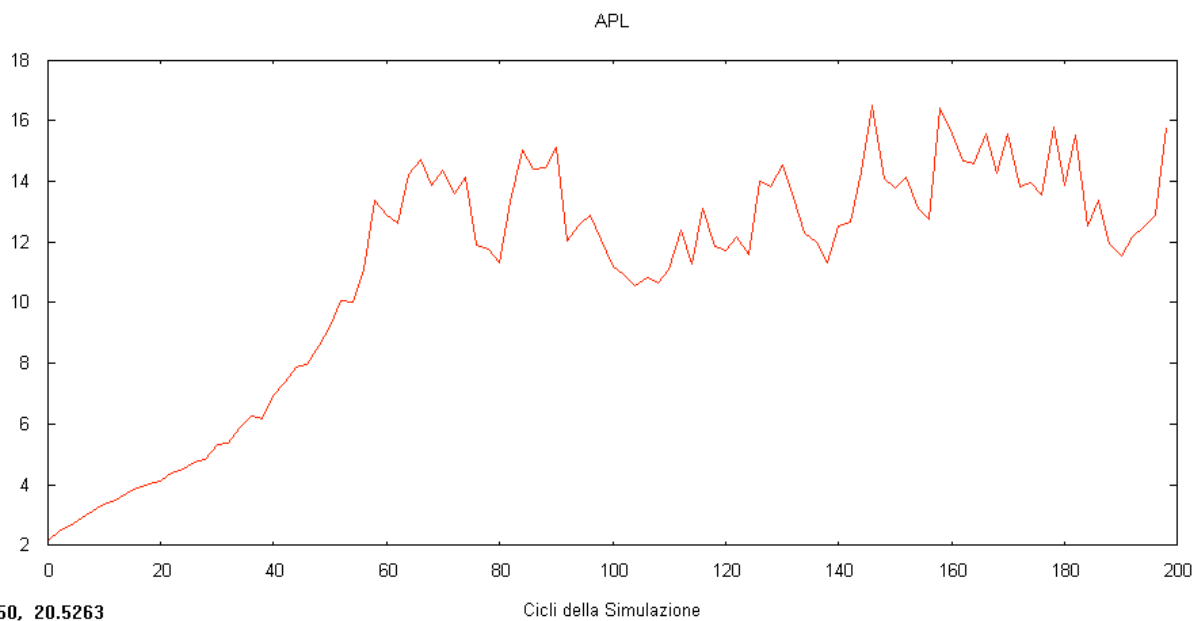
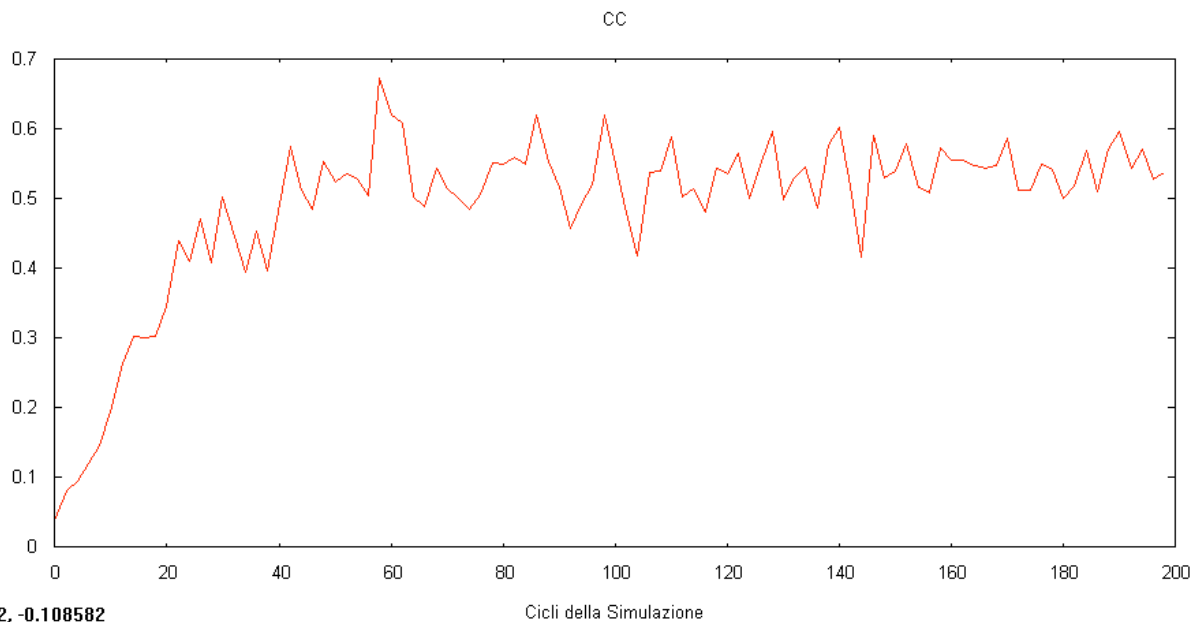
Un basso valore di CCP, nonostante la rete evolva verso uno stato di alta cooperazione, potrebbe significare che siano presenti dei nodi non cooperativi all'interno delle rete che bloccano il percorso cooperativo.

Risultati Network-size 1000, Cache 40, Overlay WireKOut.

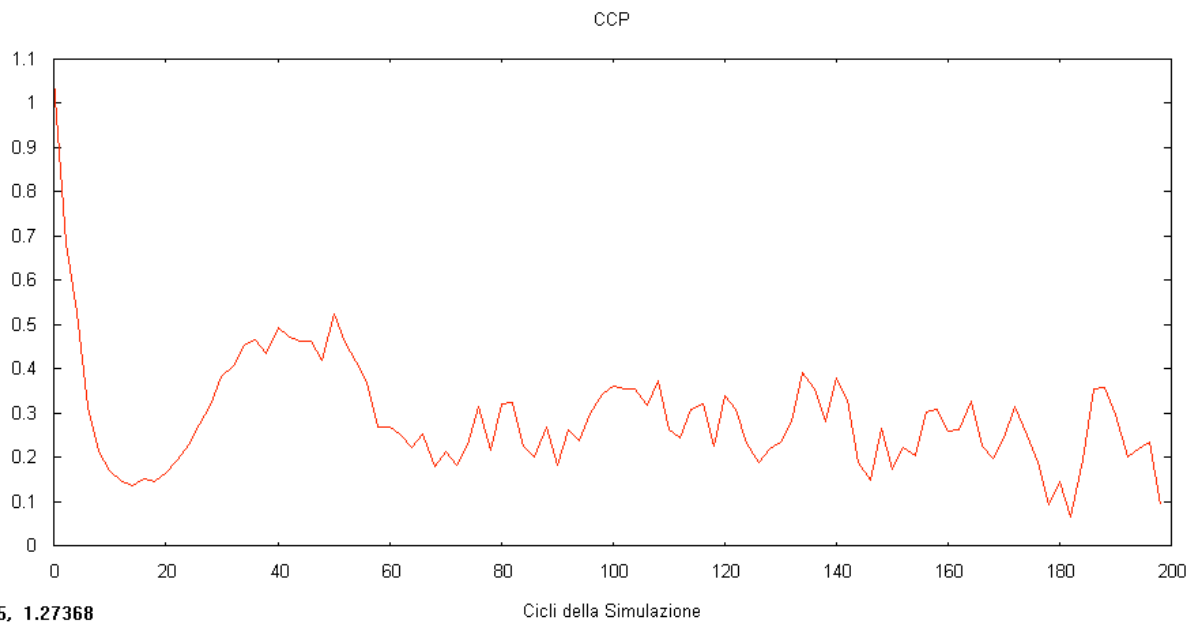
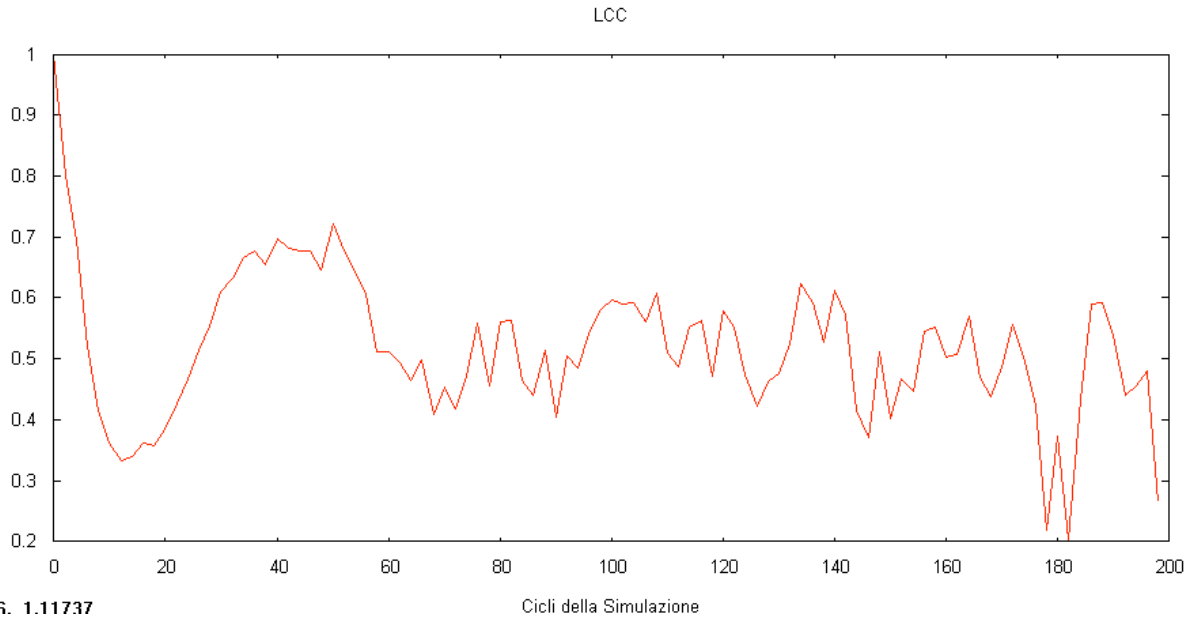


In questa simulazione ho scelto di lasciare inalterati tutti i parametri tranne il valore della cache, ovvero il valore massimo della vista che ogni nodo può assumere. Tale parametro è passato da 20 a 40.

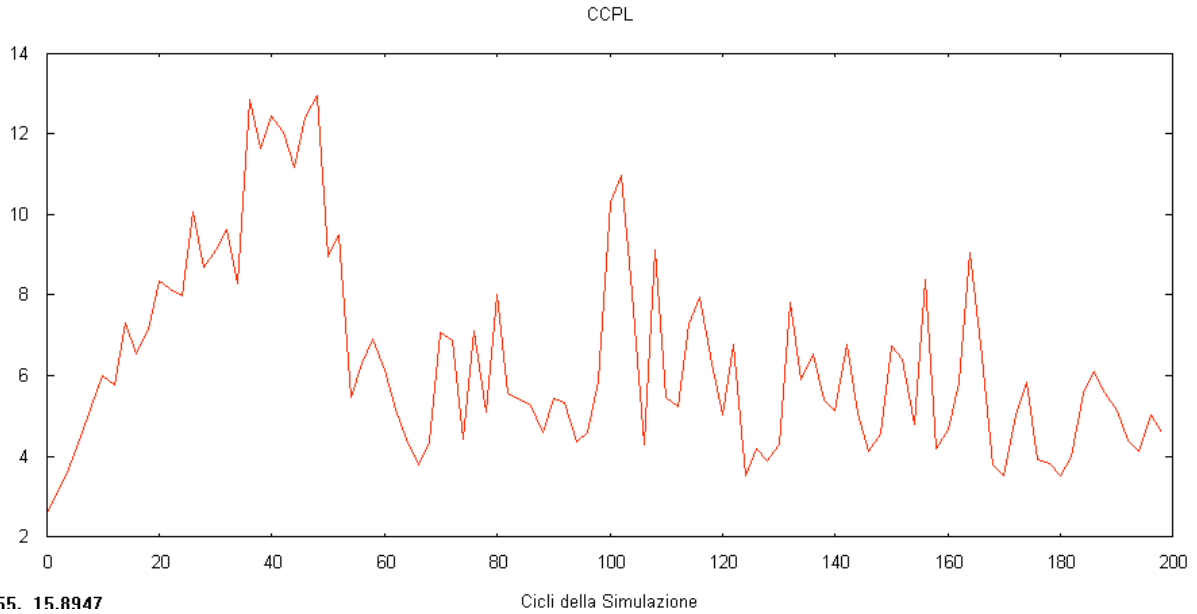
Come era facilmente intuibile si è raggiunto un livello di cooperazione tendenzialmente più basso, anche se la rete evolve verso uno stato di alta cooperazione in maniera più rapida, riuscendo ad attuare subito la strategia maggiormente redditizia .



Per quanto riguarda le proprietà topologiche della rete CC ed APL, questi parametri raggiungono valori simili, per cui le conclusioni che si possono trarre sono simili a quelle fatte nella precedente simulazione.



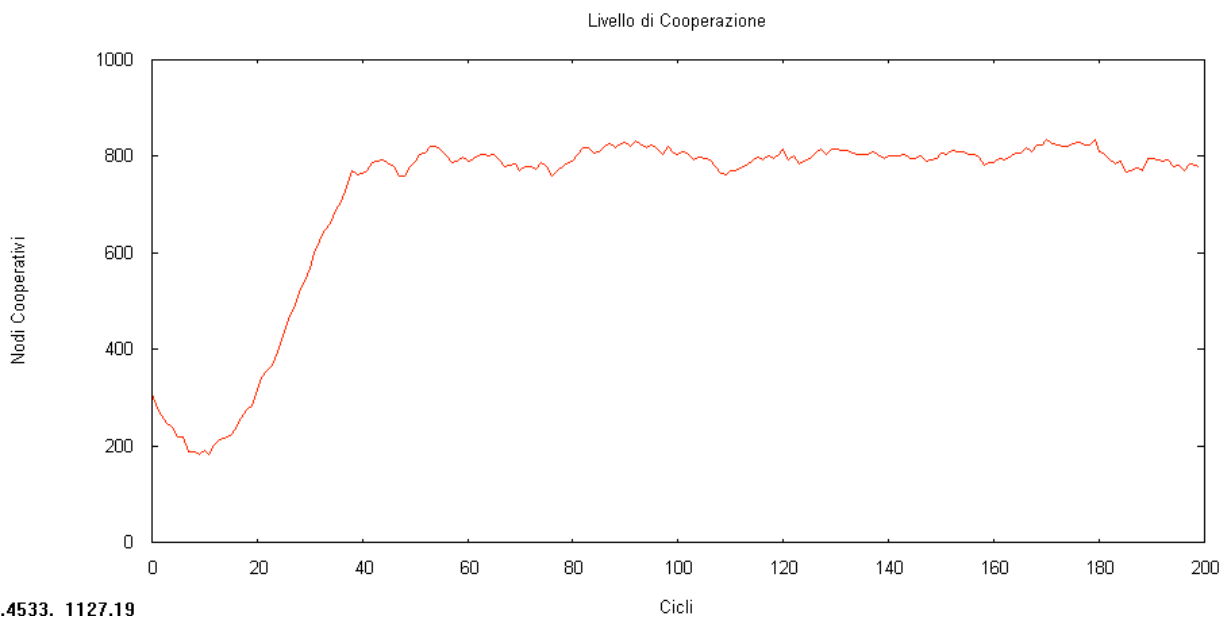
In questo caso abbiamo valori molto bassi di LCC e CCP (come in precedenza): sarebbe preferibile avere un comportamento migliore .



Per quanto riguarda il CCPL valgono le considerazioni fatte nella simulazione precedente .

Risultati Network-size 1000,Cache 40,Overlay WireKOut con M e TM x2

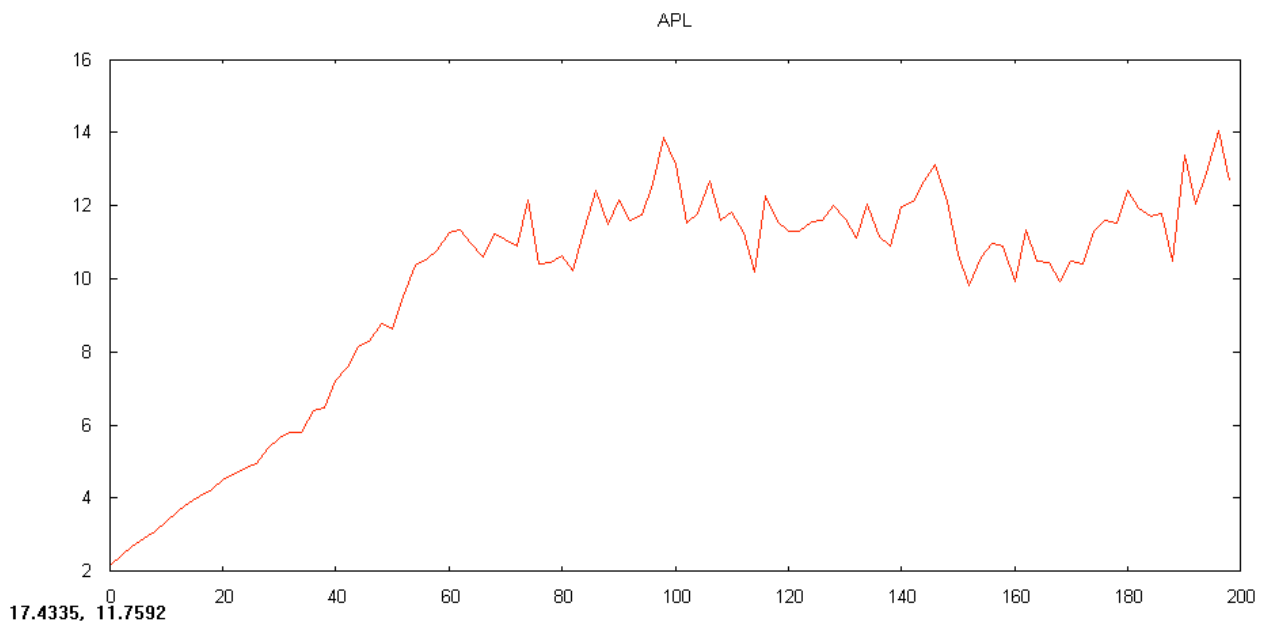
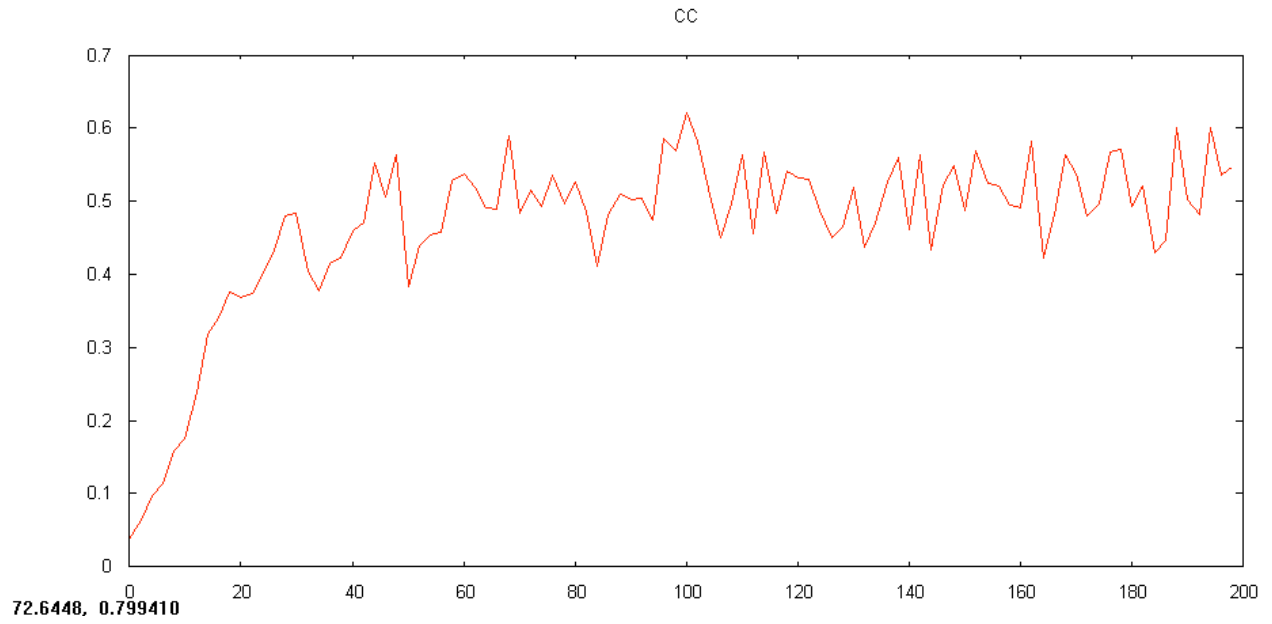
In questo scenario ho raddoppiato i valori di M e TM ,

