UNIVERSITA' DEGLI STUDI DI NAPOLI FEDERICO II

FACOLTA' DI INGEGNERIA

Corso di laurea in Ingegneria Elettronica

Elaborato di Laurea

SINCRONIZZAZIONE DI RETI COMPLESSE: CARATTERIZZAZIONE TOPOLOGICA E ROBUSTEZZA

RELATORE

CH.MO PROF.MASSIMILIANO DE MAGISTRIS

CORRELATORE

ING. PIETRO DE LELLIS

CANDIDATO

UGO MAISTO

MATR. 528/785

ANNO ACCADEMICO 2010-2011

Sincronizzazione di reti complesse: caratterizzazione topologica e robustezza

INDICE

Cap.1 Dinamiche complesse e circuito di Chua

1.1 Il circuito di Chua: breve storia pag. 1

1.2 Equazioni di stato del circuito di Chua 2

1.3 Implementazione con operazionali del circuito di Chua 4

1.4 Dinamiche e biforcazioni 10

Cap.2 Sincronizzazione e controllo di sistemi caotici

2.1 Sincronizzazione pag. 17

2.1.1 La Master Stability Function 17

2.1.2 Accoppiamento di due circuiti di Chua 22

2.1.3 Topologie regolari di N circuiti di Chua 27

2.1.4 Simulazioni con SwitcherCad 34

2.2 Controllo 37

2.2.1 sistemi drive-response 37

2.2.2 Il pinning 38

2.2.3 Collegamenti master-slave 41

2.2.4 Simulazioni con SwitcherCad 42

Cap.3 Caratterizzazione topologica di reti complesse

3.1 Introduzione pag. 45

3.2 Proprietà dei grafi 47

3.3 Topologia delle reti complesse 51

3.3.1 Reti regolari 51

3.3.2 Reti probabilistiche 52

3.4 Connettività delle reti complesse: robustezza e fragilità 58

3.5 Sincronizzazione di reti complesse 60

3.6 Robustezza e fragilità in reti scale-free 63

3.7 Controllo pinning di reti complesse 64

3.8 Analisi numerica della topologia delle reti complesse 70

3.8.1 Confronto tra varie topologie 70

3.8.2 Robustezza di reti scale-free e random all'attacco casuale di nodi o ad attacchi mirati 87

Appendice

Listati Matlab

Bibliografia

Cap. 1 Dinamiche complesse e circuito di Chua

1.1 Il circuito di Chua: breve storia

Durante una visita in Giappone nel 1983, Leon Chua, docente dell' università di Berkeley (California), realizzò un circuito elettronico in grado di produrre il caos.

La natura caotica di questo circuito fu osservata teoricamente per la prima volta da Matsumoto, che aveva ricevuto le istruzioni al riguardo dallo stesso Chua, nel 1983, e sperimentalmente da Zong e Ayrom nel 1984. In seguito tale circuito è stato oggetto di varie ricerche scientifiche ed attualmente è diventato un paradigma universale per il caos, dal momento che può generare diversi fenomeni caotici.[1]

È un circuito elettronico non lineare, costituito da quattro elementi lineari (due condensatori, un induttore ed una resistenza) ed un resistore non lineare, chiamato diodo di Chua. Soddisfa perciò i requisiti minimi e necessari, ma non sufficienti per presentare un comportamento caotico:

- almeno tre componenti dinamici (i due condensatori e l'induttore)
- almeno un componente non lineare
- almeno un componente attivo

Il diodo di Chua soddisfa le ultime due condizioni, poiché è un resistore non lineare, localmente attivo che si comporta come un generatore di corrente controllato in tensione (vedi figura 1.1a):

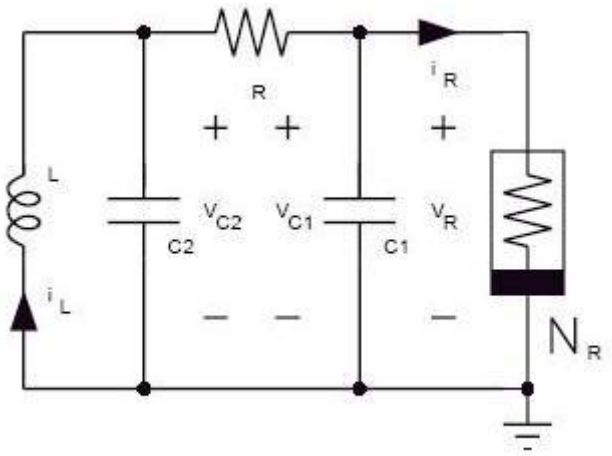


Figura 1.1a Schema elettrico del circuito di Chua

Il circuito di Chua è attualmente uno dei più significativi strumenti per studiare sperimentalmente i comportamenti oscillatori caotici, poiché è relativamente semplice da realizzare, ed attraverso le sue equazioni si può dare abbastanza agevolmente un fondamento teorico-matematico ai comportamenti caotici; inoltre tutti i suoi componenti, resistori, condensatori ed amplificatori operazionali, sono facilmente reperibili in commercio.

[1] Michael Peter Kennedy, *Three steps to chaos-Part II: A Chua's Circuit Primer,* IEEE Trans. Circuit & Systems, Fundamental Theory and Application, vol.40, n.10, oct.1993

Questo circuito presenta anche molte delle proprietà dei sistemi che si basano sul caos deterministico, ricordando che il caos si definisce come un comportamento non predicibile di un sistema dinamico deterministico a causa della sua sensibilità alle condizioni iniziali:

- È molto sensibile alle condizioni iniziali, cioè a variazioni infinitesime degli ingressi, corrispondono variazioni finite in uscita.
- Grazie al meccanismo di "stretching and folding", le traiettorie sull'attrattore restano confinate in una regione limitata dello spazio di fase, anche se, vicino all'origine, alcune di esse divergono esponenzialmente.
- Le traiettorie sono imprevedibili a lungo termine.

Possono essere implementate svariate applicazioni del suddetto circuito.
Se si aggiunge un resistore lineare in serie all'induttore il circuito diventa l'oscillatore di Chua (vedi figura 1.1b):

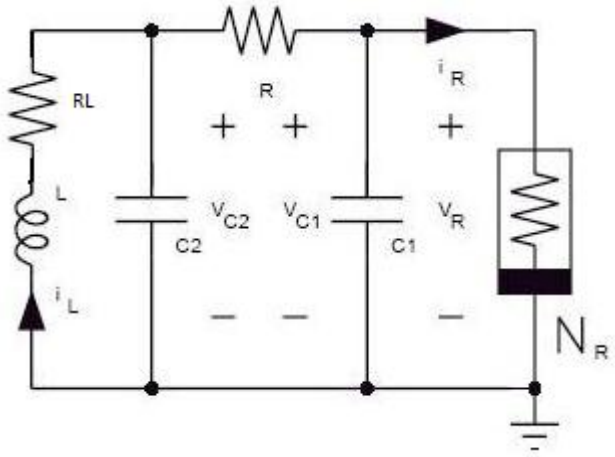


Figura 1.1b Schema elettrico dell'oscillatore di Chua

1.2 Equazioni di stato del circuito di Chua

Applicando le equazioni di Kirkhoff al circuito si ottene:

$$\frac{dv_{c1}}{dt} = -\frac{v_{c1}}{RC_1} + \frac{v_{c2}}{RC_1} - \frac{g(v_{c1})}{C_1}$$

$$\frac{dv_{c2}}{dt} = -\frac{v_{c1}}{RC_2} - \frac{v_{c2}}{RC_2} - \frac{i_L}{C_2}$$

$$\frac{di_L}{dt} = -\frac{v_{c2}}{L}$$

E ponendo $G = \frac{1}{R}$ si ha:

$$\frac{dv_{c1}}{dt} = -\frac{Gv_{c1}}{C_1} + \frac{Gv_{c2}}{C_1} - \frac{g(v_{c1})}{C_1}$$

$$\frac{dv_{c2}}{dt} = -\frac{Gv_{c1}}{C_2} - \frac{Gv_{c2}}{C_2} - \frac{i_L}{C_2}$$

$$\frac{di_L}{dt} = -\frac{v_{c2}}{L}$$

con

$$g(v_{c1}) = G_b v_{c1} + \frac{1}{2}(G_a - G_b)(|v_{c1} + E_o| - |v_{c1} - E_o|)$$

che è l'equazione caratteristica, non lineare, $v_{c1} - i$ del diodo di Chua, con una pendenza uguale a G_a nella regione interna e G_b nella regione esterna (vedi figura 1.6) [2]

- **A-dimensionalizzazione del circuito di Chua**

Per definizione, scalando opportunamente le equazioni del circuito di Chua:

$$x = \frac{v_{c1}}{E_o} \qquad y = \frac{v_{c2}}{E_o} \qquad z = \frac{i_L}{GE_o} \qquad \tau = \frac{Gt}{C_2}$$

$$a = \frac{G_a}{G} \qquad b = \frac{G_b}{G} \qquad \alpha = \frac{C_2}{C_1} \qquad \beta = \frac{C_2}{LG^2}$$

Le equazioni adimensionali del sistema, normalizzato nel tempo τ, sono:

$$\begin{cases} \dot{x}=\alpha(y-x-g(x)) \\ \dot{y}=x-y+z \\ \dot{z}=-\beta y \end{cases}$$

da cui, essendo:

$$g(x) = bx + \frac{1}{2}(a-b)[|x+1| - |x-1|]$$

ed in forma lineare a tratti:

$$g(x)= \begin{cases} bx+b-a & \text{se } x\leq -1 \\ ax & \text{se } |x|<1 \\ bx+a-b & \text{se } x\geq 1 \end{cases}$$

avendo posto:

$$\dot{x} = \frac{dx}{d\tau}, \qquad \dot{y} = \frac{dy}{d\tau}, \qquad \dot{z} = \frac{dz}{d\tau}$$

[2] Per i parametri E_o, G_a, G_b, vedi pag. 6

1.3 Implementazione con operazionali del circuito di Chua

- **Implementazione del diodo di Chua**

L'elemento non lineare del circuito di Chua si comporta come un resistore negativo asintoticamente passivo.
L'implementazione di una resistenza negativa può essere realizzata in vari modi. Un modo pratico[3] per realizzarla è utilizzare un amplificatore operazionale (OP AMP) e tre resistori (vedi figura 1.2):