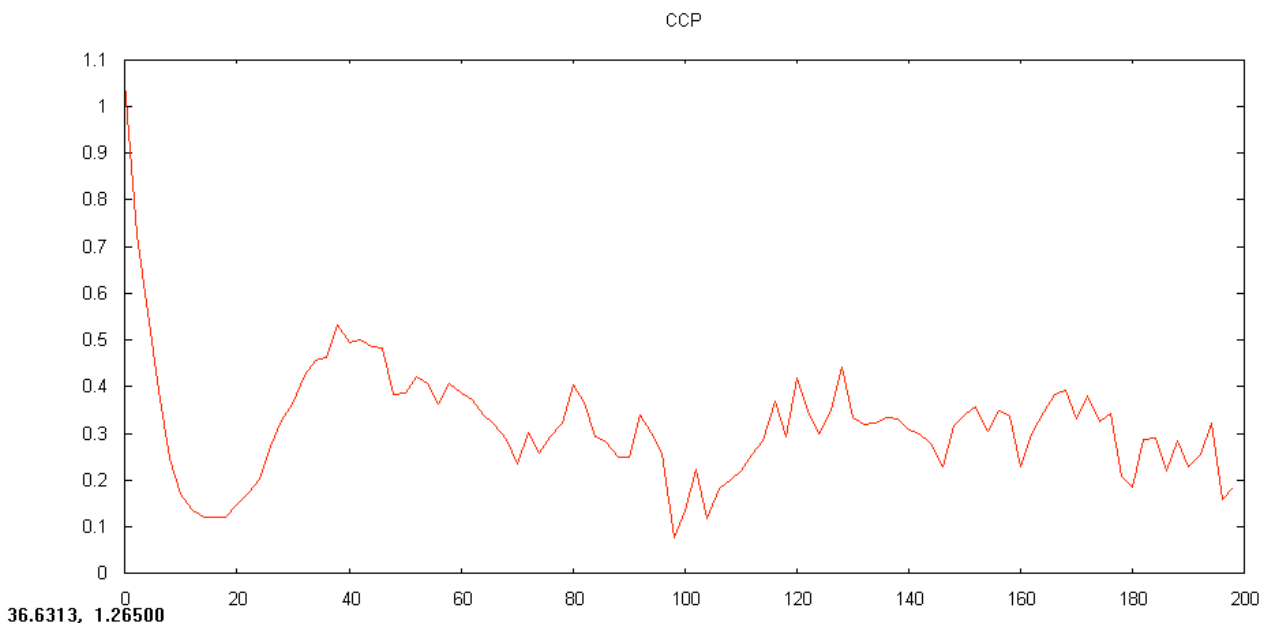
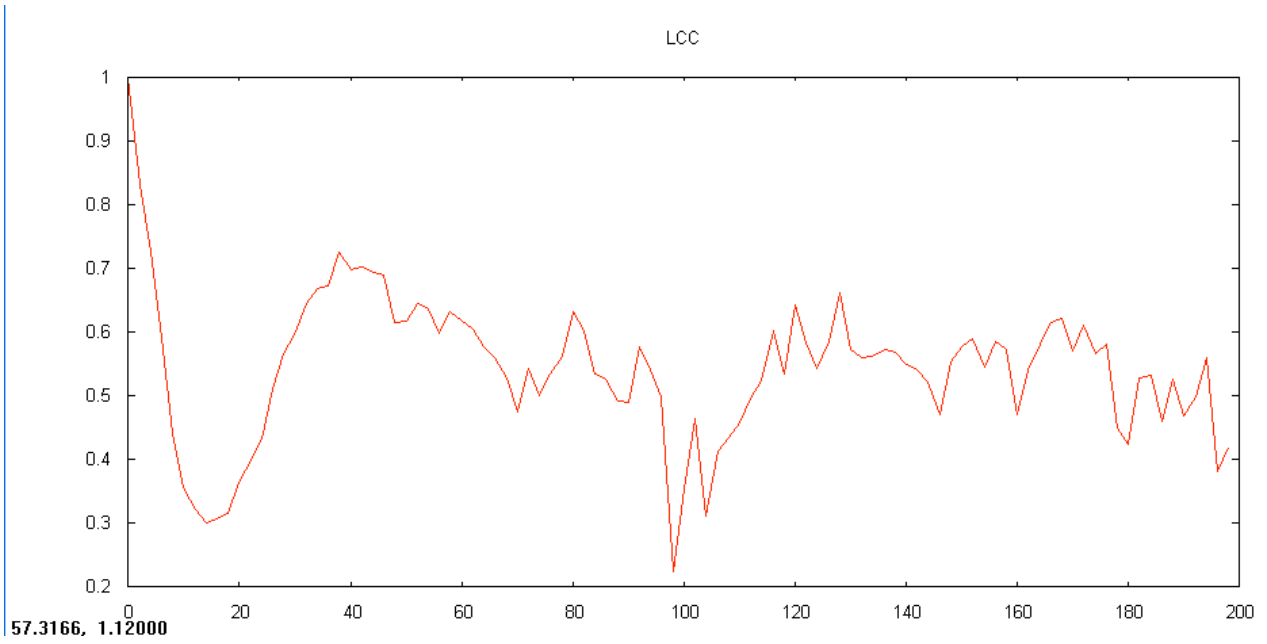


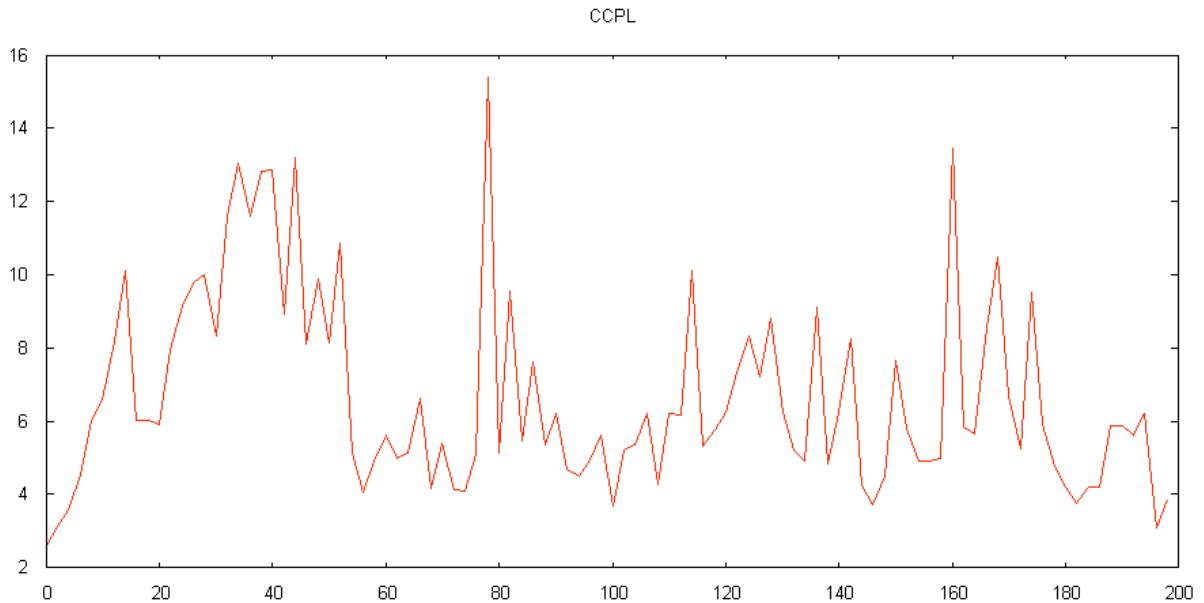
Possiamo notare come valori maggiori di M e TM abbassano di qualche punto percentuale il livello di cooperazione finale che si attesta di poco al di sotto dell'80% .

Questo scenario era ampiamente prevedibile dato che i nodi cambiano con una frequenza maggiore la loro strategia.

Di seguito riporto i grafici che forniscono una stima delle proprietà topologiche



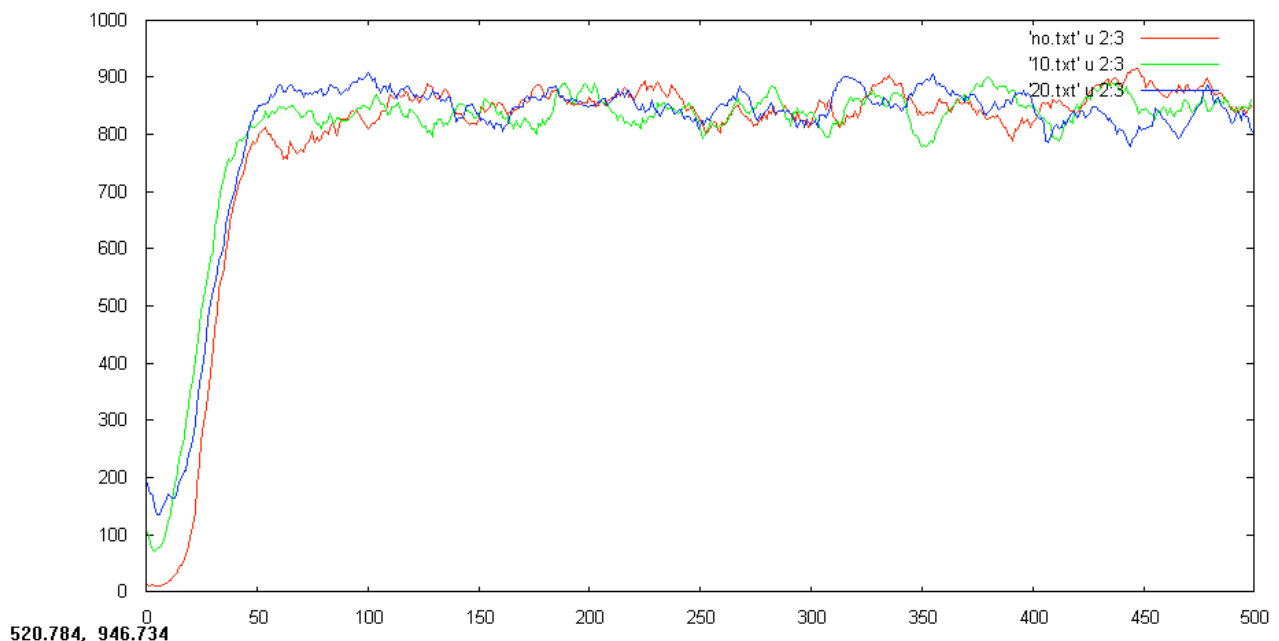


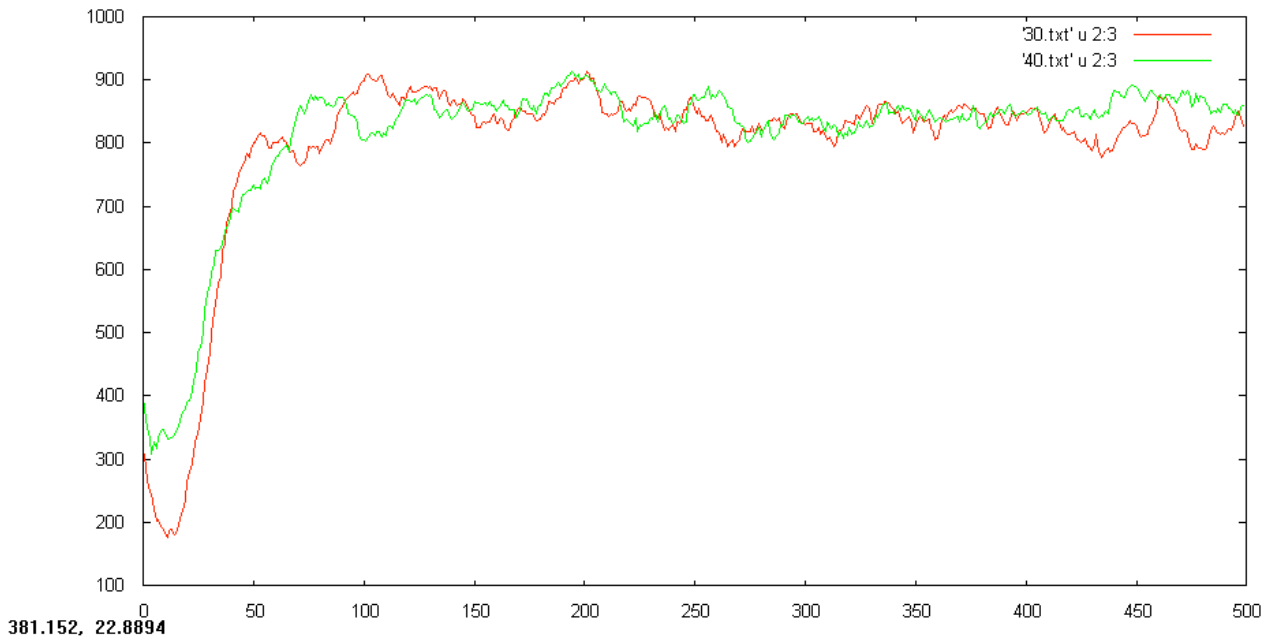


E'interessante vedere come il CCPL, ovvero la lunghezza media dei cammini cooperativi, sia nettamente inferiore al APL, ovvero la lunghezza media di tutti i cammini possibili . Questo ci lascia immaginare uno scenario dove vi siano alcune zone cooperative distaccate da nodi non cooperativi, ma non molto estese dato che i valori finali di LCC si attestano sul 40 % .

Risultati Network-size 1000, Cache 2 , Overlay WireKOut con % di nodi cooperativi iniziale variabile .

In questa simulazione ho variato più volte il parametro “quantity” che indica la percentuale iniziale di nodi cooperativi.



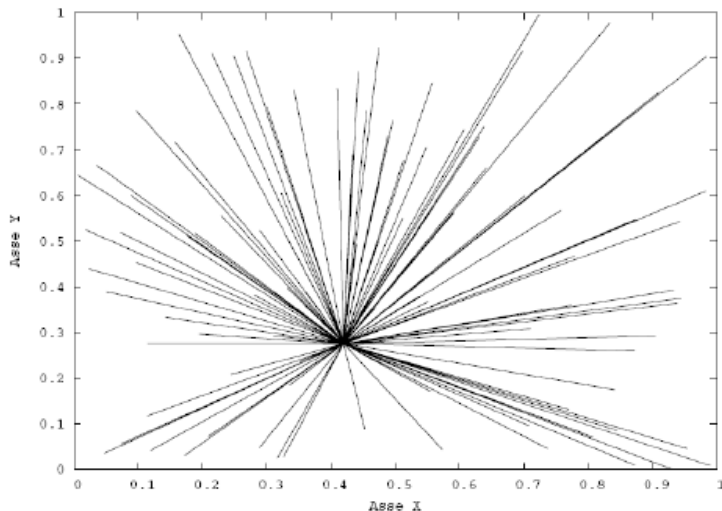


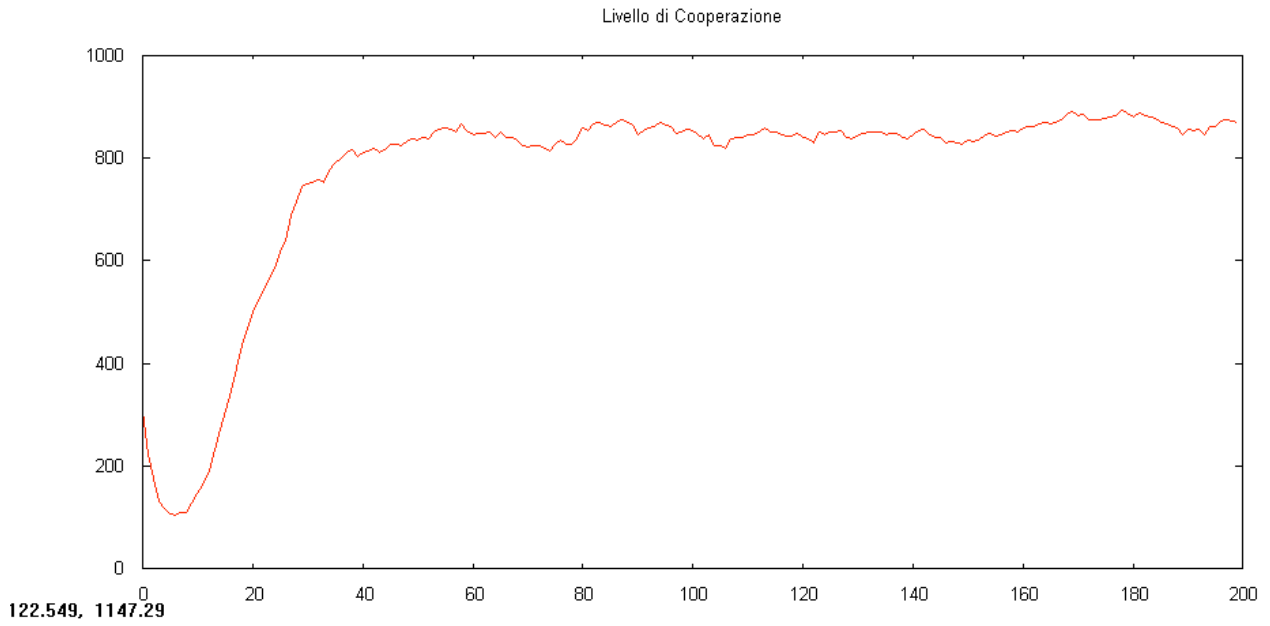
Le due immagini illustrano i nodi cooperativi in funzione dei cicli della simulazione (in questo caso 500) . Da questi dati si può evincere che indipendentemente dal numero di nodi cooperativi iniziali (nessuno, 10%, 20%, 30%, 40%) la rete evolve subito (prima di 50 cicli) verso uno stato di alta cooperazione oscillando, dopo un centinaio di cicli, verso valori compresi tra l'80 % ed il 90 % di cooperazione, che è un discreto risultato.

Alla luce di queste considerazioni, la mia analisi si concentrerà su una simulazione con uno stato iniziale del 30 % di nodi cooperativi, dato che, come abbiamo descritto, l'evoluzione delle rete è molto simile indipendentemente dallo stato iniziale. Inoltre l'aver rilevato un alto livello di cooperazione in un tempo relativamente breve, ci assicura una certa robustezza delle rete a situazioni di "collasso" che si potrebbero venire a creare.

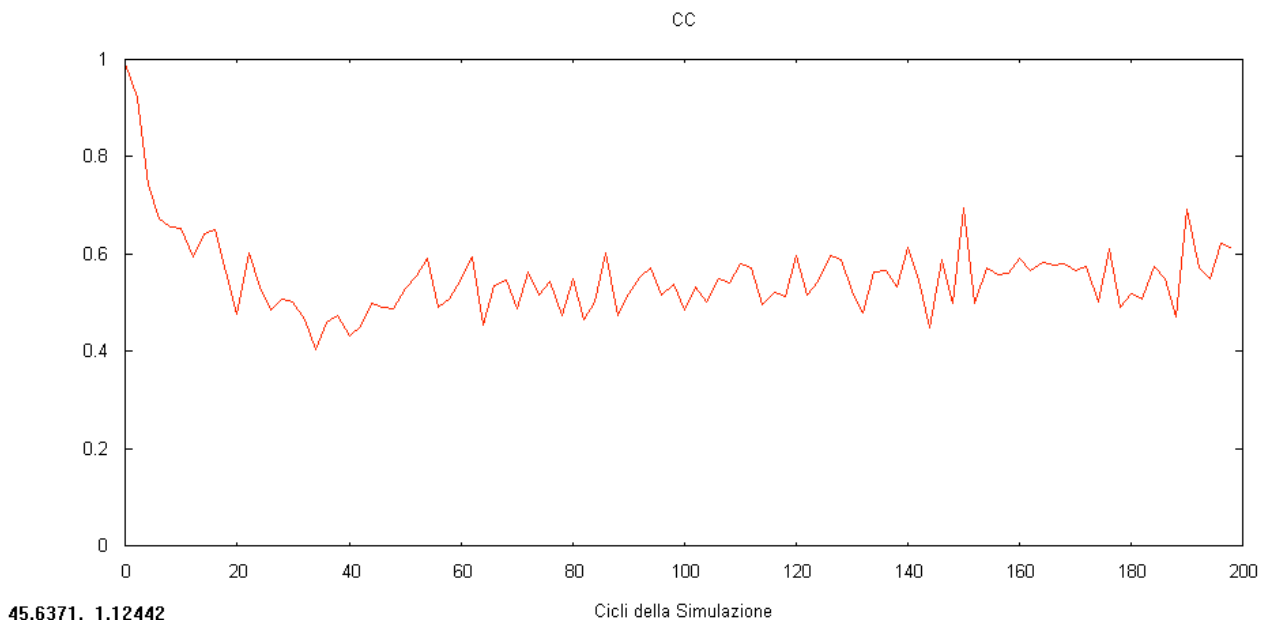
Risultati Network-size 1000, Cache 20 , Overlay WireStar

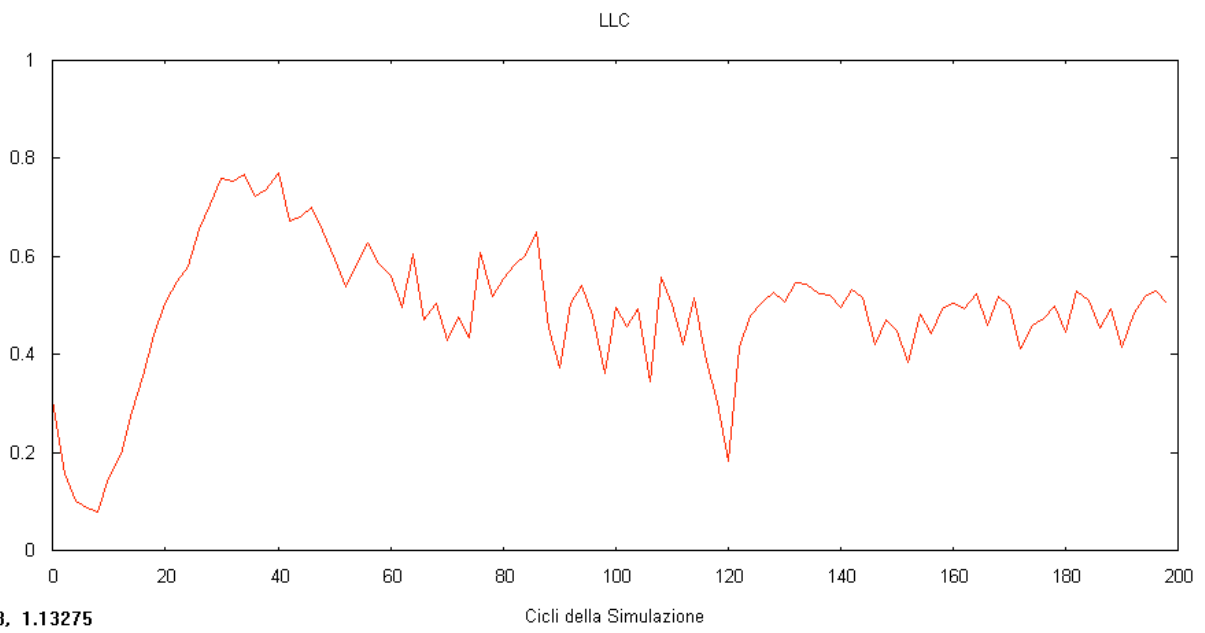
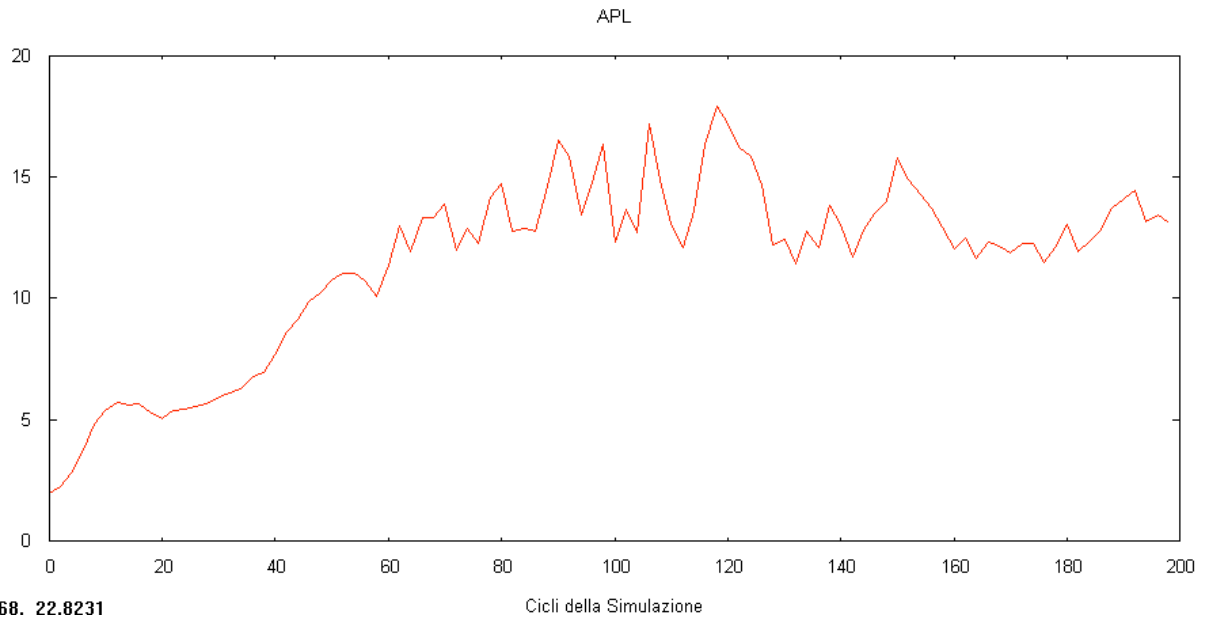
In questa simulazione è cambiata la topologia di rete iniziale, si è scelto di usare la WireStar che fornisce una struttura sicuramente più rigida e particolare. Riporto di seguito un possibile scenario iniziale :



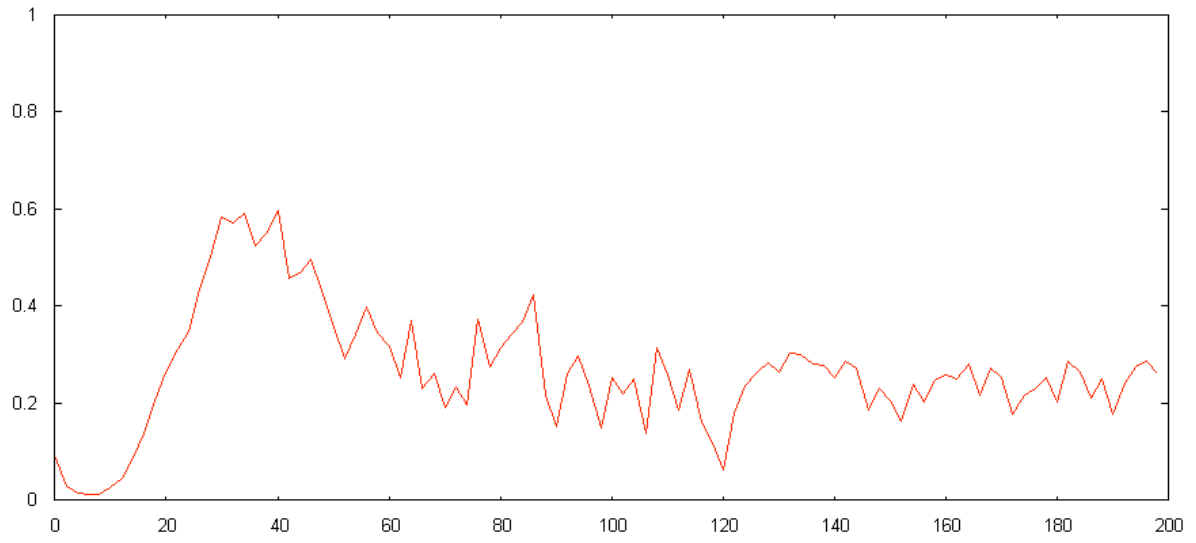


Nonostante la diversa topologia, i risultati ottenuti sono del tutto simili. Prove empiriche evidenziano come si ottengono risultati abbastanza simili anche utilizzando il WireRingLattice.





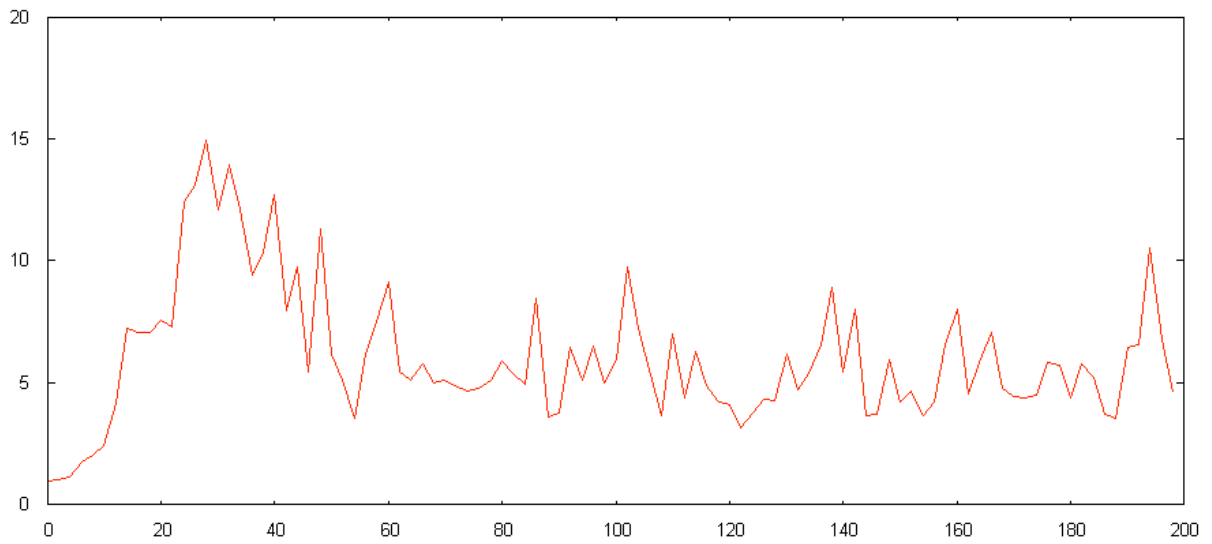
CCP



78.9768, 1.08538

Cicli della Simulazione

CCPL



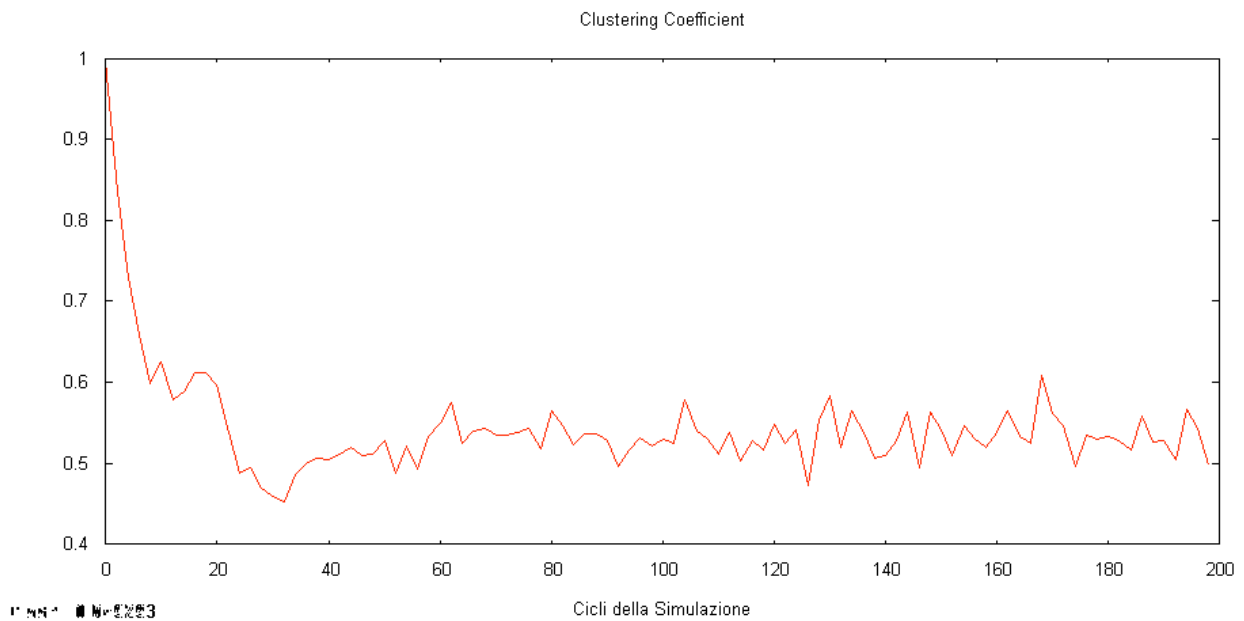
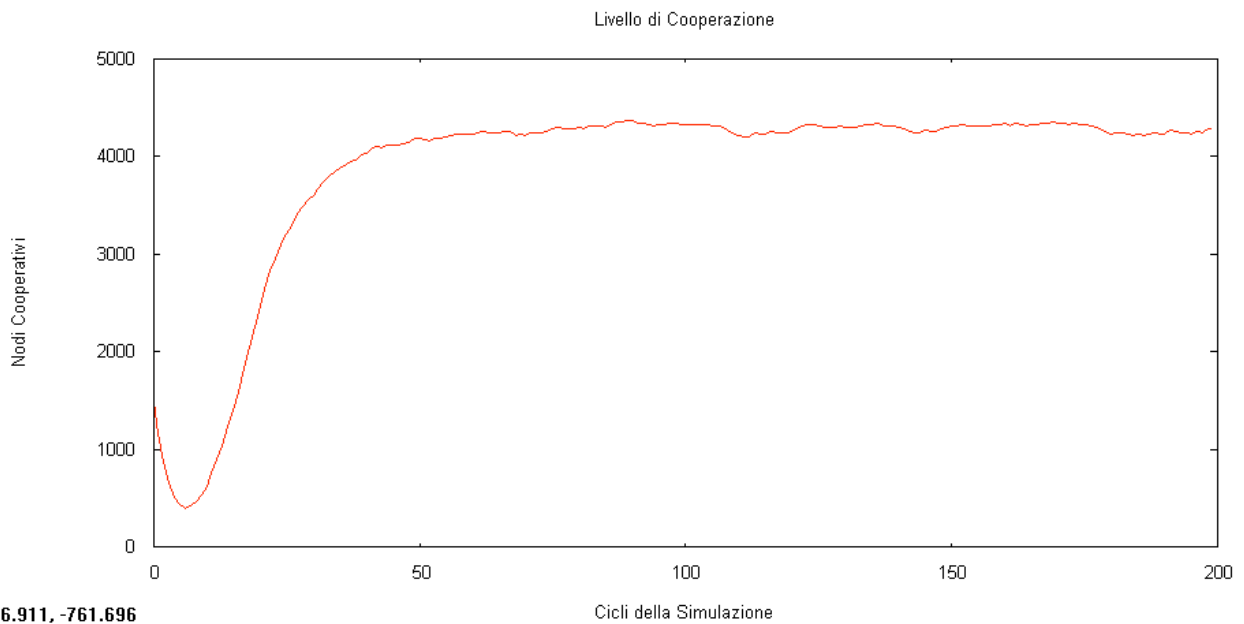
42.7091, 23.1579