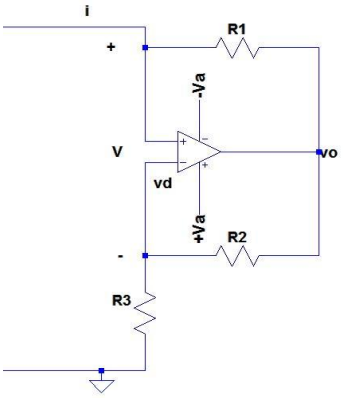


Figura 1.2 Dispositivo per implementare una resistenza negativa

Poiché a causa della resistenza di ingresso **infinita** dell'OP AMP, ritenuto ideale, non circola corrente nel terminale d'ingresso

$$i = \frac{v - v_o}{R_1}$$

per cui:

$$v = v_d + \frac{R_3}{R_2 + R_3} v_o$$

ed essendo $v_o = A_v v_d$:

$$v = \frac{R_2 + R_3(1 + A_v)}{A_v(R_2 + R_3)} v_o$$

[3] M.P. Kennedy, *Robust op-amp realization for Chua's circuits*. Frequents vol. 46, N°3-4 March-April 1992, pag. 66-80

ricavando v_o da v e sostituendolo nella prima equazione, si ha:

$$i = \frac{R_3 + R_2(1 - A_v)}{R_2 + R_3 + A_v R_3} v$$

per $A_v \to \infty$ ed $R_1 = R_2$:

$$i = -\frac{1}{R_1} v$$

che è la caratteristica a tre segmenti di una resistenza negativa a bassa tensione (vedi figura 1.3):

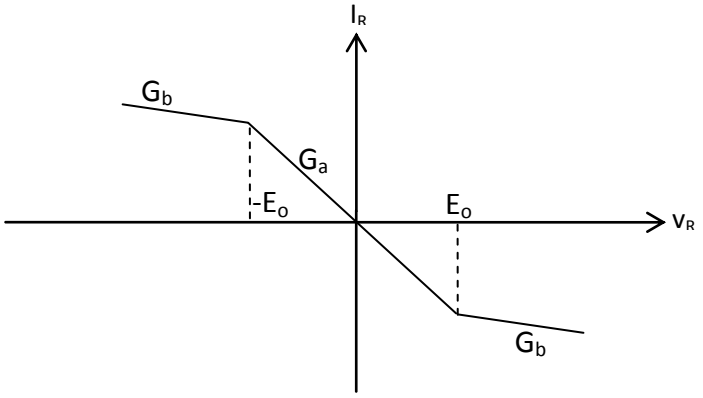


Figura 1.3 Caratteristica a tre segmenti del resistore non lineare

Questo dispositivo è asintoticamente passivo, cioè alle alte tensioni l'amplificatore operazionale satura, e di conseguenza la caratteristica sarà traslata rispetto all'origine e presenterà due pendenze positive (vedi figura 1.4):

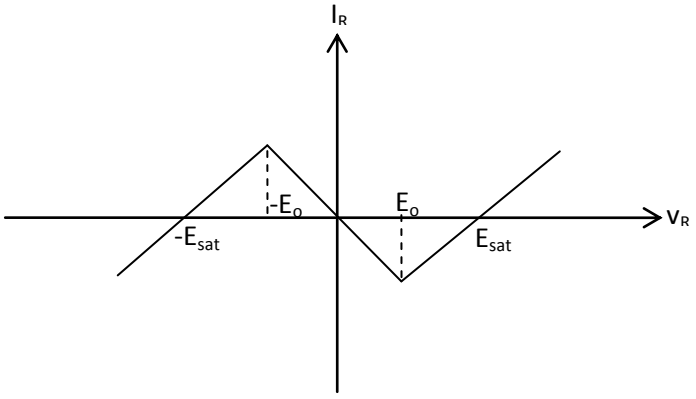


Figura 1.4 Caratteristica a tre segmenti del resistore non lineare con pendenza positiva

Sostituendo $v=E_o$ e $V_o=E_{sat}$ nelle equazioni precedenti, otteniamo:

$$i = \frac{v - E_{sat}}{R_1}$$

$$E_o = \frac{R_2 + R_3(1 + A_v)}{A_v(R_2 + R_3)} E_{sat}$$

che per gli alti guadagni diventa:

$$E_o = \frac{R_3}{R_2 + R_3} E_{sat}$$

Lo stesso risultato si ottiene per $v_o=-E_{sat}$.

se colleghiamo tra loro due resistori non lineari a tre segmenti avremo il diodo di Chua (figura 1.5):

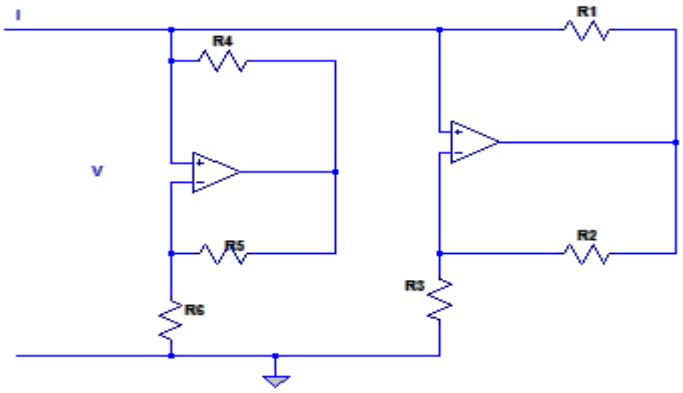


Figura 1.5 Implementazione con due OP AMP e resistenze del diodo di Chua

Poiché i due elementi sono controllati dalla stessa tensione e sono in parallelo, le loro caratteristiche v-i si possono sommare ottenendo così la caratteristica a cinque segmenti (vedi figura 1.6):

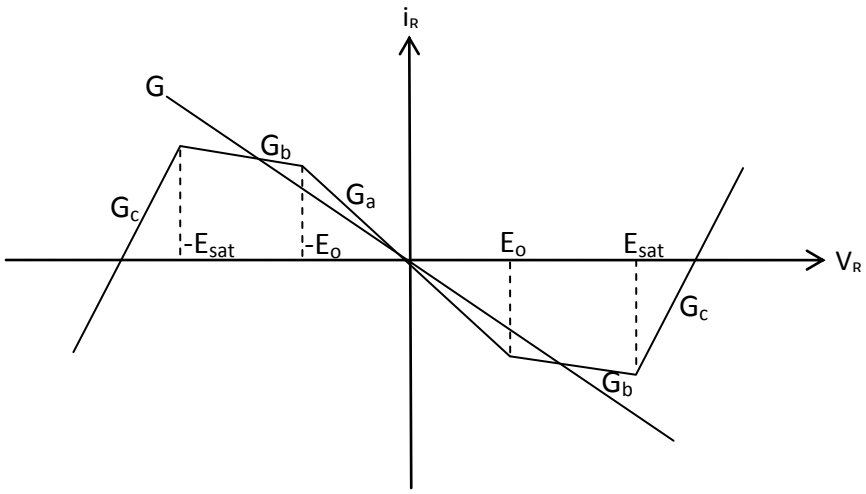


Figura 1.6 Caratteristica a cinque segmenti del diodo di Chua ottenuta sommando due caratteristiche a tre segmenti

i cui parametri sono:

$$G_a = -\frac{1}{R_3} - \frac{1}{R_6}$$

$$G_b = \frac{1}{R_4} - \frac{1}{R_3}$$

$$G_c = \frac{1}{R_1} + \frac{1}{R_4}$$

Consideriamo i breakpoints del primo e del secondo circuito, E_1 ed E_2 , e supponendo per ipotesi $E_1 < E_2$, si ha che :

$$E_1 = \frac{R_3}{R_2 + R_3} E_{sat}$$

$$E_2 = \frac{R_6}{R_5 + R_6} E_{sat}$$

Si ricava da queste formule che i parametri G_a G_b G_c E_1 E_2 possono essere facilmente modificati facendo variare le resistenze del diodo di Chua.

La caratteristica del diodo di Chua è:

$$g(V_{c1}) = \begin{cases} G_a v_{c1} & \text{se } |v_{c1}| \leq E_o \\ G_b v_{c1} + (G_b - G_a) E_o & \text{se } v_{c1} \leq -E_o \\ G_b v_{c1} + (G_a - G_b) E_o & \text{se } v_{c1} \geq E_o \end{cases}$$

Il circuito di Chua può essere allora descritto in forma lineare a tratti, poiché, come già visto nei paragrafi precedenti:

$$\frac{dv_{c1}}{dt} = -\frac{Gv_{c1}}{C_1} + \frac{Gv_{c2}}{C_1} - \frac{g(v_{c1})}{C_1}$$

$$\frac{dv_{C2}}{dt} = -\frac{Gv_{c1}}{C_2} - \frac{Gv_{c2}}{C_2} - \frac{i_L}{C_2}$$

$$\frac{di_L}{dt} = -\frac{v_{c2}}{L}$$

La prima equazione diventa, ponendo $G'_a=G+G_a$, $G'_b=G+G_b$:

$$\begin{cases} \dfrac{dv_{c1}}{dt} = \dfrac{-G'_b}{C_1} + \dfrac{Gv_{c2}}{C_1} - \dfrac{(G_b - G_a)E_o}{C_1} & se\ v_{c1} \leq -E_o \\ \dfrac{dv_{c1}}{dt} = -\dfrac{G'_a}{C_1} + \dfrac{Gv_{c2}}{C_1} & se\ |v_{c1}| \leq E_o \\ \dfrac{dv_{c1}}{dt} = \dfrac{-G'_b}{C_1} + \dfrac{Gv_{c2}}{C_1} - \dfrac{(G_a - G_b)E_o}{C_1} & se\ v_{c1} \geq E_o \end{cases}$$

- **Implementazione dell'induttore equivalente**

Nel circuito di Chua è preferibile utilizzare al posto dell'induttore reale un induttore equivalente composto da due OP AMP, quattro resistenze ed un condensatore (vedi figura 1.7) al fine di controllare più facilmente i parametri parassiti, (non dovendo tenere conto delle tolleranze dell'induttore reale), per poter misurare direttamente come tensioni le variabili di stato V_{C1}, V_{C2}, e I_L, l'ultimo parametro moltiplicato per un'appropriata resistenza. Inoltre si possono modificare i parametri dinamici semplicemente facendo variare solamente le resistenze del circuito, lavorando a basse tensioni.

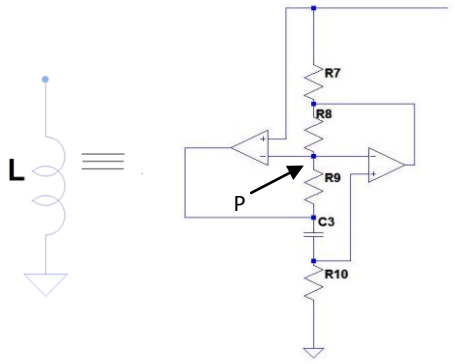


Figura 1.7 Implementazione dell'induttore equivalente realizzato per la prima volta da Tõrres e Aguirre

Tõrres e Aguirre[4] avevano rilevato che il circuito di Chua presenta uno spettro a banda larga, ma la maggior parte della densità di potenza è concentrata attorno alla frequenza determinata dal circuito risonante L-C_2 (vedi figura 1.1a). Quando si utilizza il computer per misurare i dati del dispositivo, calcolare espressioni complesse basate su questi dati, esercitare azioni di controllo o eseguire un output grafico, sono desiderabili oscillazioni caotiche molto lente (vedi figura 1.8). Per rallentarle è necessario utilizzare elementi di memoria a maggiore energia, per esempio valori più elevati di C_1, C_2, L. Mentre è facile trovare condensatori con valori elevati in un ampio range, lo stesso non si può dire per i valori degli induttori commerciali, che, quando hanno un elevato valore di induttanza, sono molto grandi e poco precisi. L'induttore equivalente proposto da Tõrres e Aguirre,

[4] L.A.B. Tõrres and L.A. Aguirre, *Inductorless Chua's circuit,* Electonics Letters, 9th November, vol. 36, N°23.

può essere considerato come il primo passo per integrare il circuito di Chua in un singolo chip, con un basso grado di disadattamento dei parametri e nessun componente esterno.[5]

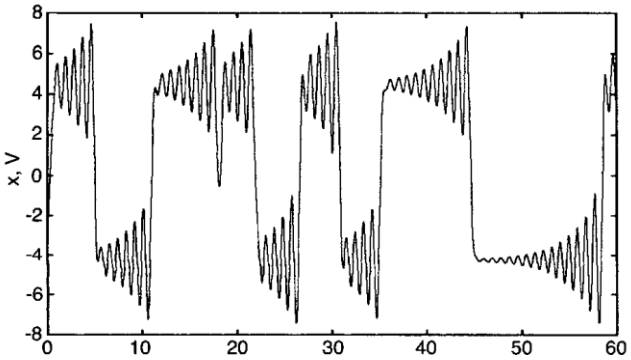


Figura 1.8 Tensione misurata attraverso il condensatore C_1 che mostra oscillazioni caotiche molto lente

Sono evidenti i vantaggi del circuito "inductorless". La terza variabile di stato z, che è una corrente attraverso l'induttore L mostrato in figura 1.7, può essere determinata misurando la tensione V_p del nodo P (vedi Figura 1.4.6.a):

$$z = \frac{(V_P - y)}{R_7}$$

dove y è la variabile di stato, corrispondente alla tensione, attraverso il condensatore C_2 e l'induttore L, nella figura 1.1.a e R_7 è il valore del resistore mostrato in figura 1.7. In questo modo si può ottenere il completo vettore di stato solo misurando le tensioni. Inoltre l'induttore equivalente esibisce una resistenza interna praticamente nulla, come verificato in pratica ed evidenziato dalle simulazioni , anche se si considerino non ideali gli amplificatori operazionali.

Il range delle frequenze, sopra le quali il circuito approssima accuratamente un induttore reale, si estende oltre quattro decadi, la qual cosa ne raccomanda l'utilizzo nel circuito di Chua. In conclusione, l'implementazione "inductorless" di tale circuito ne facilita la conoscenza delle tre variabili di stato e ne aumenta la flessibilità. Tõrres e Aguirre[6] suggeriscono una soluzione analoga anche per i condensatori.

L'induttanza equivalente L è:

$$L = \frac{R_7 R_9 R_{10}}{R_8} C_3$$

[5] Figura tratta da L.A.B. Tõrres and L.A. Aguirre, *op. cit.* pag. 1916
[6] L.A.B. Tõrres and L.A. Aguirre,*op.cit.*,pag1916

- **Implementazione dell'intero circuito di Chua**

Passiamo adesso all'implementazione dell'intero circuito di Chua (figura 1.9), utilizzando il diodo di figura 1.5 e l'induttore equivalente di figura 1.7:

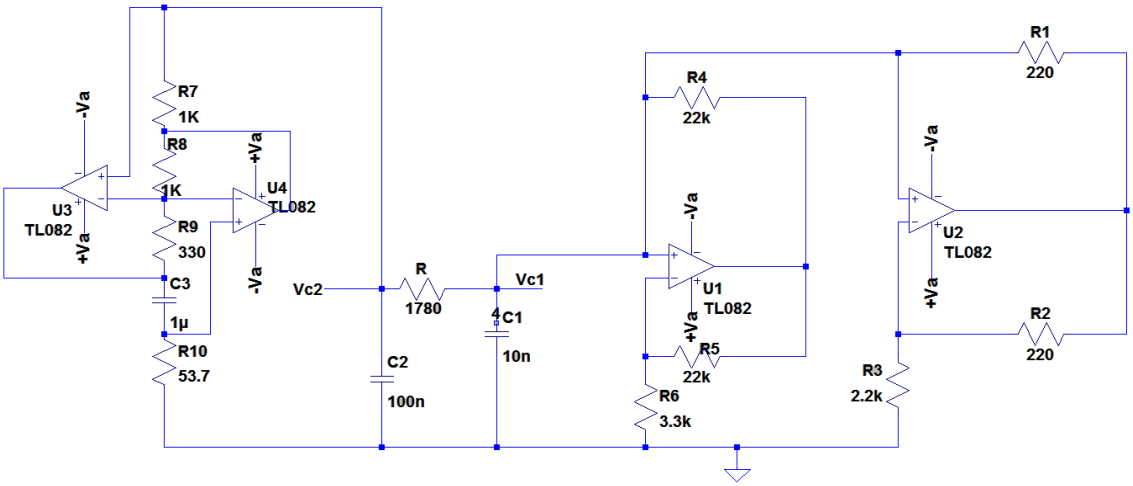


Figura 1.9 Implementazione dell'intero circuito di Chua

Abbiamo fissato i parametri del circuito di Chua e le condizioni iniziali in modo che il circuito presenti l'attrattore double scroll:

C_1=10nF, C_2=100nF, L≃18mH, R=1.78KΩ
G_a=-0.756mS, G_b=-0.409mS, E=1V
I parametri adimensionali (vedi paragrafo 1.3) risultano in questo caso:
α=10, β=17.6, a=-1.32, b=-0.728
Il parametro di biforcazione è la resistenza R.
Nel paragrafo successivo vedremo come variano le traiettorie al variare della resistenza R dal valore di 2050Ω fino a quello di 1780Ω, utilizzando una simulazione Switchercad, prima nel piano delle fasi e poi nello spazio di stato.

1.4 Dinamiche e biforcazioni del circuito di Chua

I punti di equilibrio del circuito di Chua sono i punti di intersezione della caratteristica del diodo di Chua con la retta $G = -\frac{1}{R}$ (vedi figura 1.6):

Al variare di R, variano i punti di equilibrio delle regioni esterne.

Per R sufficientemente grande (ad esempio 2050Ω), i punti di equilibrio delle regioni esterne sono stabili, mentre l'origine è un punto instabile.

Il sistema si porterà con una traiettoria a spirale, a seconda delle condizioni iniziali, in uno dei due punti di equilibrio stabile per rimanervi indefinitivamente (vedi figure 1.10):

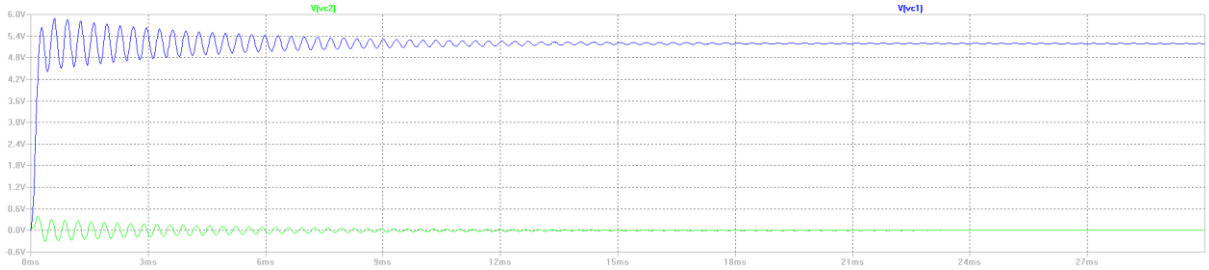


Figura 1.10a Simulazione spice, soluzione stazionaria stabile per R=2050Ω

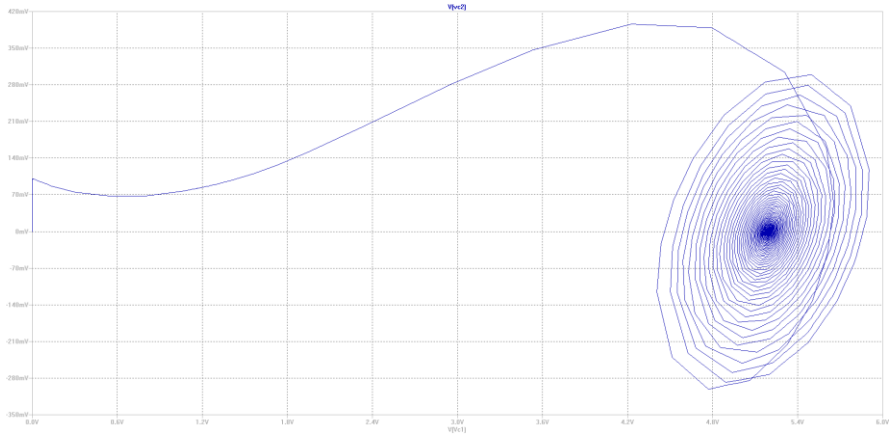


Figura 1.10b Simulazione spice nello spazio di stato per R=2050Ω

Se diminuisce R, il numero di oscillazioni cresce, finché si arriva ad una soluzione periodica di periodo T intorno al precedente punto di equilibrio (vedi figura 1.11):