Cicli della Simulazione

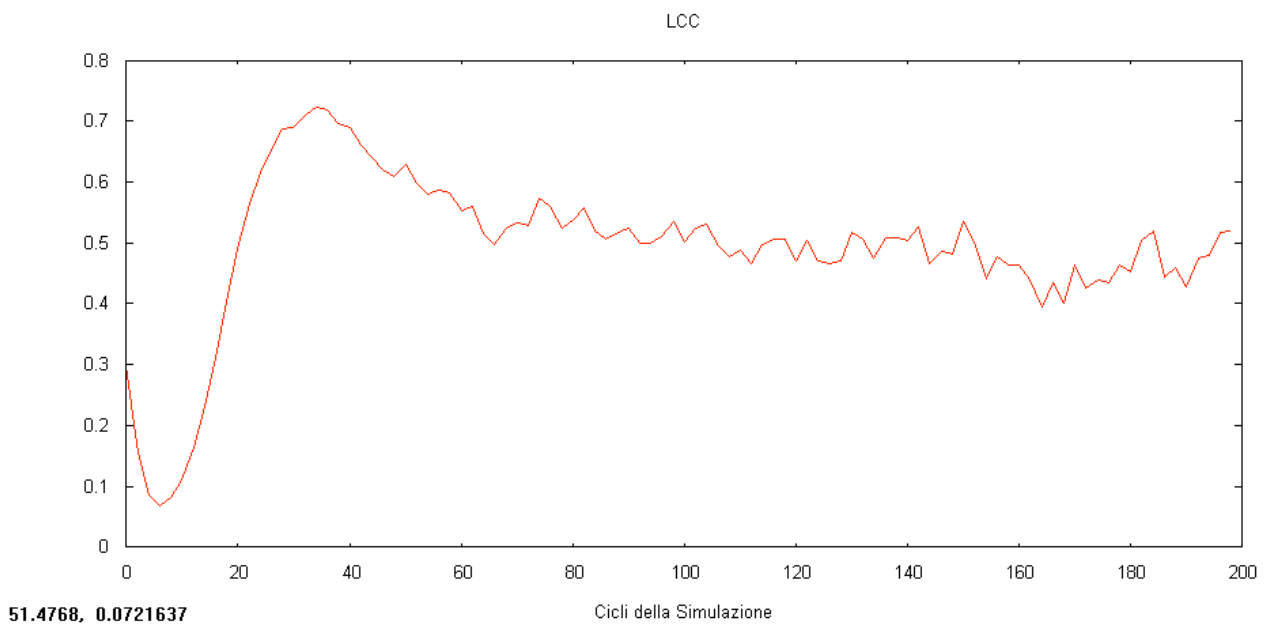
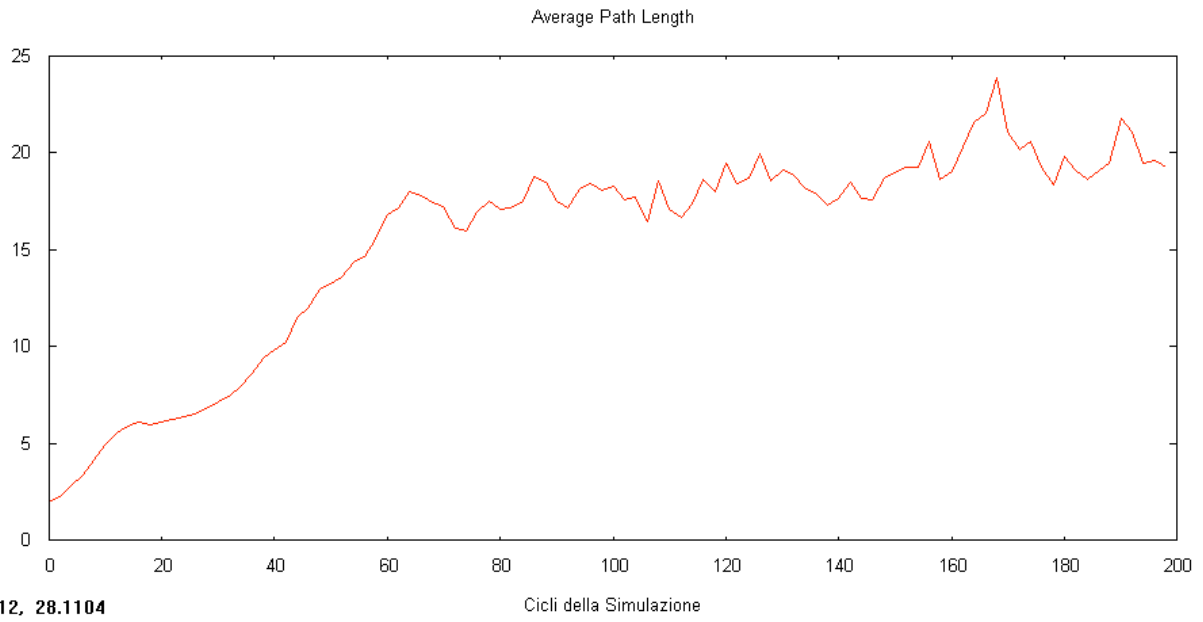
Risultati Network-size 5000, Cache 2 , Overlay WireStar.

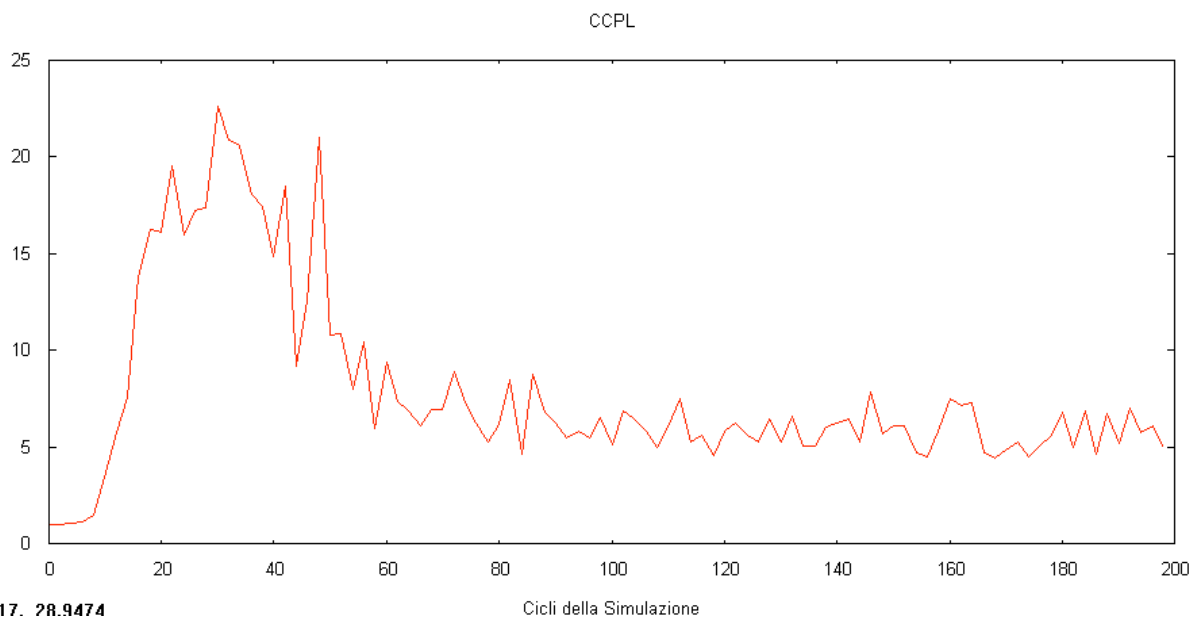
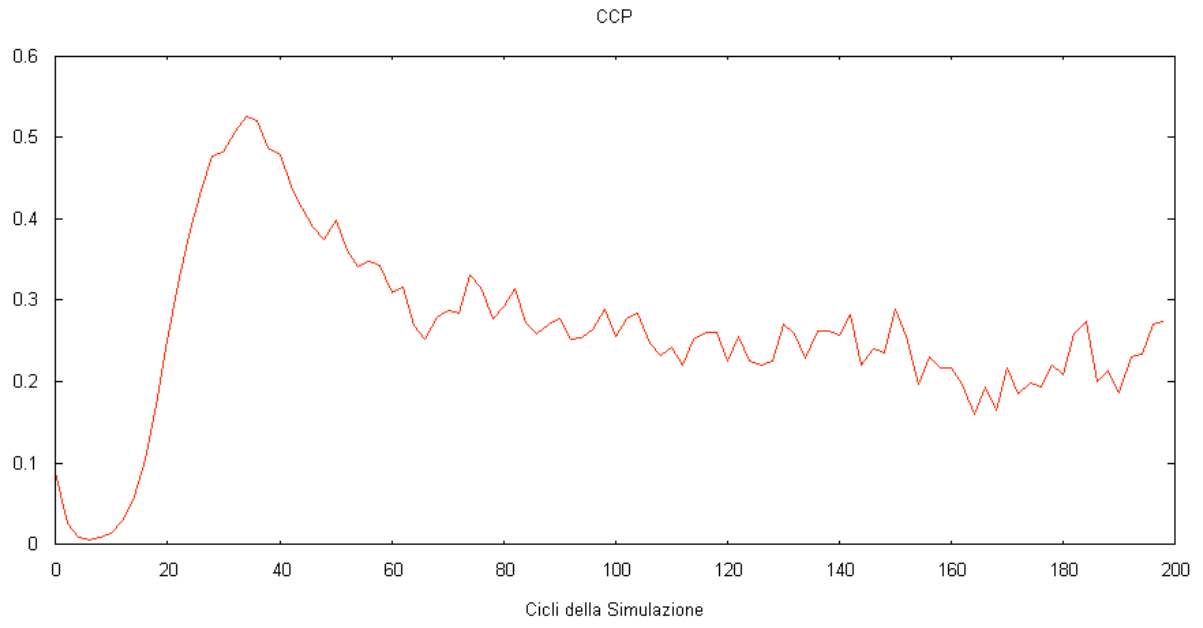
In questa simulazione è stata variata la dimensione delle rete ed il suo valore è stato impostato a 5000.

Paradossalmente la rete giunge ad un stato di alta cooperazione leggermente prima rispetto ad i precedenti test.



Il coefficiente di clustering si aggira sempre verso lo 0.5



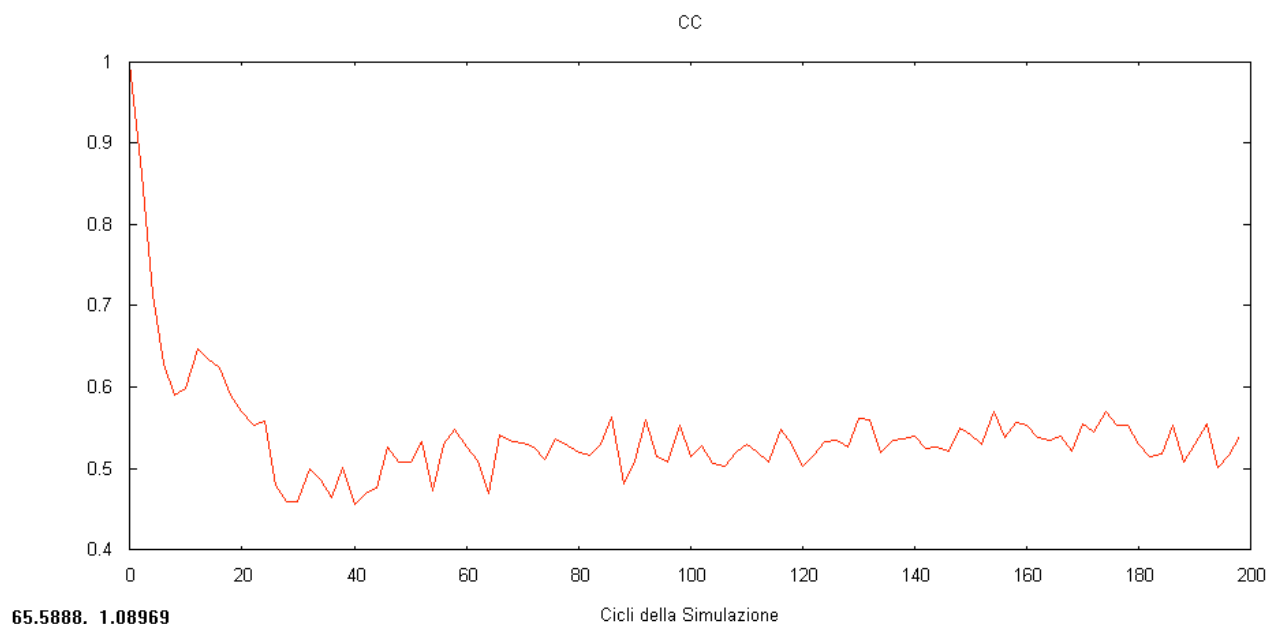
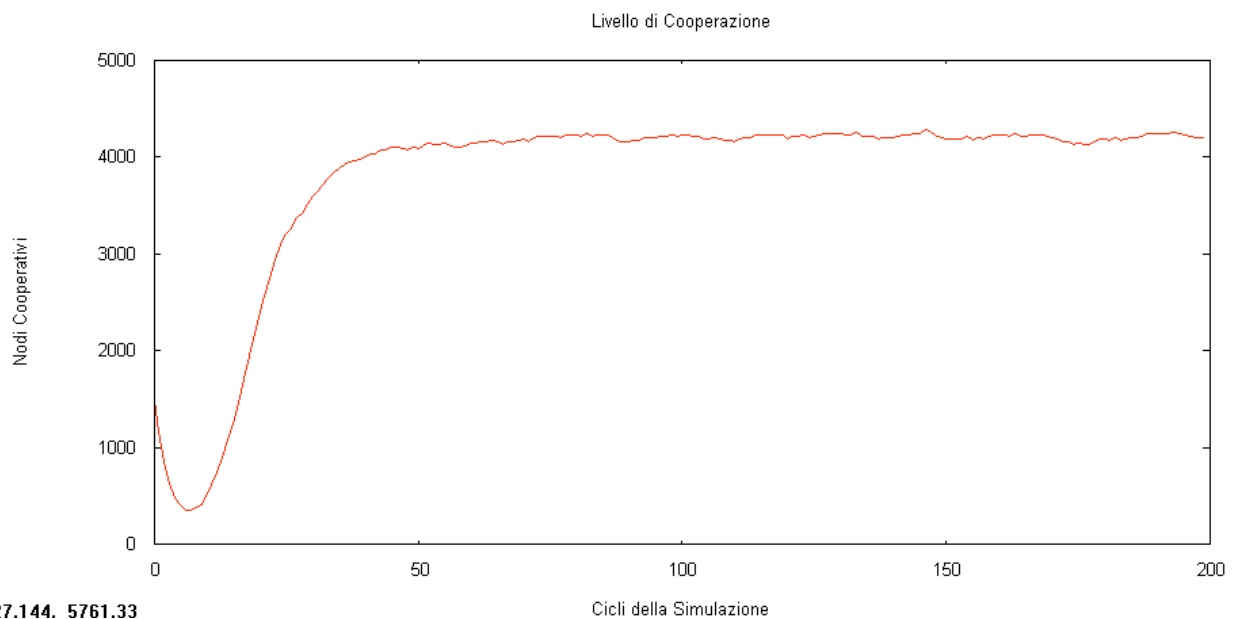


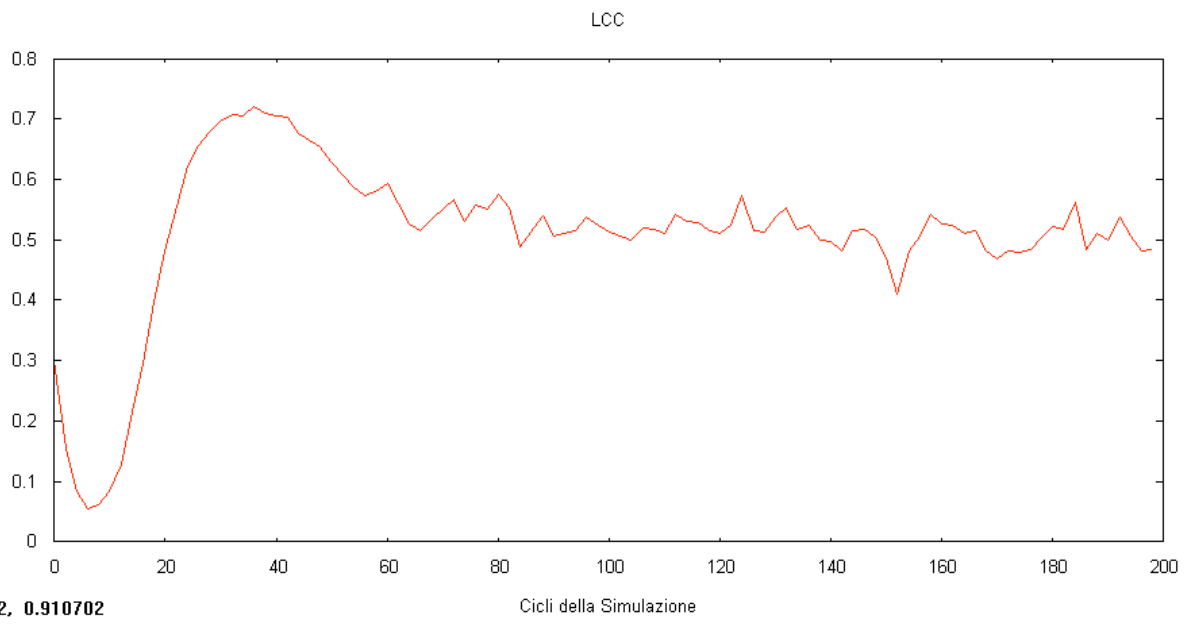
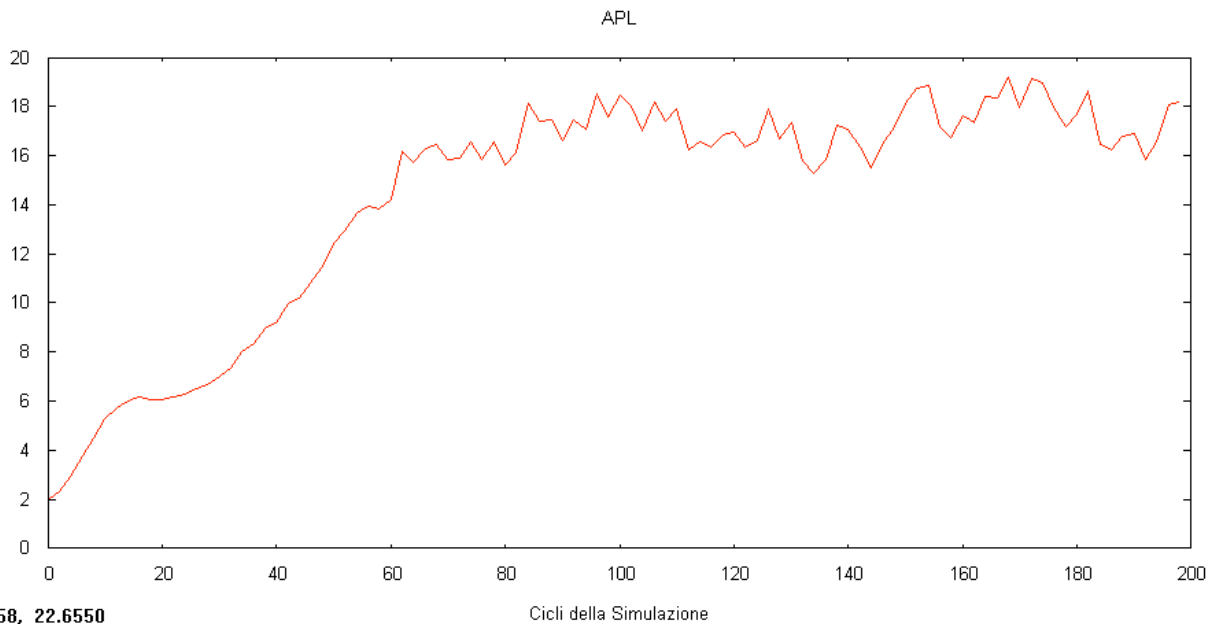
74.5817. 28.9474

Considerazioni su test successivi.