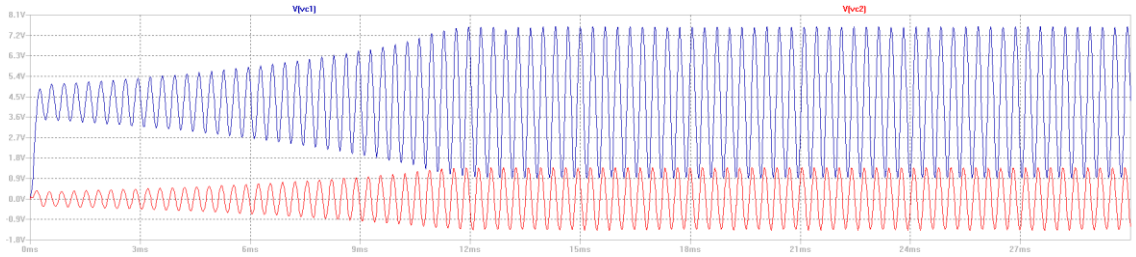


Figura 1.11 Simulazione spice, soluzione periodica stabile di periodo T per R=1980Ω

Per valori ancora più bassi della resistenza R, le orbite avranno periodo 2T, 4T……. (figura 1.12):

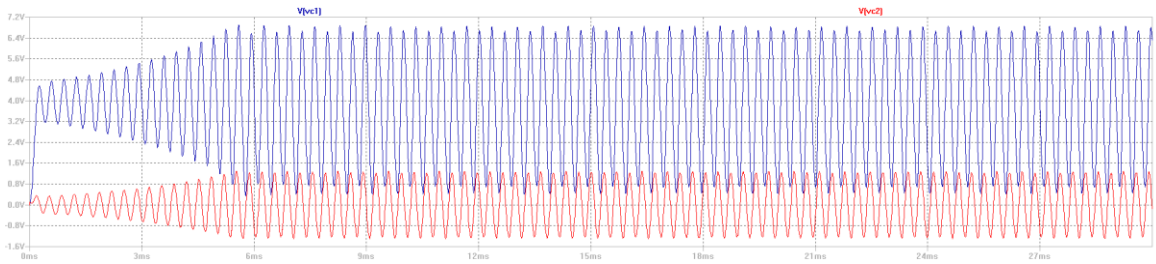


Figura 1.12a Simulazione spice, soluzione periodica stabile di periodo 2T per R=1950Ω

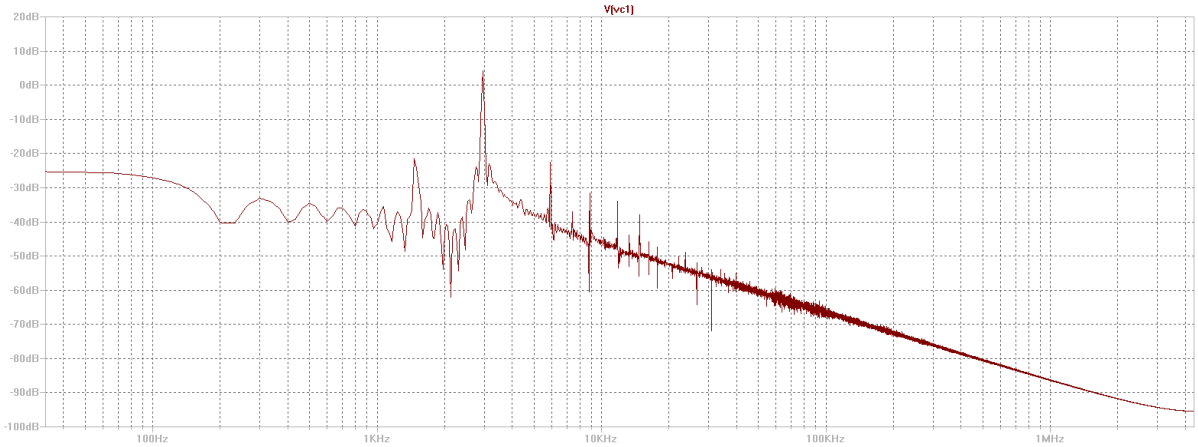


Figura 1.12b Trasformata di Fourier per R=1950Ω .Lo spettro in frequenza evidenzia le subarmoniche

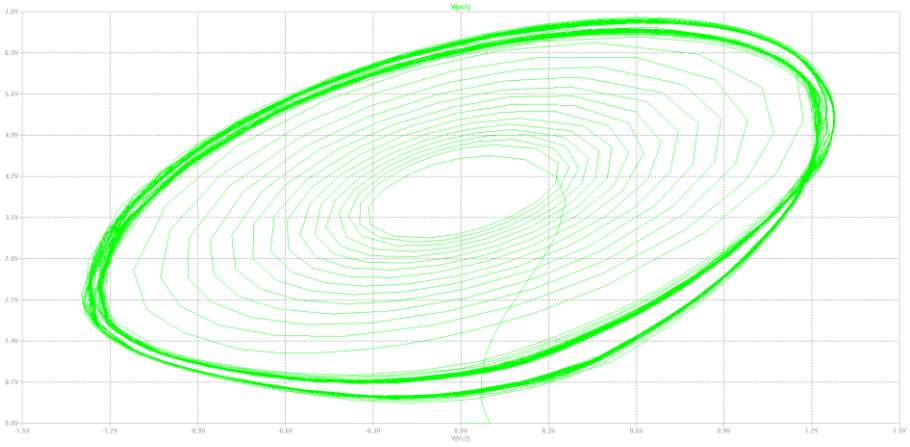


Figura 1.12c Simulazione spice, nello spazio di stato, per R=1950Ω

Con un ulteriore lieve diminuzione della resistenza si osservano raddoppiamenti del periodo (vedi figura 1.13):

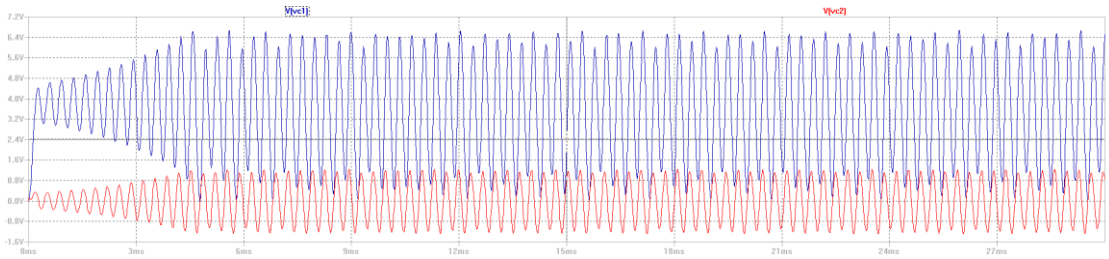


Figura 1.13a Simulazione spice: soluzione periodica stabile di periodo 4T, per R=1938Ω

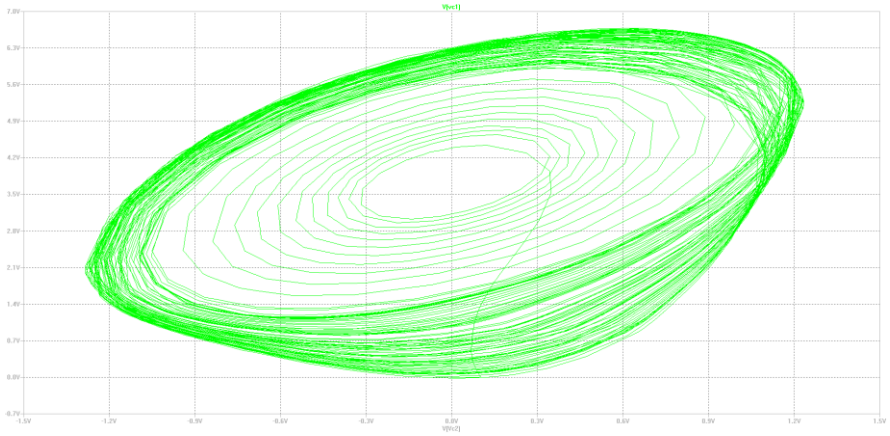


Figura 1.13b Simulazione spice: soluzione periodica stabile di periodo 4T, per R=1938Ω

Se si fa diminuire ancora la resistenza, i cicli limite assumono un periodo che tende a diventare infinito, i punti delle regioni esterne diventano instabili e di conseguenza si osserva un continuo cambio di regione da parte dell'orbita, che non può stare indefinitivamente in nessuna delle regioni. Il moto della traiettoria è apparentemente irregolare in una regione di tipo a spirale detta "attrattore di Chua" (vedi figura 1.14):

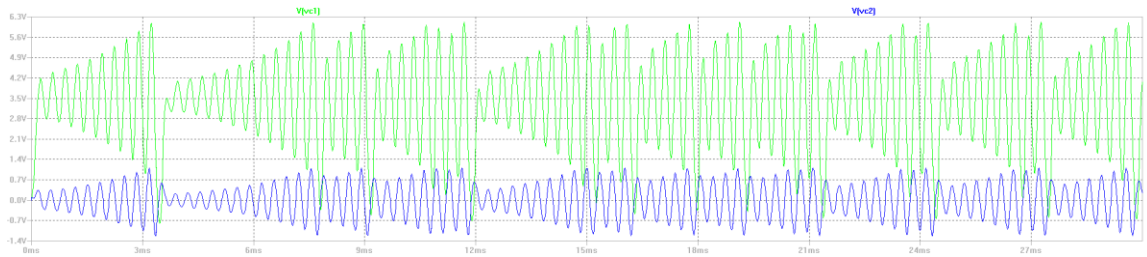


Figura 14a Traiettoria caotica, per R=1910Ω

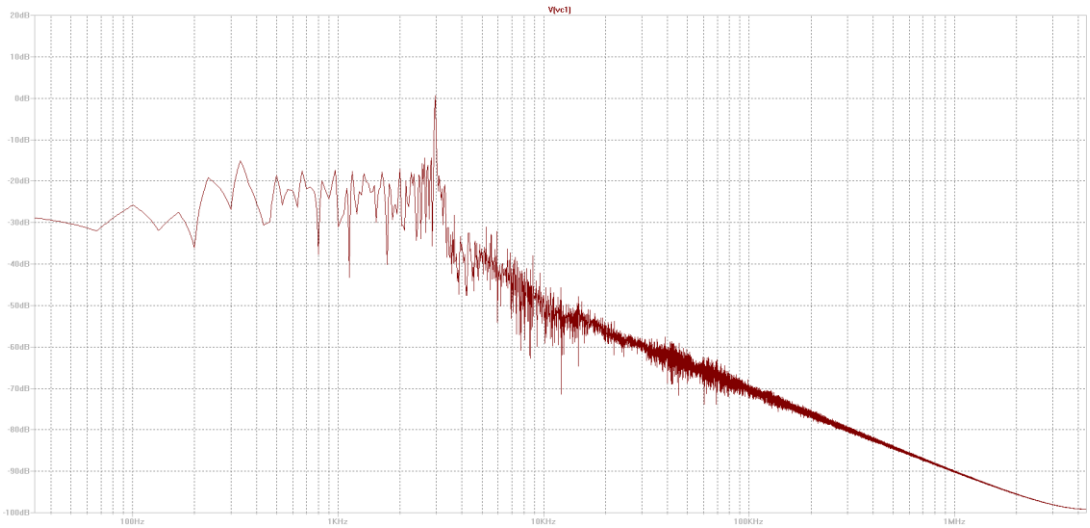


Figura 1.14b Trasformata di Fourier, per R=1910Ω

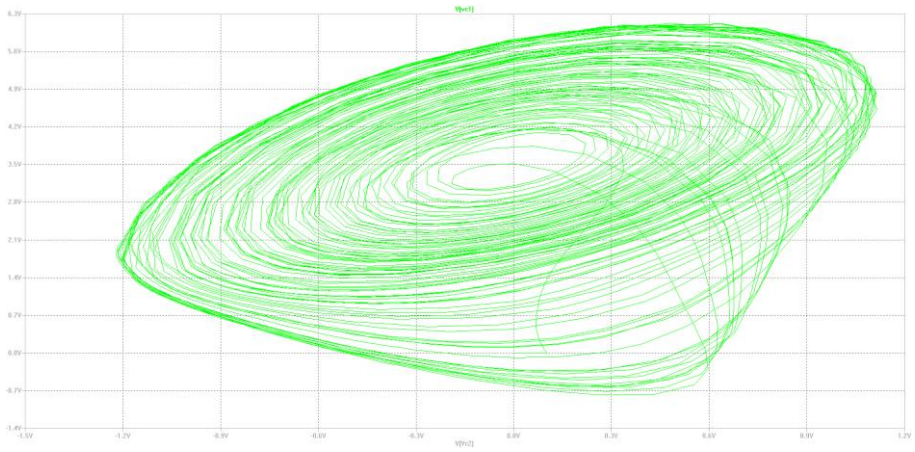


Figura 1.14c Simulazione nello spazio di stato, per R=1910Ω

Con ulteriori riduzioni del valore di R, compaiono due attrattori di questo tipo separati da una zona ambigua, in questo caso si parla di “attrattore double scroll“(vedi figura 1.15):

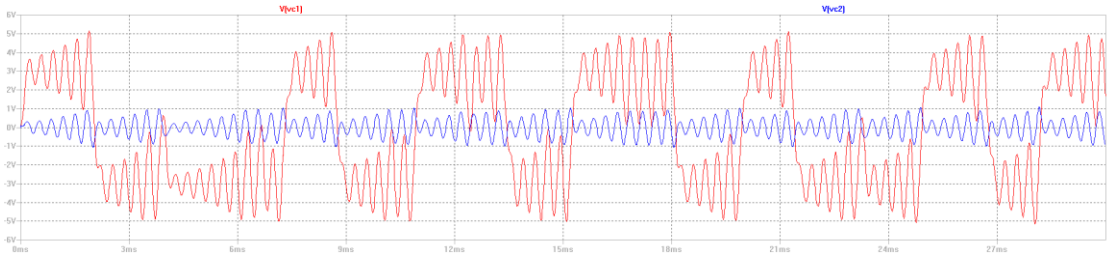


Figura 1.15a Traiettoria caotica di tipo double scroll, per R=1840Ω

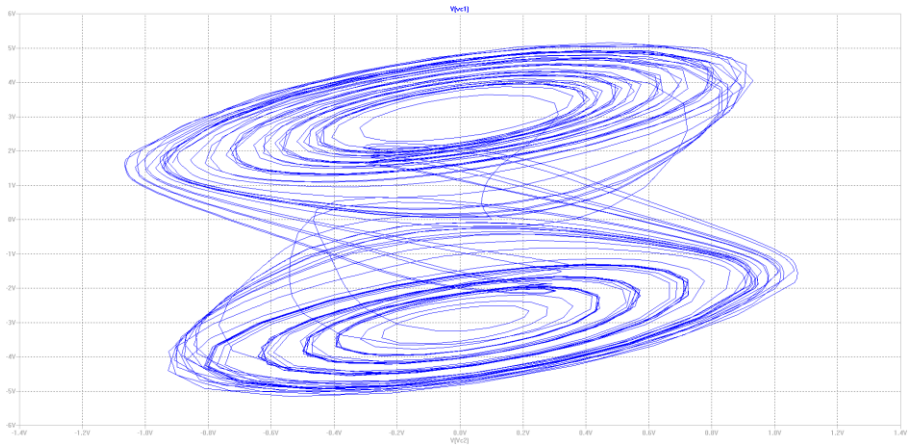


Figura 1.15b Simulazione nello spazio di stato, per R=1840Ω

Per valori di R ancora più bassi si ottiene la cosiddetta “finestra nel caos” tra i due attrattori (vedi Figura1.16 e 1.17):