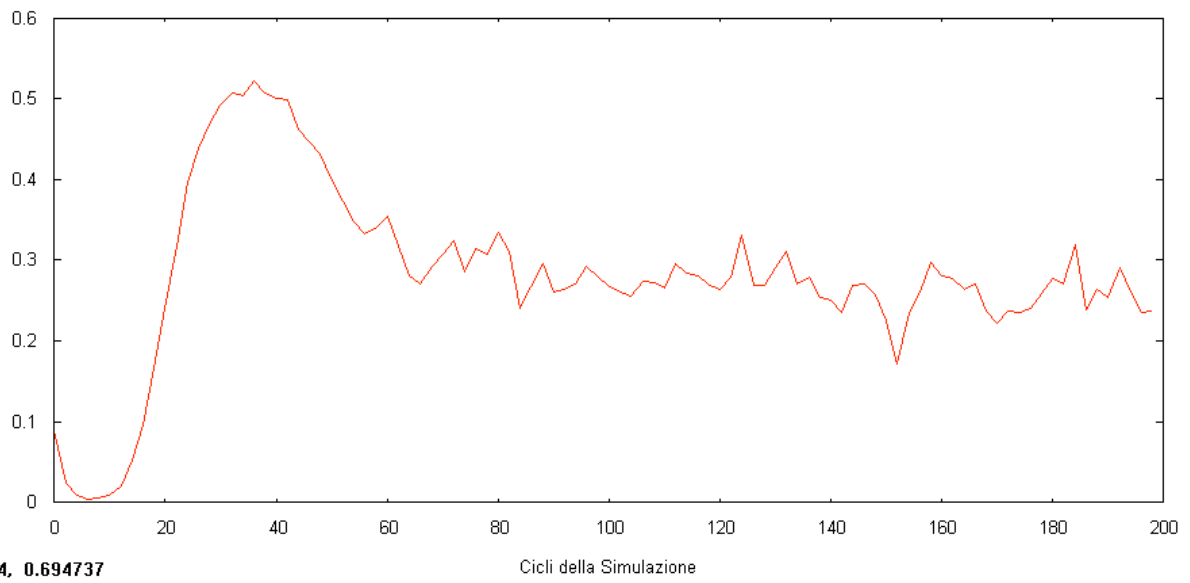
Nei successivi test ho variato i parametri di Network Size, cache oltre che l'overlay network .
Ho ottenuto però valori molto simili in tutte le simulazione, quindi ho ritenuto non necessario commentare tutti grafici. Come già accennato in precedenza, tutti i test raggiungono un discreto livello di cooperazione, ma valori molto elevati e non desiderati per quanto riguarda l'APL, o valori molto bassi di CPP. Successivamente ho fatto delle considerazione sul cambiamento dei parametri in funzione della grandezza di rete. Riporto per completezza, di seguito i risultati dei test .

Risultati Network-size 5000, Cache 40 , Overlay WireStar

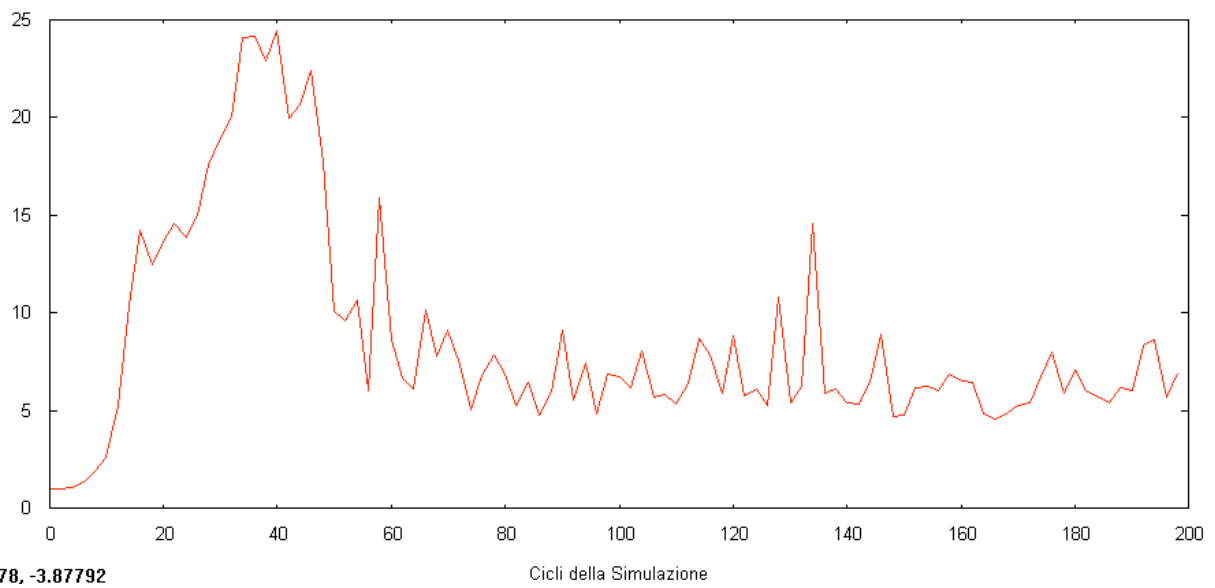




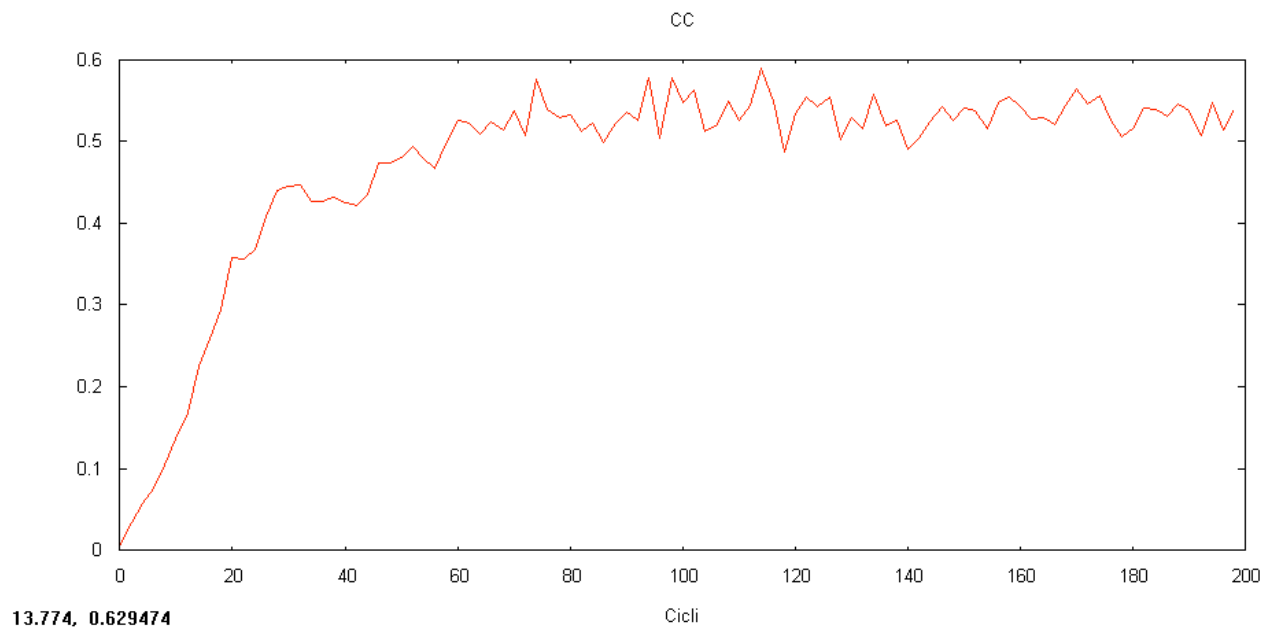
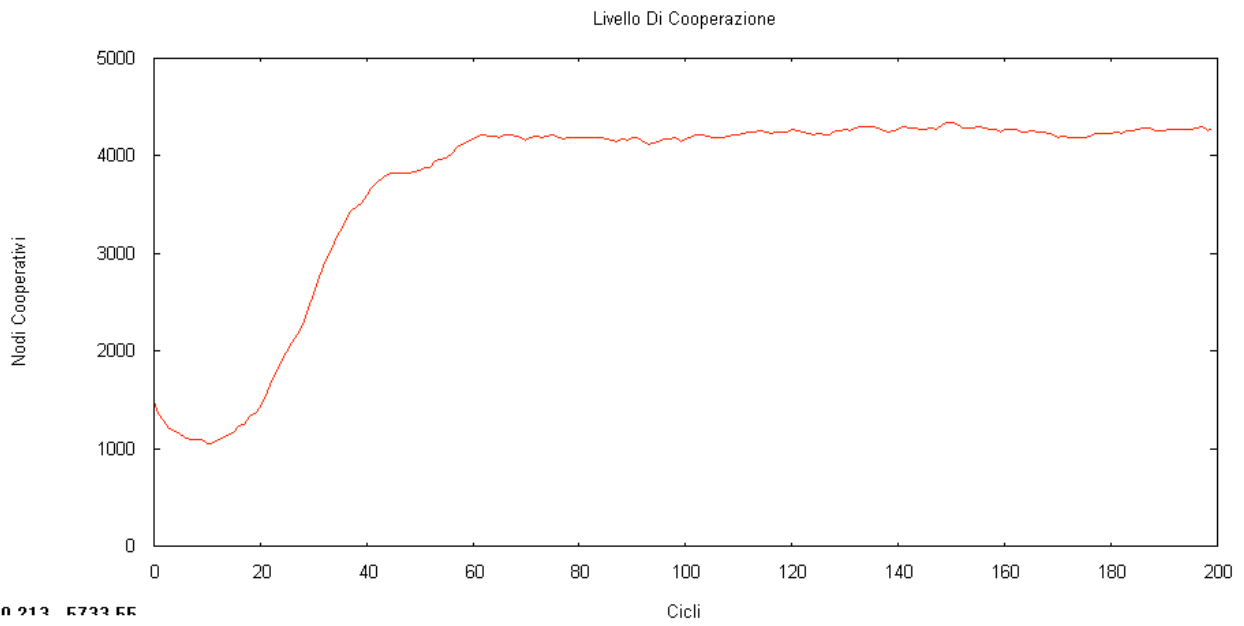
CCP