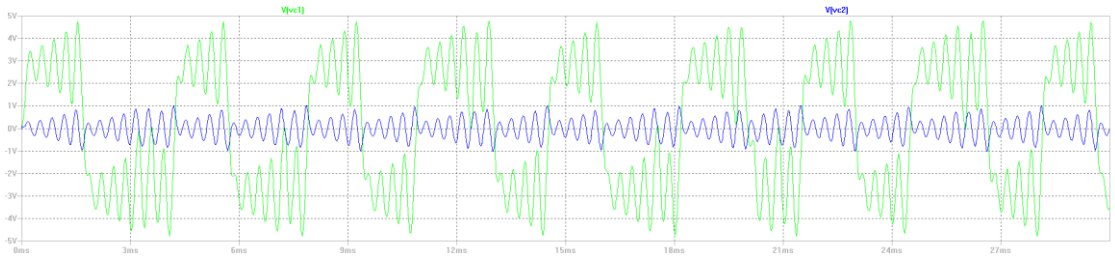


Figura 1.16a Traiettoria di tipo finestra nel caos, per R=1810Ω

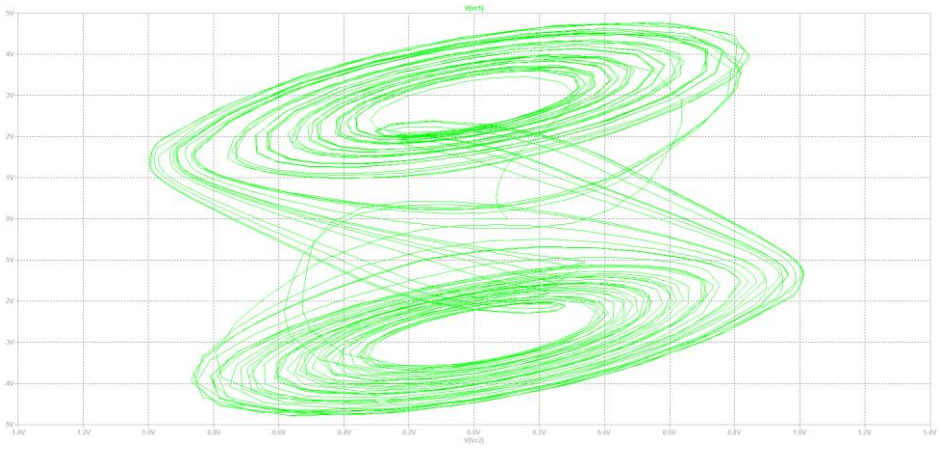


Figura 1.16b Simulazione nello spazio di stato, per R=1810Ω

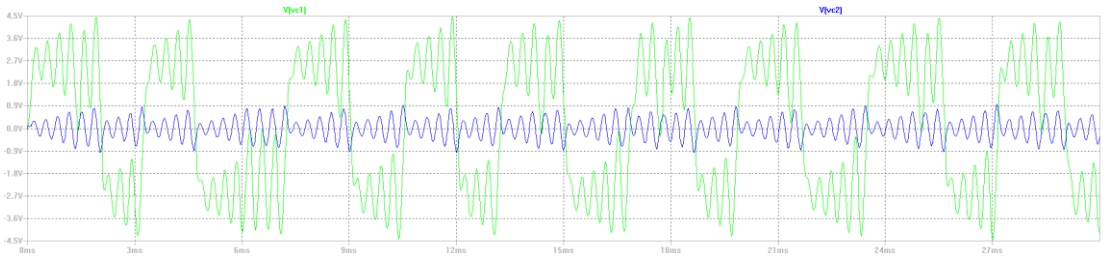


Figura 1.17a Simulazione nel piano delle fasi, per R=1780Ω

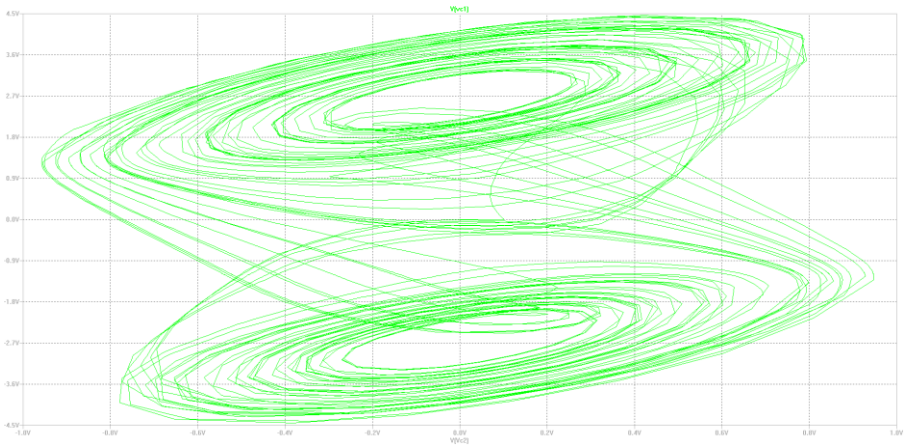


Figura 1.17b Simulazione nello spazio di stato, per R= 1780Ω

Se la resistenza R continua a diminuire, si giunge al limite critico, oltre il quale una ulteriore diminuzione dei valori di R provocherebbe un'instabilità generale del sistema e porterebbe l'orbita ad una divergenza a spirale verso l'infinito, ma a tale valore non è possibile arrivare con un circuito di Chua reale, quindi questo risultato si utilizza come un ipotetico limite inferiore per la resistenza R.

Cap. 2 Sincronizzazione e controllo di sistemi caotici

2.1 Sincronizzazione

La proprietà essenziale di un segnale caotico[7] è di non essere asintoticamente stabile, di avere cioè traiettorie strettamente dipendenti dalle condizioni iniziali, e che rapidamente diventano incorrelate; recentemente è stato dimostrato che è possibile costruire una rete di circuiti caotici tali che le loro traiettorie abbiano un comportamento identico o "sincronizzato". Quando si ha a che fare con un sistema caotico si può avvertire l'esigenza di eliminare completamente o parzialmente il comportamento caotico, oppure scegliere in quale attrattore si vuole che il sistema funzioni. Si ha la sincronizzazione di due sistemi quando le traiettorie di uno dei due sistemi convergono l'una agli stessi valori dell'altra, ed entrambe rimangono al passo tra di loro. Poiché i sistemi dinamici caotici sono molto sensibili alle condizioni iniziali, anche se presentano gli stessi attrattori nello spazio di fase, le rispettive traiettorie diventano rapidamente non correlate, e quindi è molto difficile costruire in laboratorio dei sistemi sincronizzati; per questo motivo sta assumendo una grande importanza lo studio delle tecniche di sincronizzazione di due o più circuiti. Pecora e Carroll hanno dimostrato che condizione necessaria e sufficiente affinché una traiettoria caotica w(t) sia asintoticamente stabile, è che gli esponenti condizionali di Lyapunov, (usualmente chiamati CLE), siano a parte reale negativa (vedi pag.18 e seguenti). La sincronizzazione in questo caso è quindi indipendente dalle condizioni iniziali dei due sistemi.

Le tecniche di sincronizzazione si dividono in due classi:
1)accoppiamento direzionale (o drive-response), quando il circuito driver controlla il circuito slave (o response).
2)accoppiamento non direzionale quando entrambi i circuiti, connessi tra loro, si influenzano a vicenda.

Vediamo come possiamo applicare queste tecniche a due o a più circuiti di Chua.

2.1.1 La Master Stability Function

La sincronizzazione nei sistemi dinamici complessi è stato un elemento di continuo interesse. Uno strumento basilare nell'analizzare i vari problemi di sincronizzazione è la Master Stability Function (MSF) [8]. La sincronizzazione di un sistema dinamico complesso, consistente in un numero di oscillatori accoppiati identici, si raggiunge dal momento in cui le traiettorie di tutti gli oscillatori si avvicinano l'una all'altra asintoticamente.

In assenza di collegamento i singoli circuiti si comporterebbero indipendentemente, la dimensione del sottospazio nel quale si trova la soluzione sincrona è necessariamente più piccola di tutto lo spazio di fase, matematicamente il sottospazio sincrono è chiamato collettore di sincronizzazione. Se il collettore è stabile rispetto alle perturbazioni, nel sottospazio complementare o nel sottospa-

[7] Cfr. Leon Chua, Makoto Itoh, Ljupco Kocarev e Kevin Eckert , *Chaos Synchronization in Chua's circuit*, Journal of circuits, Systems and Computers, Vol. 3 N° 1(1993)Pag. 93 -108
[8]Cfr. Huang, Chen, Lai, Pecora, *Generic behavior of master-stability functions in coupled non linear dynamical system*, Phisical review E 80,036204, pag.1 e sgg.

zio trasversale, la sincronizzazione può essere osservata in un ambiente fisico; la MFS misura il tasso di velocità esponenziale alla quale un'infinitesima perturbazione cresce nello spazio trasversale. Nella terminologia dei sistemi dinamici, MSF è il più grande esponente trasverso di Lyapunov del sottospazio di sincronizzazione. Una condizione necessaria per la sincronizzazione è che la MSF sia negativa e che i corrispondenti parametri normalizzati di accoppiamento (determinati da un sistema di oscillazione accoppiata) cadano nella regione negativa dell' MSF.

Nel 2002 il formalismo MSF fu utilizzato per sincronizzare reti formate da pochi componenti; questo segna l'inizio di un'area di ricerca in rapida evoluzione, nell'ambito della scienza non lineare ed ingegneristica: la sincronizzazione nelle reti complesse. Il formalismo MSF è richiesto perché permette che le proprietà dei singoli oscillatori siano separate dalla matrice di accoppiamento A, che caratterizza la topologia della rete. In particolare, se K è un parametro normalizzato di accoppiamento, la MSF può essere calcolata come una funzione di K, basata solo sulla conoscenza delle dinamiche dei singoli accoppiatori e sulla funzione di accoppiamento. Cioè, la MSF può essere ottenuta indipendentemente dalla topologia della rete collegata, che supporta un grande numero di tali oscillatori. Una tacita assunzione nella letteratura della sincronizzazione delle reti è che la SMF sia negativa in alcuni intervalli (K_a, K_b), dove $K_a<K_b$. La rete è sincronizzabile se tutti gli elementi diversi da zero $K_i=c\lambda_i$ (i=2, …. ,N) cadono in detto intervallo, dove c è un parametro di accoppiamento che caratterizza l'interazione media tra gli oscillatori, λ_i sono gli autovalori della matrice accoppiata, N è il numero di oscillatori nella rete, e si assume $\lambda_1=0$. Dato l'intervallo (K_a, K_b) una rete di una specifica topologia può essere più facilmente sincronizzata, se l'estensione dello spettro di un autovalore minore di zero è più piccolo.

Ci chiediamo se la proprietà dell'esistenza di intervalli dei valori negativi dell'MSF attenga ai tipici oscillatori non lineari. In generale, questo problema è difficile da risolvere poiché esiste un infinito numero di possibilità per gli oscillatori dinamici.

In questo paragrafo noi esamineremo sistematicamente una rete di i cui nodi sono circuiti di Chua. Vari calcoli ed analisi (vedi gli studi di Chuang, Chen, Cheng Lai e Pecora) portano a credere fortemente che vi sia una generica proprietà del'MSF a essere negativa in un intervallo finito di parametri, il risultato è che per ognuno di questi oscillatori esiste sempre in questo intervallo uno schema di accoppiamento per il quale si può realizzare la sincronizzazione della rete di un gran numero di tali circuiti

Analisi di sincronizzazione con la MSF

Consideriamo un sistema dinamico complesso, costituito da circuiti di Chua tutti collegati, per i quali, a causa dell'intrinseca non linearità delle connessioni , i comportamenti caotici sono comuni. Ogni circuito isolato è descritto da

$$\dot{x} = F(x) \qquad (1)$$

Per ottenere una sincronizzazione caotica scegliamo i parametri di ogni circuito, in modo tale che questo risulti un attrattore caotico. La tipica equazione per le dinamiche di una rete di N oscillatori accoppiati è:

$$\dot{x}_i = F(x_i) - c\sum_{j=1}^{N} a_{ij}\, H(x_j) \qquad x_i \in R^N \qquad (2)$$

dove $H(x) \in R^{NXN}$ è la funzione che definisce quale variabile di stato è stata usata per l'accoppiamento (tensione sul primo o sul secondo condensatore o sull'induttore, quest'ultimo legame non conveniente come abbiamo già visto). Più dettagliatamente:

$H(x)=\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix}$, se il collegamento avviene attraverso le tre variabili di stato

$H(x)=\begin{bmatrix}1 & 0 & 0\\ 0 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}$, se il collegamento avviene attraverso la x (ovvero la tensione sul condensatore $V(c_1)$,

$H(x)=\begin{bmatrix}0 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 0\end{bmatrix}$, se il collegamento avviene attraverso la y, ovvero la tensione sul condensatore $V(c_2)$,

$H(x)=\begin{bmatrix}0 & 0 & 0\\ 0 & 0 & 0\\ 0 & 0 & 1\end{bmatrix}$, se il collegamento avviene attraverso la z, ovvero la tensione sull'induttore.

Come abbiamo visto nei paragrafi precedenti, $c>0$ è il parametro globale che rappresenta l'intensità dell'accoppiamento, A è la matrice di connessione, determinata dalla particolare topologia della rete ed N è il numero di circuiti che compongono la rete.
La matrice A soddisfa la condizione:

$$\sum_{j=1}^{N} a_{ij} = 0 \quad \forall\, i \tag{3}$$

Di conseguenza lo stato sincrono $x_1=x_2=\ldots=x_n=s$, dove $\frac{ds}{dt} = F(x)$, è una soluzione dell'equazione (2). D'altra parte, noi assumiamo che la matrice A di collegamento può essere diagonalizzata con un set di autovalori reali (λ_i, i=1,…,N) ed i loro rispettivi autovettori($v_1,v_2\ldots v_N$).La rete è connessa in modo che vi sia un solo autovalore uguale a 0 e gli altri autovalori siano del tipo $\lambda_1=0>\lambda_2>\lambda_3>\ldots\lambda_N$. Per il sistema descritto dall'equazione (2) l'equazione differenziale:

$$\frac{d\delta x_i}{dt} = J(s)\cdot\delta x_i - c\sum_{j=1}^{N} a_{ij}\, H(s)\cdot\delta x_j \tag{4}$$

Dove $\delta x_i=x_i-s$ e J è lo jacobiano della funzione F .

A questo punto, potendo diagonalizzare la matrice A con un insieme di autovalori reali ($\lambda_i=0$, i= 1,2, ….. ,N) e con i rispettivi autovettori(e_i, i=1,2, … ,N) si può porre $\delta y=Q^{-1}\delta x$, dove Q è una matrice che ha per colonna gli autovettori di A, si ottiene quindi:

$$\frac{d\delta y}{dt} = [J(s) - c\lambda_i H(s)]\cdot\delta y_i \tag{5}$$

ponendo $K_i = c\lambda_i$, l'equazione di ogni circuito è uguale, escluso il parametro K_i, per cui l'equazione si può riscrivere come:

$$\frac{d\delta y}{dt} = [J(s) - kH(s)]\cdot\delta y_i \qquad (6)$$

Dall'ultima equazione si possono ricavare gli esponenti di Lyapunov, il più grande dei quali è la Master Stability Function $\psi(k)$. Se $\psi(k)$ è negativo, anche un piccolo disturbo dello stato di sincronizzazione decrescerà asintoticamente a zero e sarà mantenuta la sincronizzazione, o al limite nelle vicinanze.
La soluzione sincrona è instabile e non può essere realizzata fisicamente se $\psi(k)$ è positiva, poiché eventi di disturbo si amplificheranno facendo perdere del tutto la sincronizzazione.
Se si trova il valore $\bar{k}$ per cui $\psi(\bar{k})=0$ e ordinando gli autovalori come
$0= \lambda_1>\lambda_2\geq\lambda_3\geq \ldots\ldots \geq\lambda_N$ si avrà sincronizzazione se $c \geq \left|\frac{\bar{k}}{\lambda_2}\right|$.
Una soluzione per calcolare la MSF nel risolvere in contemporanea l'equazione (1) per trovare s e l'equazione (6) usando come condizioni iniziali di questa $\delta y(0)=I(N)$ dove I è la matrice identità di ordine N.
Per calcolare l'MSF in un circuito di Chua riscriviamo le equazioni di detto circuito (vedi pag. 3):

$$\dot{x}=\alpha(y-x+g(x))$$
$$\dot{y}=x-y+z$$
$$\dot{z}=-\beta yd$$

le equazioni del diodo di Chua sono:

$$g(x)= \begin{cases} -bx-b+a & \text{se } x<-1 \\ ax & \text{se} |x|<1 \\ -bx-a+b & \text{se } x>1 \end{cases}$$

Per calcolare la MSF possiamo considerare una versione più semplice del diodo di Chua, cioè una versione con solo tre tratti, poiché il diodo nelle nostre simulazioni non lavorerà mai nelle fasce esterne.
Scegliendo $H(x)=[0]$ in modo da utilizzare solo la prima variabile di stato per l'accoppiamento, avremo la seguente matrice jacobiana di F(s):

$$J(s) = \begin{pmatrix} -\alpha - \alpha\cdot\begin{cases} a & \text{se } |x| \leq 1 \\ b & \text{se } |x| > 1 \end{cases} & \alpha & 0 \\ 1 & -1 & 1 \\ 0 & -\beta & -\gamma \end{pmatrix}$$

La MSF sarà allora:

$$\frac{\delta dy}{dt} = J(s)\delta y - k\delta y_1$$

Tale funzione può essere tranquillamente integrata anche nei suoi punti di discontinuità, poiché sia nel punto 1 che nel punto -1 , $\delta x^+= \delta x^-$.

Per collegamenti con la variabile di stato x, la MSF ha un valore decrescente, per cui aumentando K aumenta la sincronizzazione tra i Chua. Poiché per ψ(k)=0 , $k=6.31$ (valore trovato per parametri misurati sperimentalmente)[9], questo è il valore limite per ottenere la sincronizzazione tra i Chua. Con un valore maggiore del valore limite, si ha una migliore sincronizzazione poiché gli esponenti di Lyapunov sono tutti negativi, con un valore minore del valore limite, c'è almeno un esponente di Lyapunov positivo, e questo porta alla desincronizzazione del sistema.
Tornando al modello fisico:

$$\bar{k} = \left|\frac{\lambda_2}{R_c}\frac{C_2 \quad R}{C_1}\right|$$

Vediamo adesso come la variazione dei parametri dei singoli circuiti Chua influiscono sulla sincronizzazione del sistema. La variazione del valore dell'induttanza L è praticamente insignificante ai fini della sincronizzazione, mentre un aumento di C_2 fa aumentare $\bar{K}$, e quindi migliora la sincronizzazione, viceversa con un aumento di C_1 si ha una diminuzione di $\bar{K}$ con relativo peggioramento della sincronizzazione. Per quel che riguarda i valori della resistenza R, la situazione è alquanto diversa, poiché non si può parlare di una relazione lineare tra R e $\psi(K)$, essendo l'orbita caotica del Chua molto sensibile a piccole variazioni di R.
La MSF calcolata come sopra, riguarda circuiti identici che compongono una rete, ipotesi molto difficile da verificarsi nella realtà, soprattutto per quel che riguarda i condensatori che difficilmente hanno tolleranze inferiori al 3%.

- **Analisi delle soglie di sincronizzazione per una rete di Chua**

I parametri nominali di ogni circuito sono, come già detto al capitolo primo:

$R=1780\Omega$, $L=18mH$, $C_1=10nF$, $C_2=100nF$, $G_a=-0.757mS$, $G_b=-0.409mS$

Da cui si ricavano i valori dei parametri adimensionali:

$$\alpha = \frac{C_2}{C_1} = 10\,, \qquad \beta = \frac{C_2 R^2}{L} = 17.6\,, \qquad a = G_a R = -1.34\ \ , b = G_b R = -0.72$$

Con il Matlab abbiamo trovato il grafico della MSF per questi valori nominali dei parametri dei Chua(vedi figura 2.1)

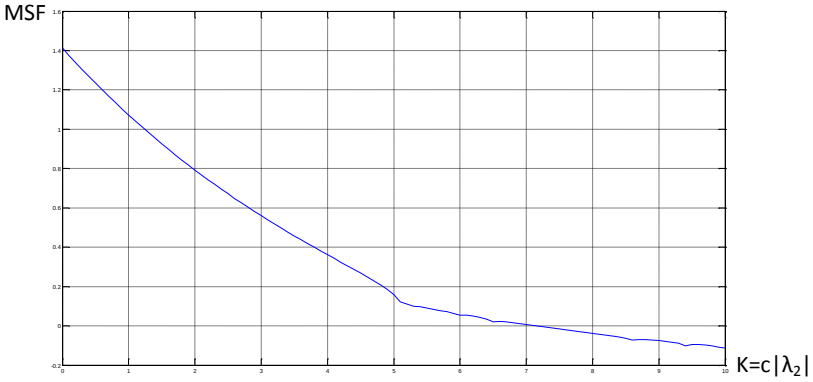


Figura 2.1 Simulazione Matlab della MSF di I tipo (i valori sono inizialmente positivi,per poi decrescere, fino a diventare negativi in corrispondenza di un certo valore di K).

[9] Vedi tesi: Dr. Maiorano Claudio, *Analisi sperimentale della sincronizzazione di circuiti di Chua: effetti di accoppiamenti dinamici e topologie con elevati numeri di nodi* ;Dr. Cesarino Claudia, *Analisi sperimentale della sincronizzazione dei circuiti di Chua*: *effetti topologici di link direzionali e non direzionali*

Per poter ottenere una buona sincronizzazione degli N Chua di una rete, dobbiamo scegliere K nella regione negativa della MSF,e perciò si avrà in questo caso un K>=7.01 , poiché per questi valori di K gli esponenti di Ljapunov avranno tutti valori negativi. Per valori minori del valore limite (K=7.01) ci sarà almeno un esponente di Ljapunov con parte reale positiva e ciò porterà ad una desincronizzazione del sistema.

Essendo :

$$\overline{K} = \left|\frac{\lambda_2 C_2 R}{R_c C_1}\right|$$

Anche la variazione dei parametri C_1, C_2 , R, R_c , come abbiamo già visto nel paragrafo precedente, facendo aumentare o diminuire K, ha influenza sulla sincronizzazione. Naturalmente questo non vale tanto per il parametro R, dal quale dipende il comportamento caotico. Per trovare il valore della resistenza di collegamento tra i circuiti del sistema, è necessario prima trovare λ_2 che è il massimo autovalore, diverso da zero, della matrice di collegamento A, la quale varia a seconda della topologia di connessione.