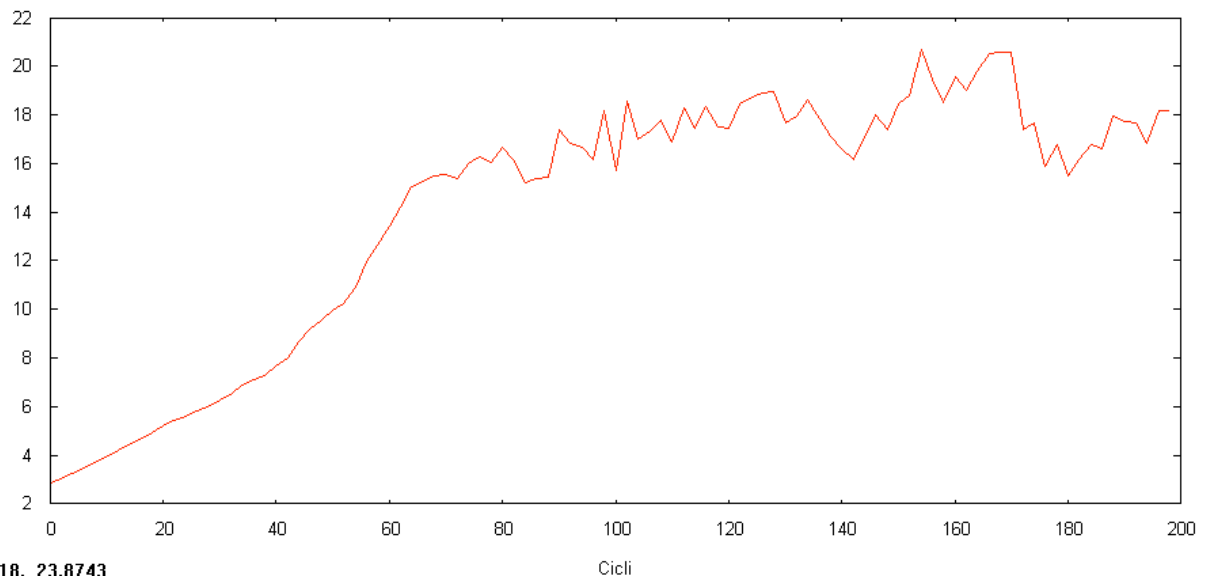
CCPL



Risultati Network-size 5000, Cache 20 , Overlay WireKOut

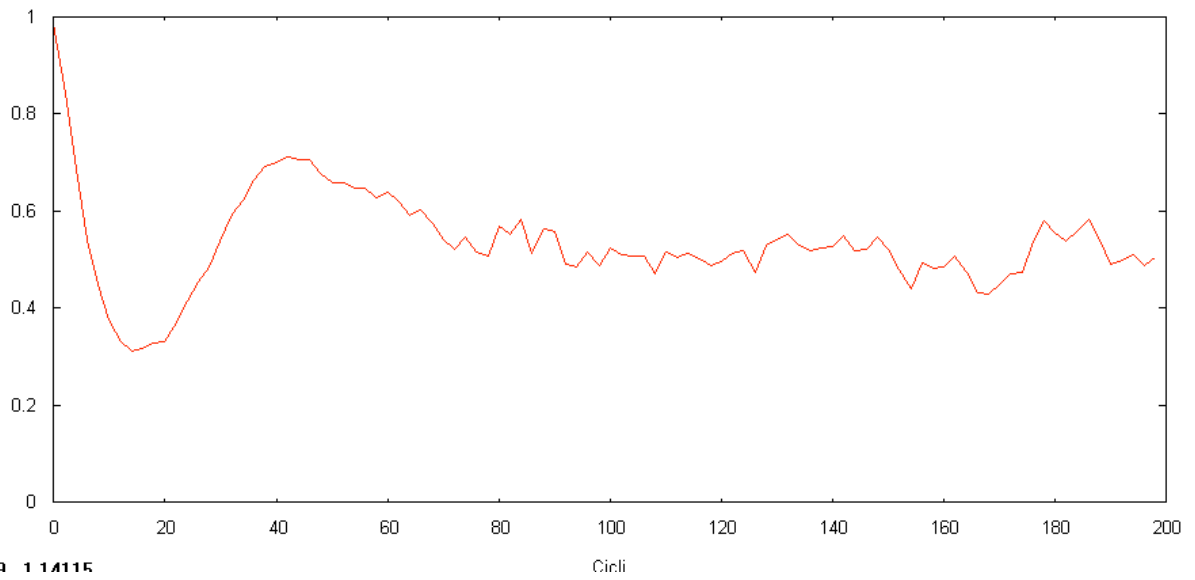


APL



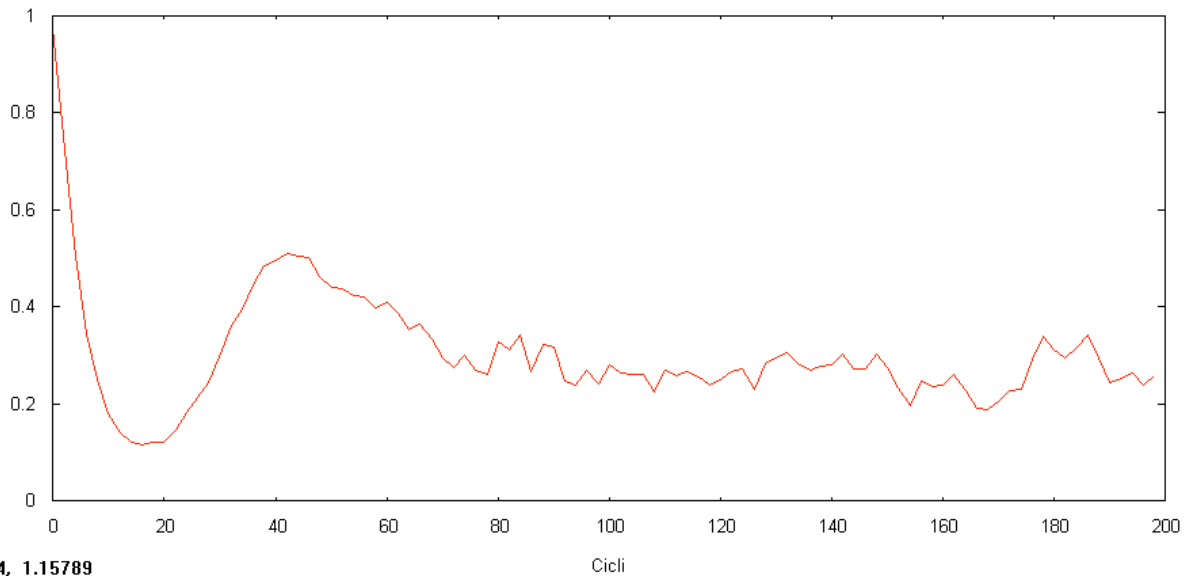
28.8118. 23.8743

LCC

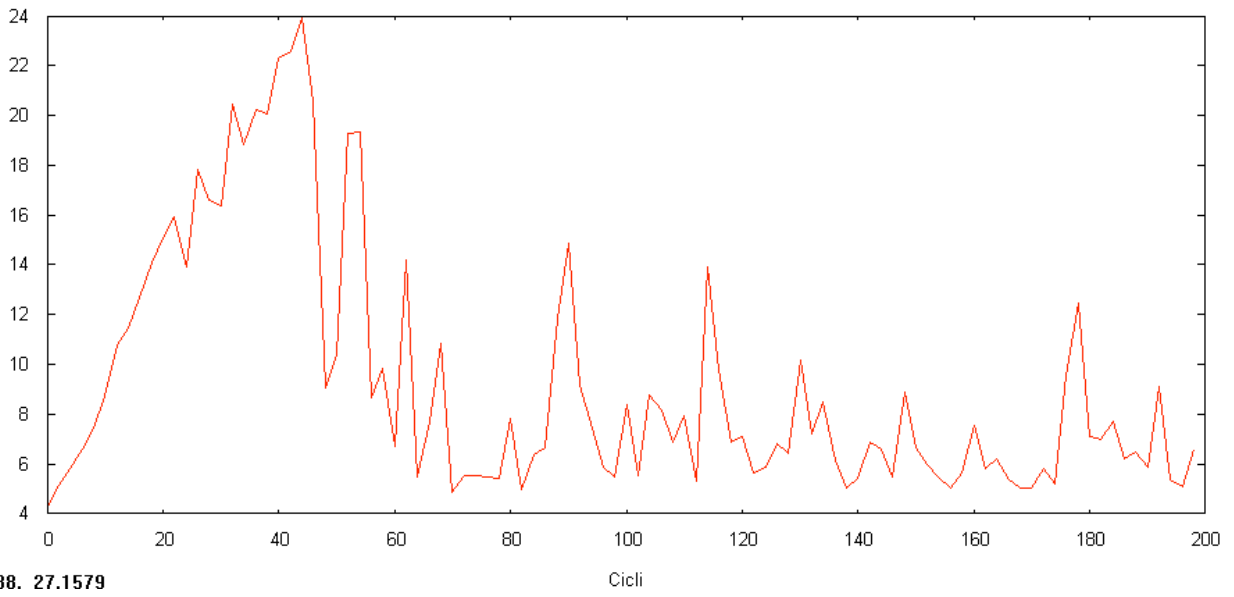


179.469 1.14115

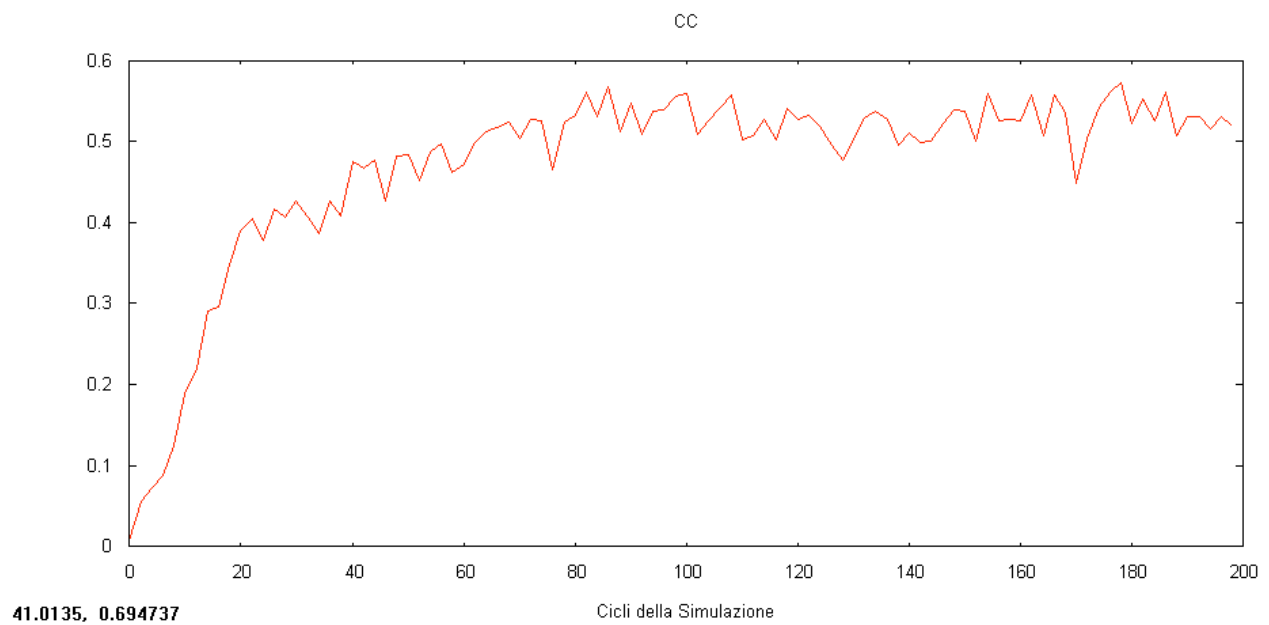
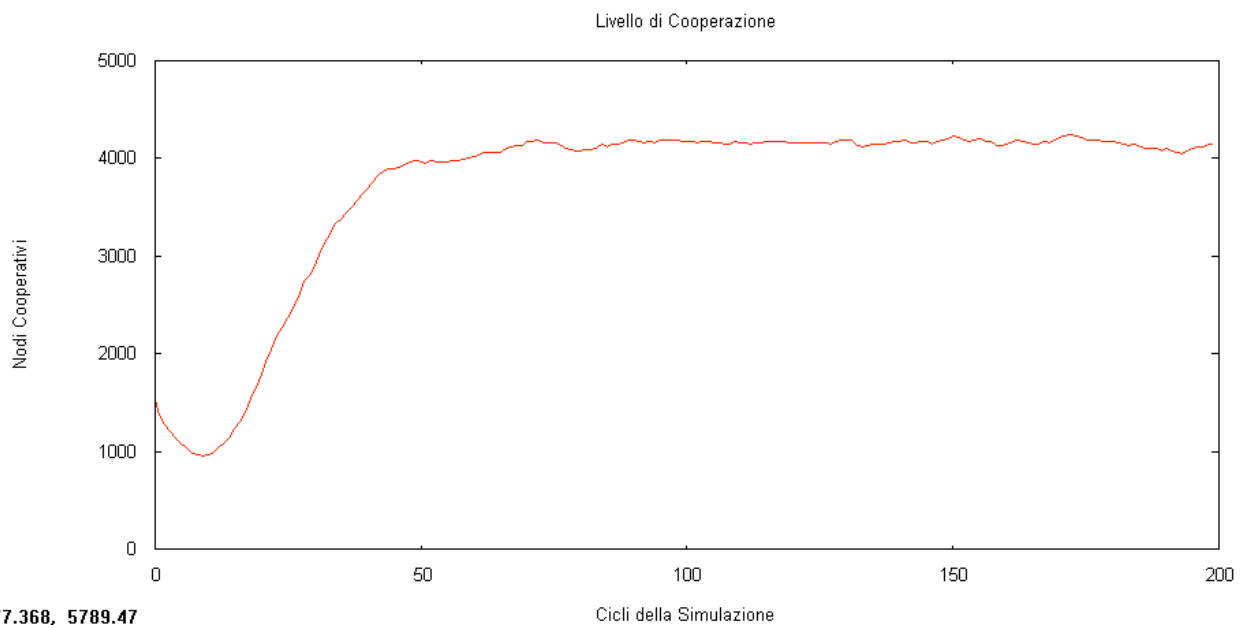
CCP

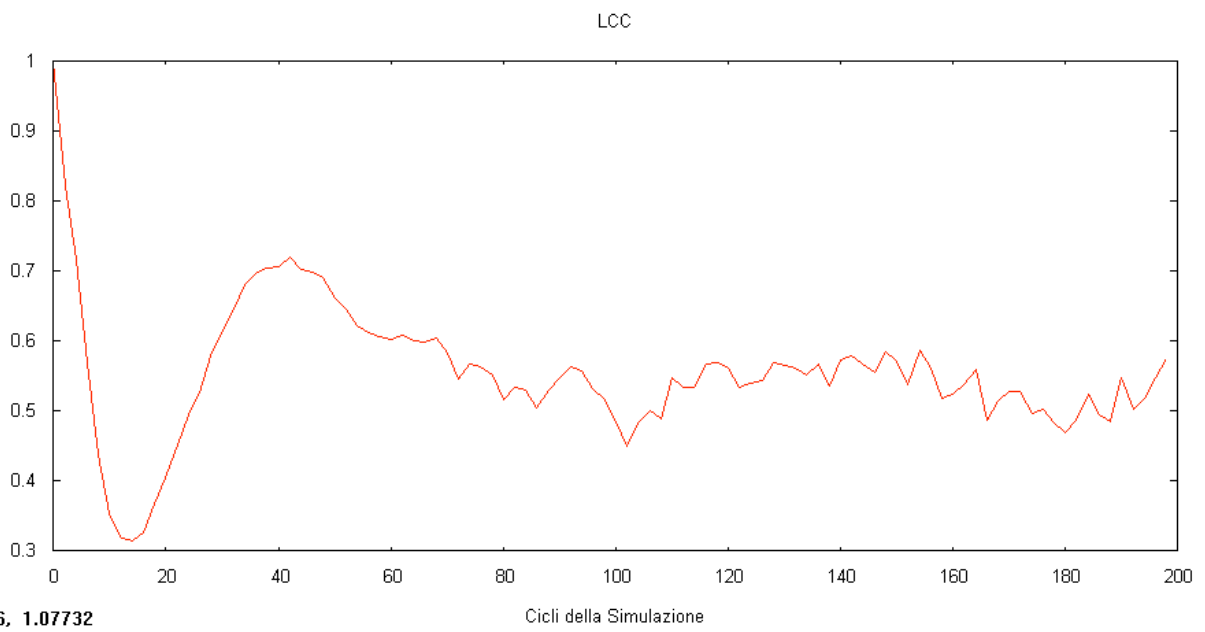
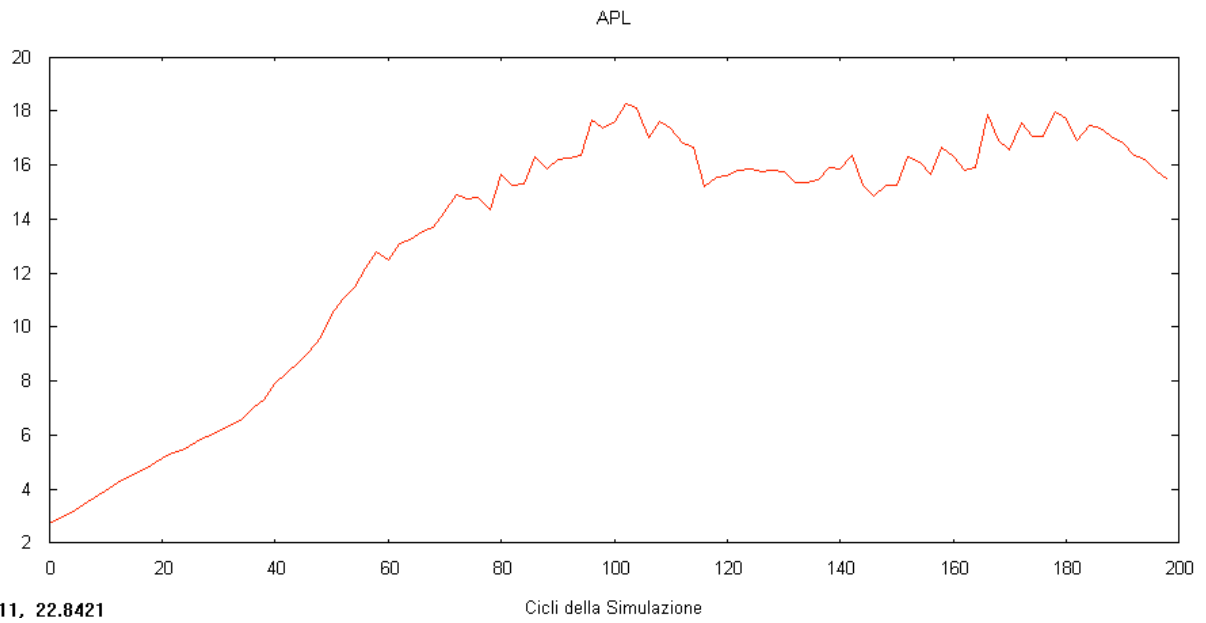


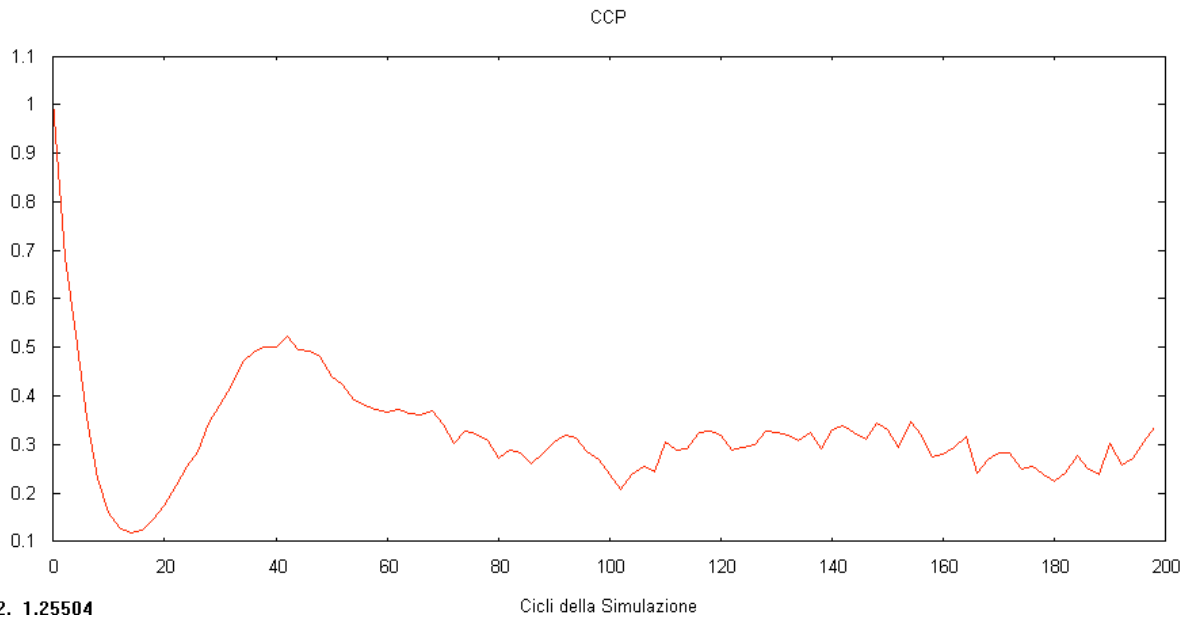
CCPL



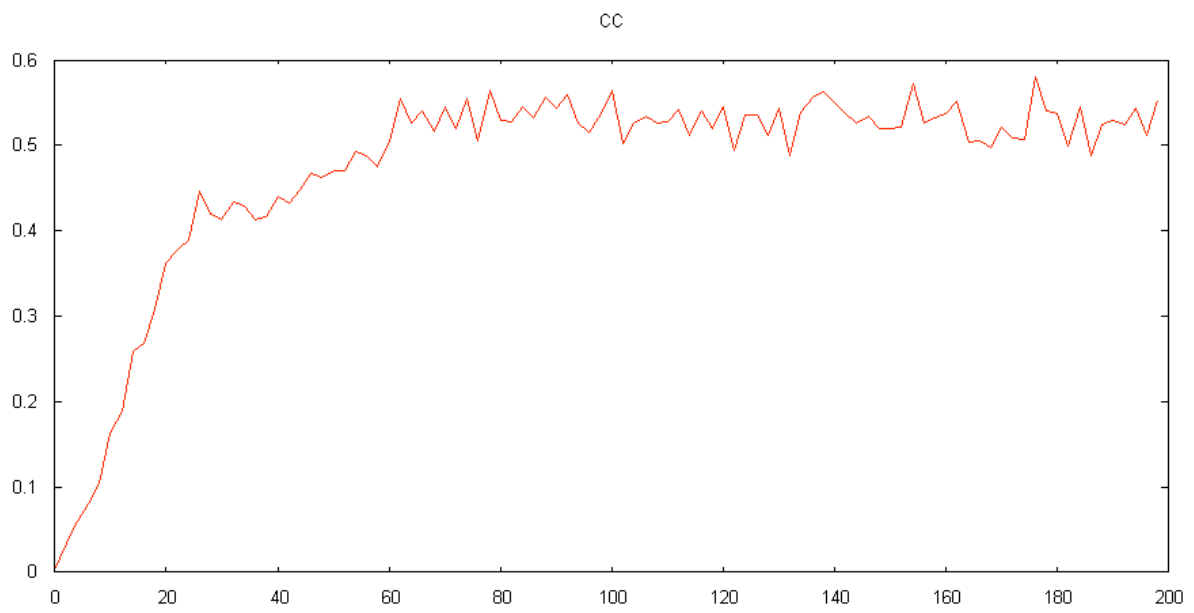
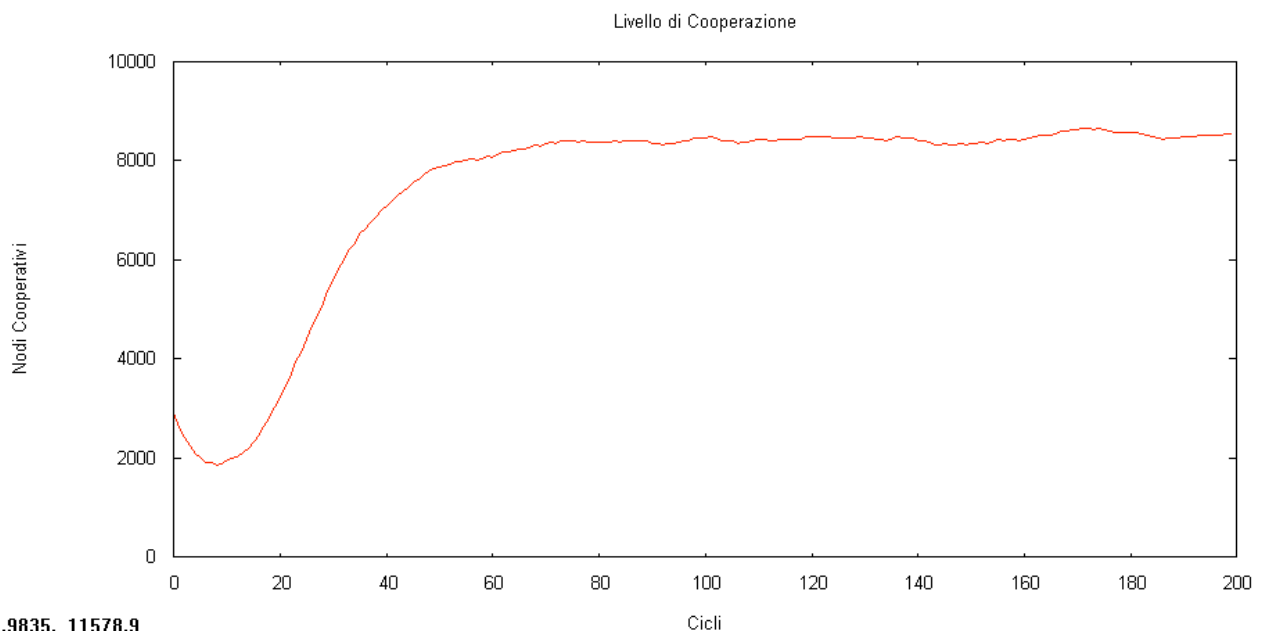
Risultati Network-size 5000, Cache 40 , Overlay WireKOut

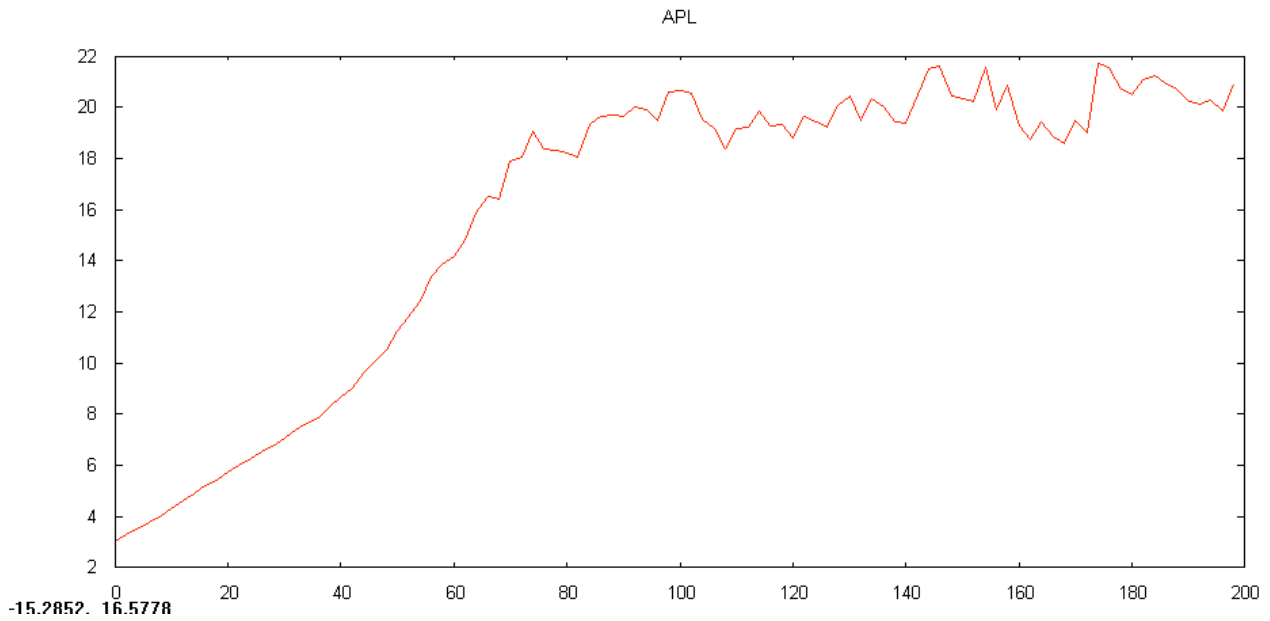




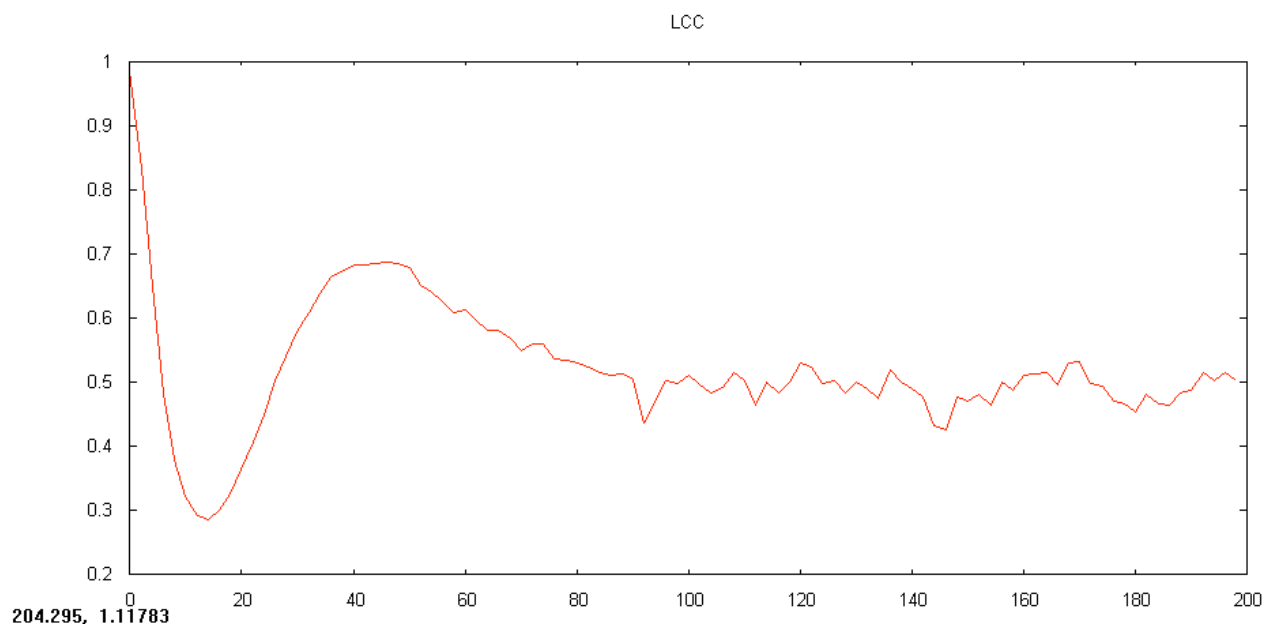


Risultati Network-size 10000, Cache 20 , Overlay WireKOut

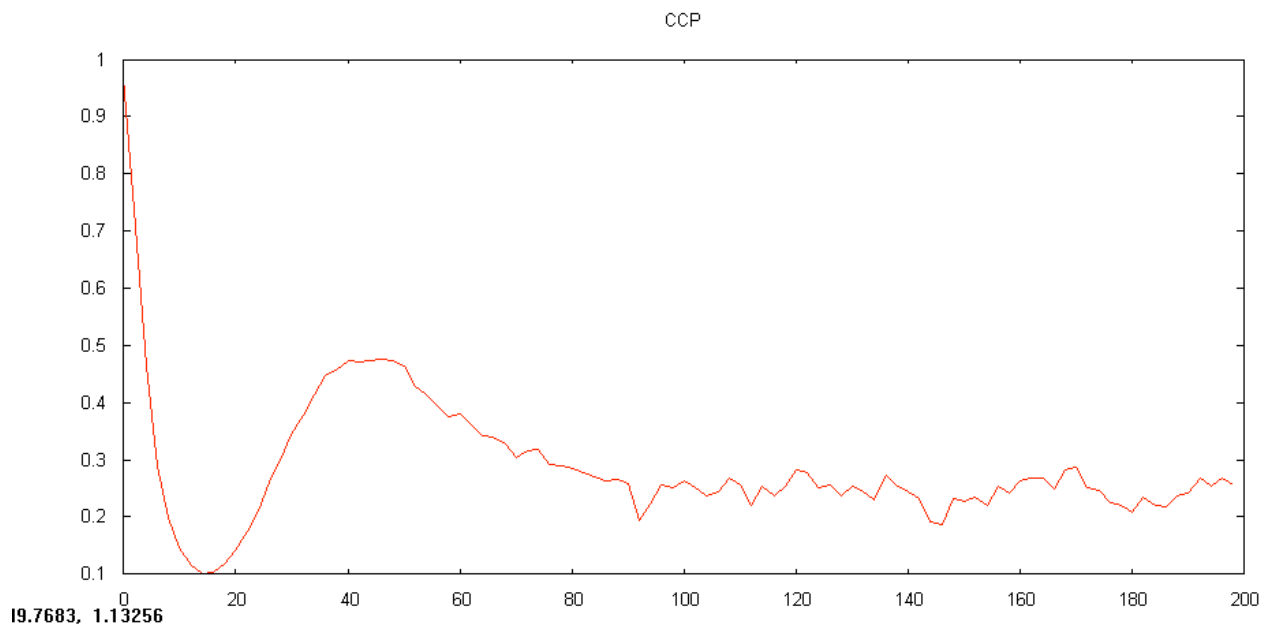




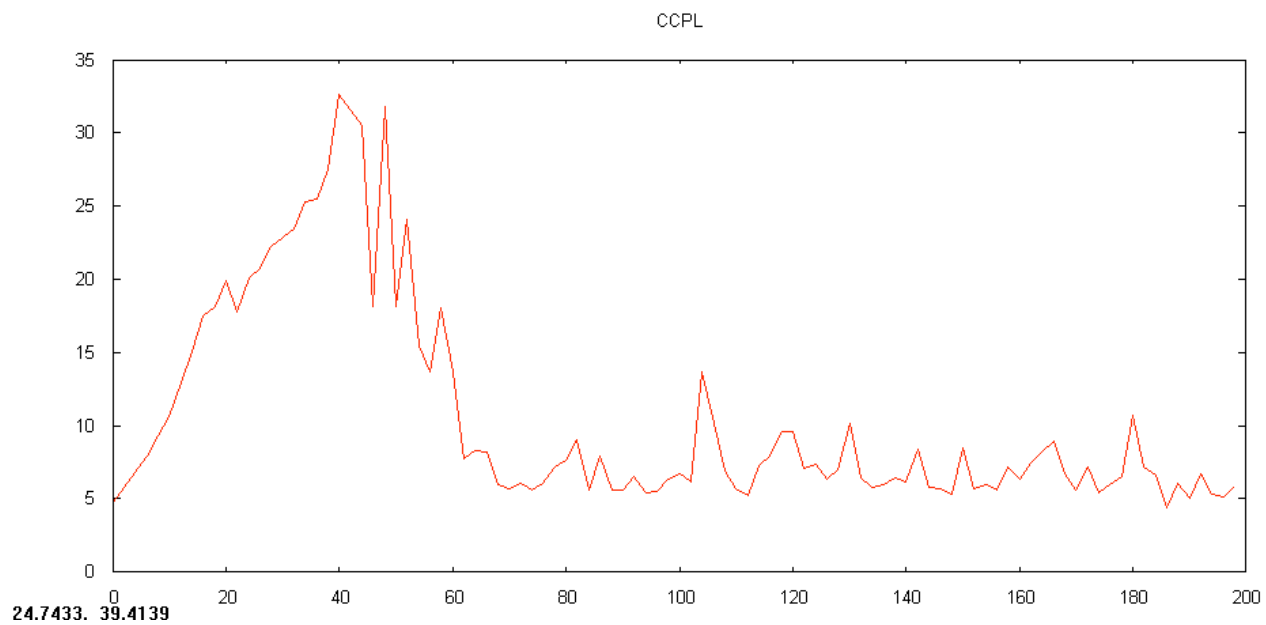
L'APL si attesta su livelli molto più alti rispetto alle precedenti simulazione: le ragioni di tale comportamento sono da ricercare nella dimensione della rete.



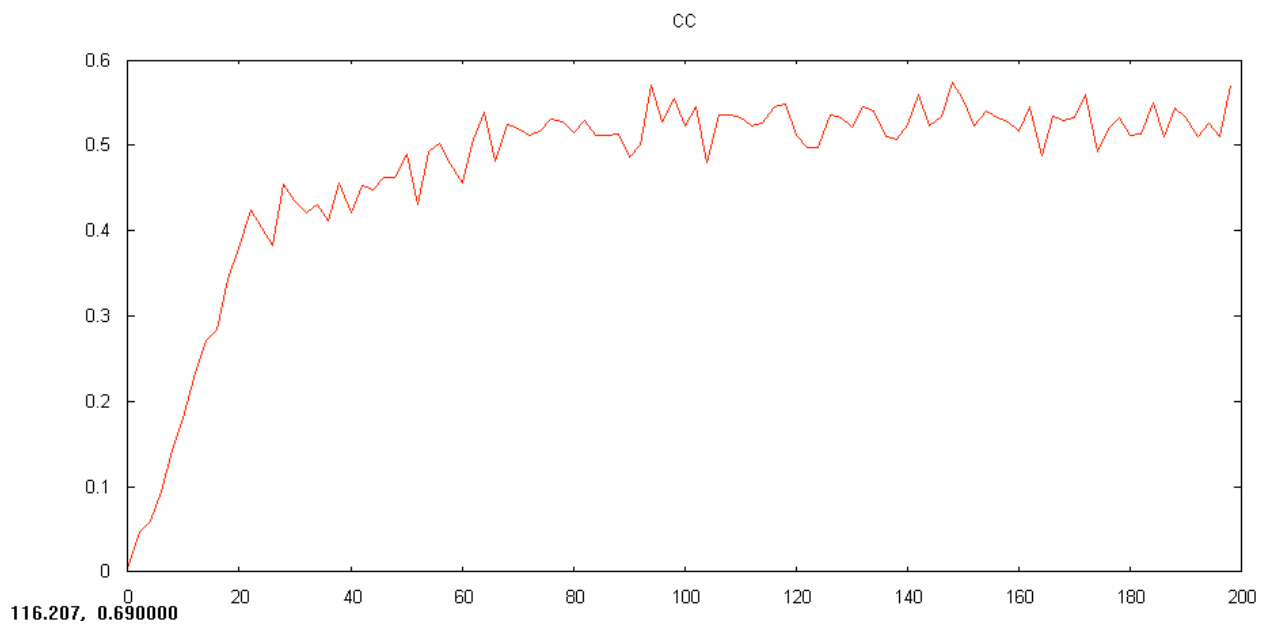
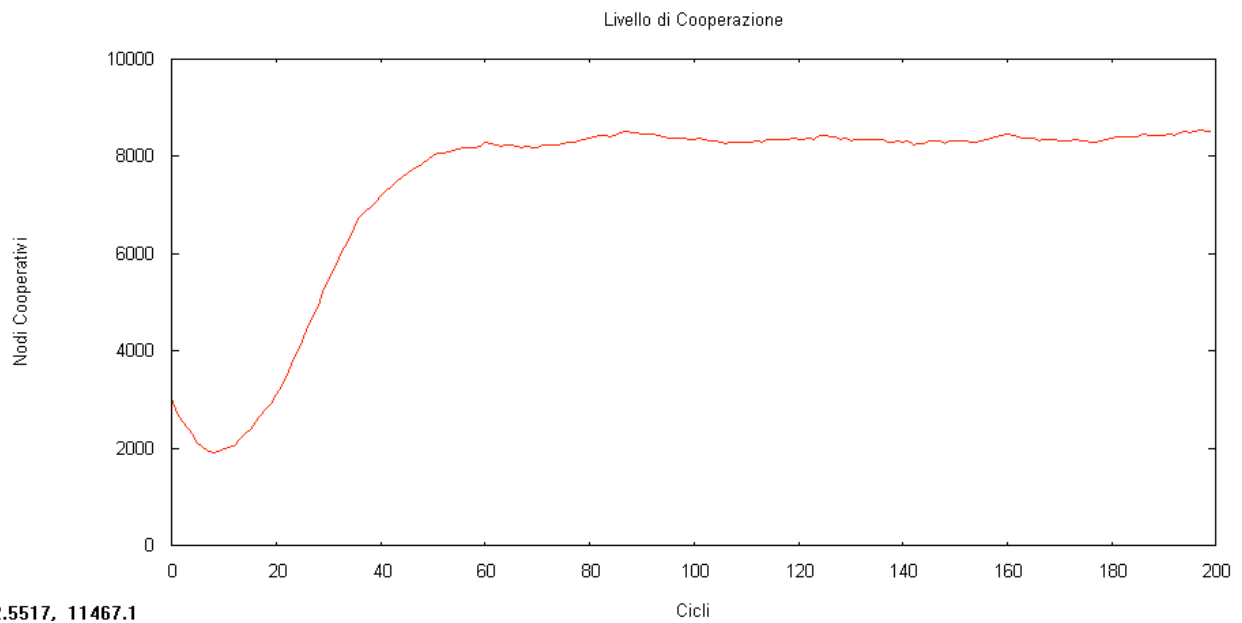
Il valore di LCC si attesta sempre su valori pari a 0.5: questo indica che è presente una zona cooperativa abbastanza estesa.



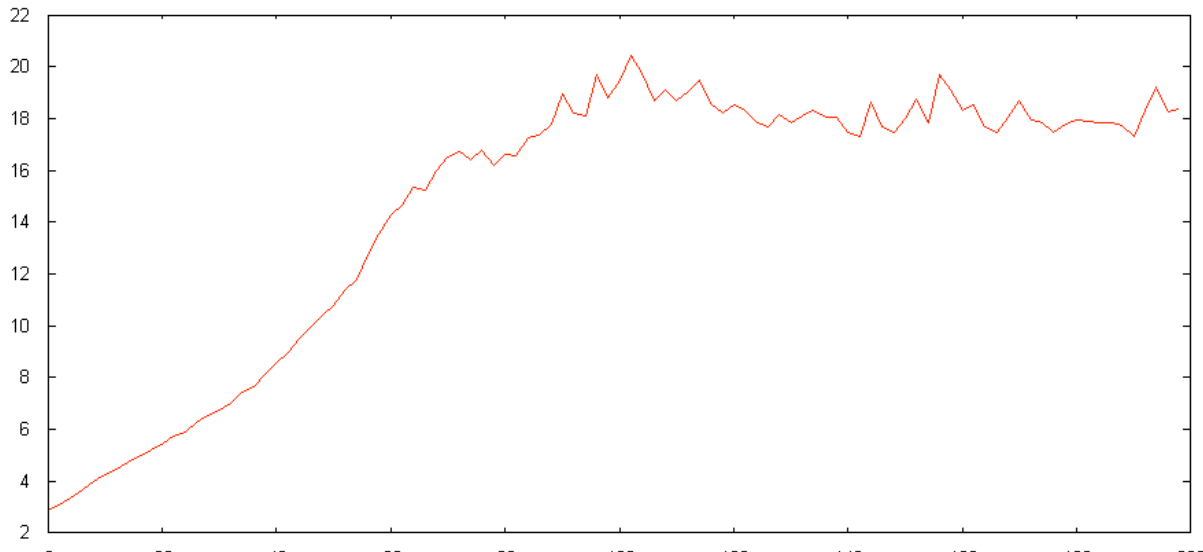
Il valore di CCP risulta basso (0,28 in alta cooperazione)



Risultati Network-size 10000, Cache 40 , Overlay WireKOut

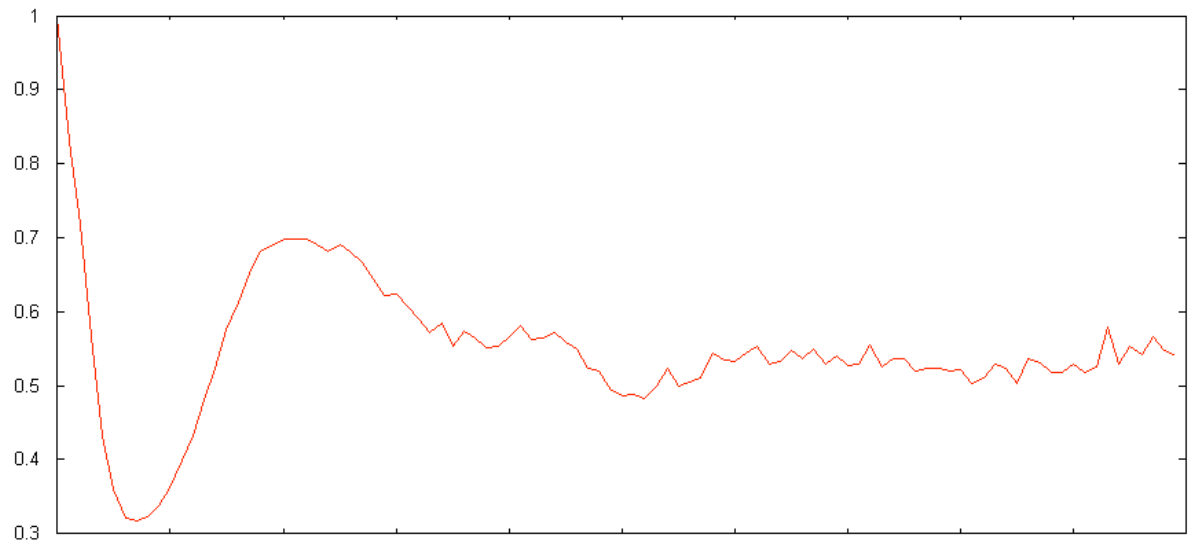


APL



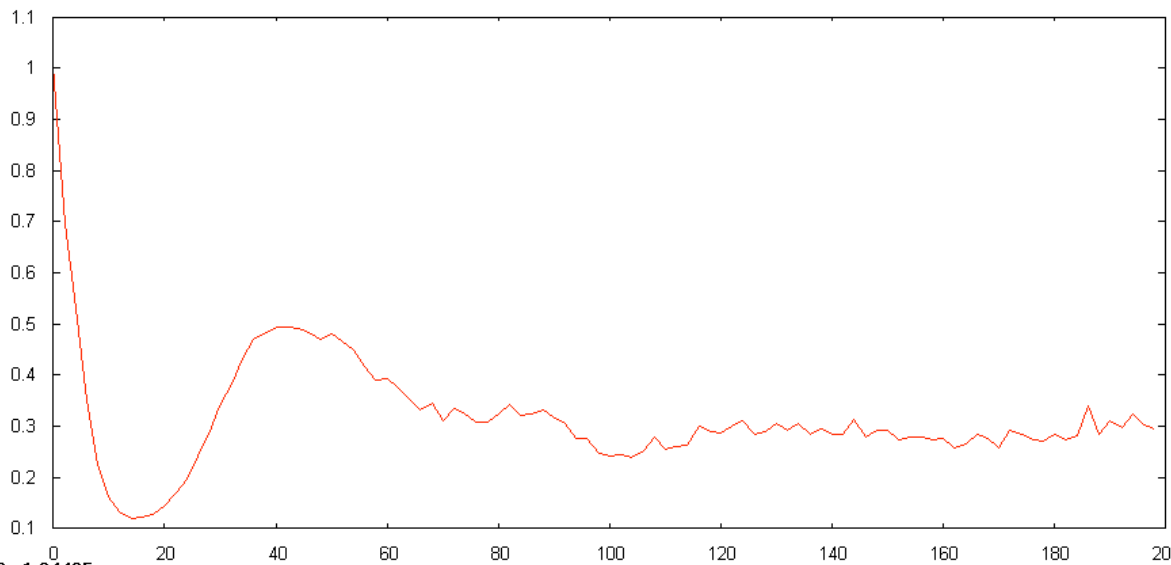
125.390, 24.9458

LCC



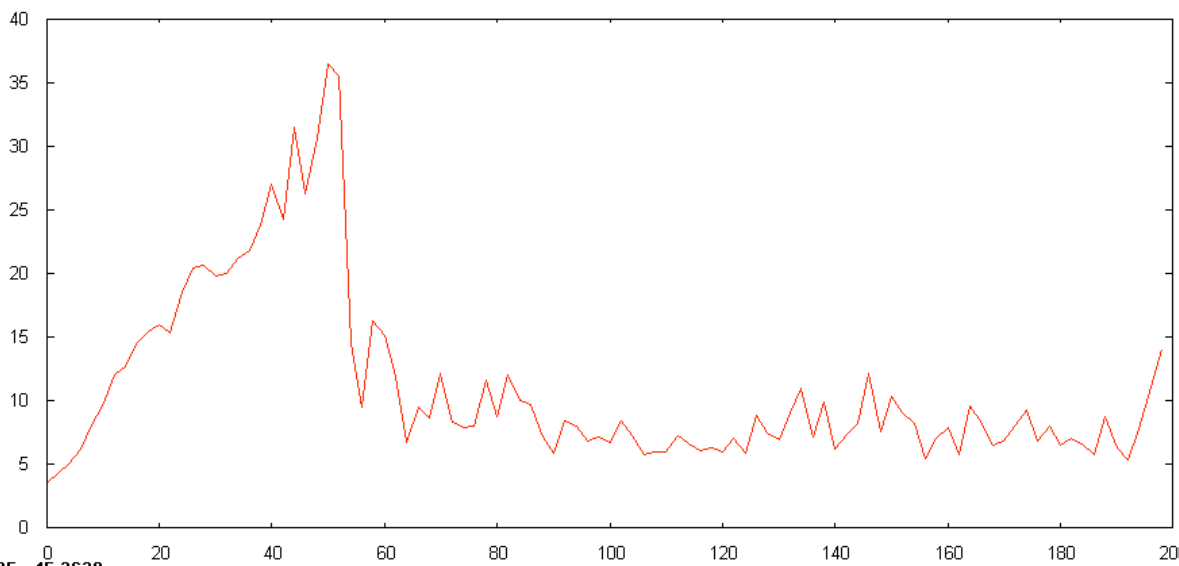
7.7799, 1.10500

CCP



100.386, 1.24465

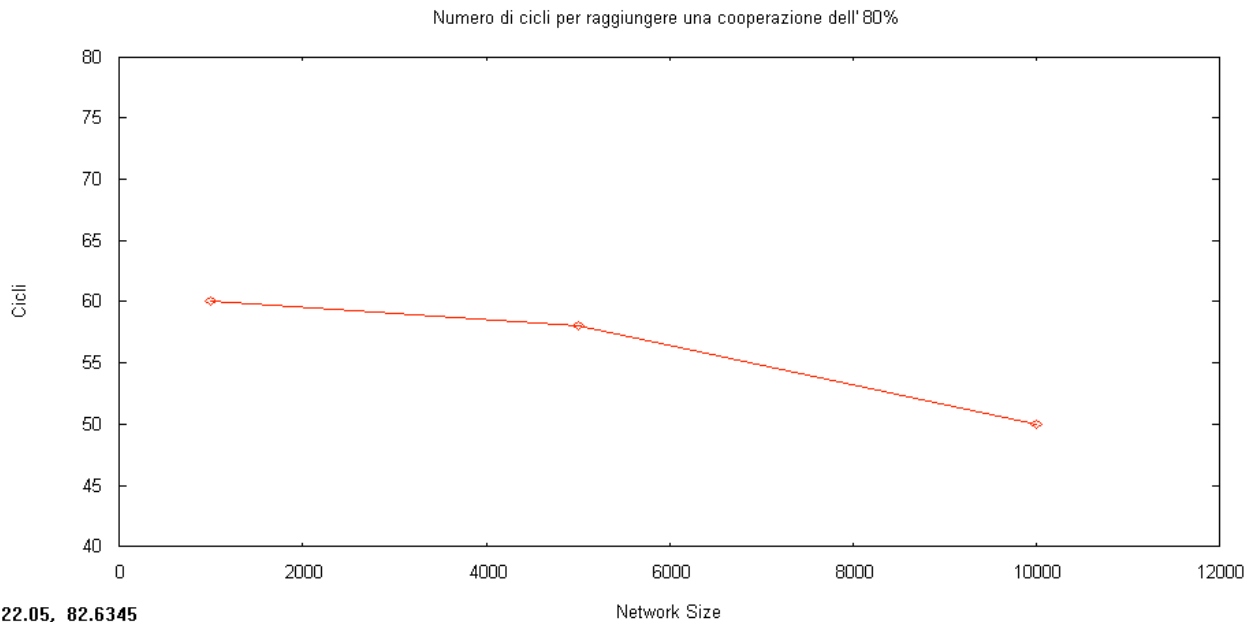
CCPL



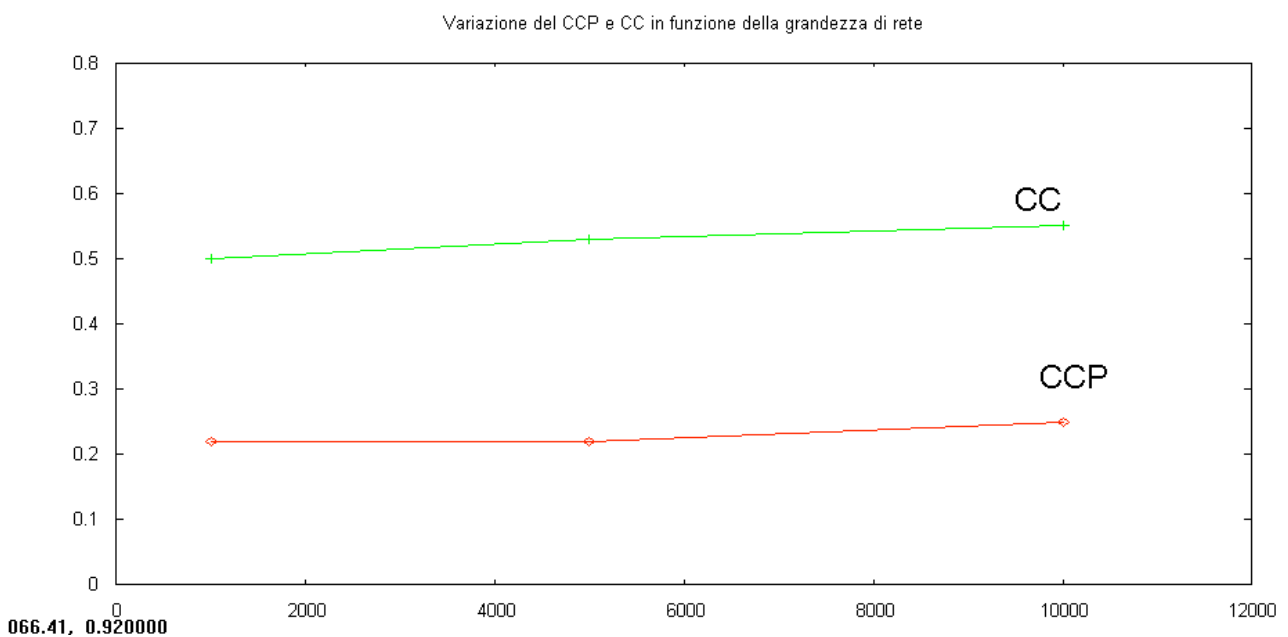
87.5285, 45.3639

Considerazioni di carattere generale

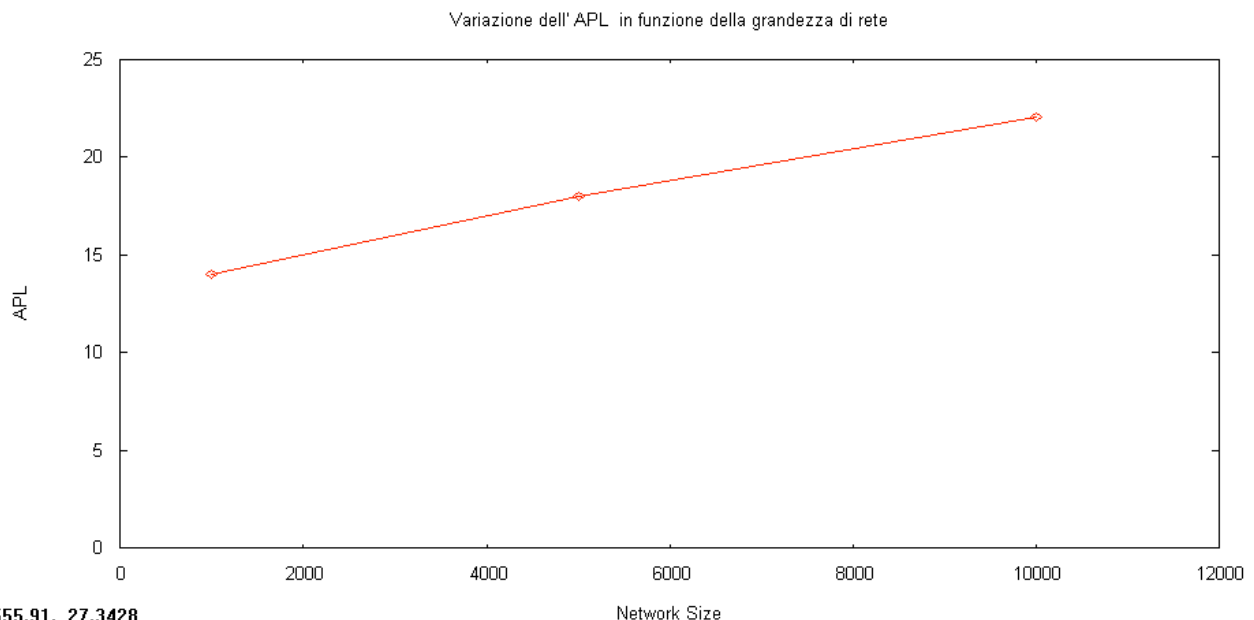
In ultima analisi ho analizzato i valori al variare della grandezza delle rete, per poter avere una stima della differenza di comportamento delle rete . L'overlay di partenza è il WireKOut.



In questa figura viene indicato il numero di cicli che la simulazione impiega per raggiungere un livello di cooperazione dell'80%. Paradossalmente, all'aumentare della grandezza delle rete diminuisce il numero di cicli necessari,



I valori di CC e CCP si attestano su valori molti simili indipendentemente dalla grandezza della rete. I valori sono stati analizzati alla fine della simulazione (200 cicli) ovvero quando si era già giunti ad un alto livello di cooperazione.



Il valore di APL tende ad aumentare all'aumentare della grandezza del network: questo comportamento è più che normale dato che cresce il numero di nodi e quindi il cammino medio è sicuramente maggiore. Come già accennato in precedenza si è ottenuto un alto APL nelle nostre simulazioni: questo è un comportamento sicuramente non desiderato per reti di questo tipo .

Le cause di questo comportamento sono da ricercare solo in parte nel particolare tipo di overlay di partenza; ritengo che un elevato APL sia il frutto del crearsi, all'interno delle rete, di una grossa componente cooperativa, come illustra il valore di LCC (pari quasi sempre a 0,5 in tutti i test) e di piccole zone periferiche che fanno aumentare notevolmente il valore di APL ed abbassare il CCP .

Capitolo 4 - Conclusioni

Il tentativo effettuato nel presente lavoro è stato quello di sviluppare metodi di DROP e REWIRE dello SLACER facendo evolvere la rete verso una stato di alta cooperazione. Logicamente lo stato di cooperazione o meno può essere visto come il prestare un servizio o meno alla rete P2P esistente, il grande successo di queste reti è infatti dato dell'essere decentralizzate rispetto ad un server o super-nodo: questa caratteristica fa sì che la rete si risulti più robusta e tollerante a possibili attacchi o anomalie che si potrebbero venire a creare.

Non è un caso che applicazioni, ormai largamente diffuse, quali BitTorrent o Skype fanno uno di questo paradigma per fornire servizi alla loro rete, immaginiamo quindi uno scenario futuro in cui vi sarà un esplosione di reti costruite sopra la rete Internet.