2.1.2 Accoppiamento di due circuiti di Chua

Consideriamo due sistemi caotici identici, di cui il secondo sia $x'(\tau)$, $y'(\tau)$, $z'(\tau)$, con le stesse equazioni di stato che abbiamo visto nel paragrafo precedente per il singolo circuito $x(\tau)$, $y(\tau)$, $z(\tau)$. Definiamo un sistema differenza $\xi(\tau)=p(\tau)$, $q(\tau)$, $r(\tau)$, dove:

$$p(\tau)=x(\tau)-x'(\tau)$$

$$q(\tau)=y(\tau)-y'(\tau)$$

$$r(\tau)=z(\tau)-z'(\tau)$$

Le equazioni adimensionali dei due circuiti saranno allora:

$$\begin{cases} \dot{x}=\alpha(y-x-g(x))+\delta_x(x'-x) \\ \dot{y}=x-y+z+\delta_y(y'-y) \\ \dot{z}=-\beta y+\delta_z(z'-z) \\ \dot{x}'=\alpha(y'-x'+g(x'))+\delta_x(x-x') \\ \dot{y}'=x'-y'+z'+\delta_y(y-y') \\ \dot{z}'=-\beta y'+\delta_z(z-z') \end{cases}$$

se i due circuiti sono collegati attraverso tutte e tre le variabili di stato.
Supponiamo che esistano valori (δ_1 , δ_2) tale che per $\delta_1 < \delta < \delta_2$, dove δ indica δ_x , δ_y o δ_z , la parte

reale degli autovalori delle due matrici jacobiane ricavate dalle equazioni precedenti[10] :

$$\begin{bmatrix} -\alpha - a\alpha - 2\delta_x & \alpha & 0 \\ 1 & -1 - 2\delta_y & 1 \\ 0 & -\beta & -2\delta_z \end{bmatrix}$$

$$\begin{bmatrix} -\alpha - b\alpha - 2\delta_x & \alpha & 0 \\ 1 & -1 - 2\delta_y & 1 \\ 0 & -\beta & -2\delta_z \end{bmatrix}$$

sia negativa. Allora il sottosistema $\{x,y,z,x',y',z': x=x',y=y',z=z'\}$ è stabile per ogni $\delta_1 < \delta < \delta_2$, nel senso che tutte le traiettorie nei due sistemi si avvicinano l'una all'altra asintoticamente, indipendentemente dalle condizioni iniziali, ciò significa che esse sono nella zona di attrazione. Conseguentemente abbiamo:

$$\dot{p}=\alpha q-\alpha p-\alpha[g(x)-g(x')]-2\delta_x p$$

$$\dot{q}=p-q+r-2\delta_y q$$

$$\dot{r}=-\beta q-2\delta_z r$$

dal momento che $g(x)-g(x')=g'(\eta)(x-x')$ e $g'(\eta)$ contiene i due valori a , b , detto sistema si riduce al sistema lineare:

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = A \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

Dove la matrice A è costituita dall'una o dall'altra tra le matrici appena menzionate. Se la parte reale di tutti gli autovalori di queste due matrici è negativa, ne consegue che l'equilibrio è stabile. È importante tener presente che questo teorema dà solo condizioni sufficienti per la sincronizzazione, quindi anche se non possiamo trovare δ_1 e δ_2 , questo non significa che il sistema non può essere sincronizzato debolmente, nel senso che se le traiettorie perturbate corrispondenti alla regione interna D_o divergono dall'origine (che si assume essere instabile), quelle corrispondenti alle altre regioni (per ipotesi assunte come asintoticamente stabili), possono convergere allo zero, così che non c'è abbastanza tempo per ogni traiettoria per divergere troppo lontano dall'origine, prima di raggiungere di nuovo le altre due regioni D_{+1} o D_{-1} , dove D_o è la regione dello spazio compresa tra i piani $x=-1$ e $x=1$, D_{-1} è la regione dello spazio a sinistra del piano $x=-1$, D_{+1} è la regione a destra del piano $x=1$.

[10] Cfr. Leon Chua, op. cit. , pag. 97

- **Circuiti accoppiati attraverso le variabili x, y, e z**

Consideriamo due circuiti accoppiati attraverso la sola variabile di stato x (vedi figura 2.2)

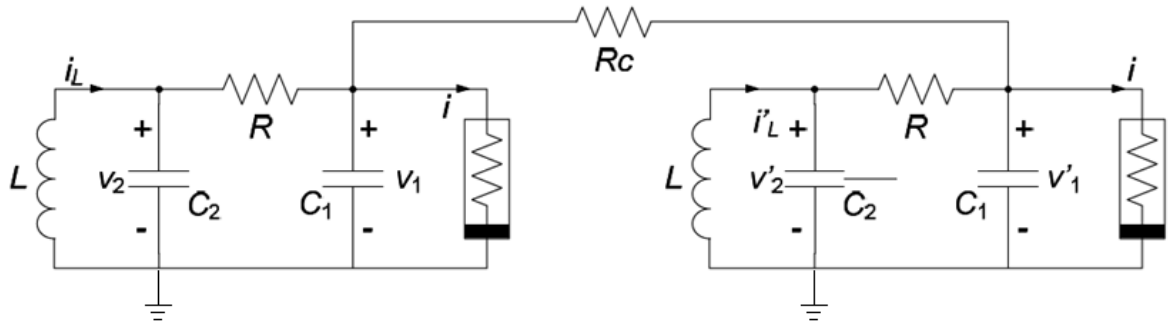


Figura 2.2 Due circuiti di Chua accoppiati attraverso la variabile x

Le equazioni in questo caso diventano:

$$\begin{cases} \dot{x}=\alpha(y-x-g(x))+\delta_x(x'-x) \\ \dot{y}=x-y+z \\ \dot{z}=-\beta y \\ \dot{x}'=\alpha(y'-x'+g(x'))+\delta_x(x-x') \\ \dot{y}'=x'-y'+z' \\ \dot{z}'=-\beta y' \end{cases}$$

dove:

$$\delta_x = \frac{R_a}{R_c}$$

Il sistema differenza è :

$$\dot{p}=\alpha q-\alpha p-\alpha s_i p-2\delta_x p$$

$$\dot{q}=p-q+r$$

$$\dot{r}=-\beta q$$

quindi :

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} -\alpha - s_i\alpha - 2\delta_x & \alpha & 0 \\ 1 & -1 & 1 \\ 0 & -\beta & 0 \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

Dove $s_i=a$, b; $i=1$, 2

L'equazione caratteristica è:

$$\lambda^3+k\lambda^2+\rho\lambda+\sigma=0$$

se $\sigma>0$, $\rho>0$, e $k\rho-\sigma=0$ allora $p=q=r=0$ è un punto stabile e i sottosistemi saranno sincronizzati.

Ponendo il valore di $\delta_x = 5.56$ di conseguenza i sottosistemi saranno sincronizzati per ogni $\delta_x \geq 5.56$. Per valori minori è impossibile fare previsioni, ma dalle misure sperimentali e numeriche[11], si ricava che il circuito sarà sincronizzato per $\delta_x>0.5$. Consideriamo due circuiti accoppiati attraverso la variabile y (vedi figura2.3):

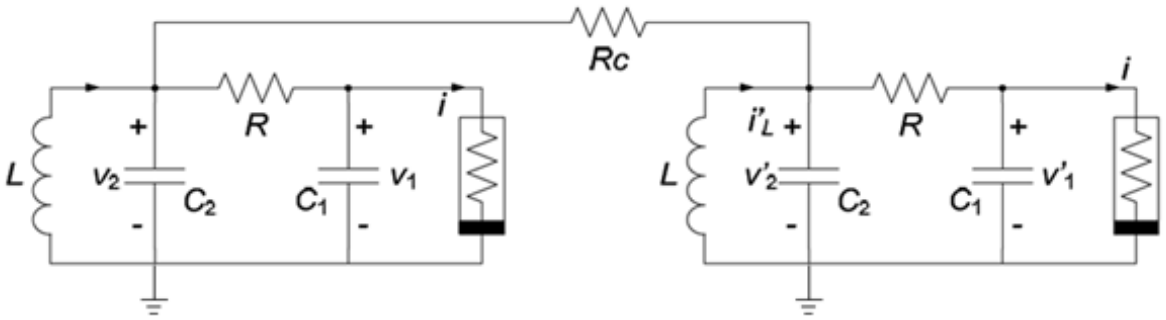


Figura 2.3 Due circuiti di Chua accoppiati attraverso la variabile y

Le equazioni di stato in questo caso sono:

$$\begin{cases} \dot{x}=\alpha(y-x-g(x)) \\ \dot{y}=x-y+z+\delta_y(y'-y) \\ \dot{z}=-\beta y \\ \dot{x}'=\alpha(y'-x'+g(x')) \\ \dot{y}'=x'-y'+z'+\delta_y(y-y') \\ \dot{z}'=-\beta y' \end{cases}$$

dove:

$$\delta_y = \frac{R}{R_c}$$

Sfortunatamente è impossibile applicare il teorema precedente in questo caso, poiché si ha sempre almeno un autovalore positivo, ma questo non implica che questo sistema non sarà mai sincronizzato. Sperimentalmente si ha sincronizzazione per $\delta_y>5.5$ e con calcoli numerici[12] si ha per $\delta_y>1$. Consideriamo adesso due circuiti accoppiati attraverso la variabile z:

$$\begin{cases} \dot{x}=\alpha(y-x-g(x)) \\ \dot{y}=x-y+z \\ \dot{z}=-\beta y+\delta_z(z'-z) \\ \dot{x}'=\alpha(y'-x'+g(x')) \\ \dot{y}'=x'-y'+z' \\ \dot{z}'=-\beta y'+\delta_z(z-z') \end{cases}$$

[11] Cfr. Leon Chua, Makoto Itoh, Ljupco Kocarev e Kevin Eckert , *Chaos Synchronization in Chua's circuit*, Journal of circuits, Systems and Computers, Vol. 3 N° 1(1993)Pag. 100
[12] Cfr. Leon Chua, op. cit. , pag. 100

come per l'accoppiamento attraverso la variabile y, anche in questo caso non possiamo applicare il teorema precedente, per la stessa ragione. Numericamente la sincronizzazione è stata trovata per $0.7<\delta_z<2$, per un set particolare di condizioni iniziali.[13]

- **Condizioni di accoppiamento**

Si consideri una rete di N circuiti caotici collegati tra di loro attraverso una o più variabili di stato. Chiamiamo A la matrice di connessione, tale che:

$\forall\ i\neq j$

se il nodo i ed il nodo j sono accoppiati, $a_{ij}=1$, altrimenti $a_{ij}=0$

$\forall\ i=j$

gli elementi della diagonale principale soddisfano l'eguaglianza:

$$a_{ii} = -\sum_{j=1\ j\neq i}^{N} a_{ij} \qquad \text{per } i = 1,2,\dots,N$$

Questa condizione implica che un autovalore della matrice A è 0 con molteplicità 1 e gli altri sono negativi.

Gli stati di sincronizzazione dei circuiti nella rete possono essere caratterizzati da questi valori diversi da zero della matrice A.

Si dice che una rete dinamica è sincronizzata (asintoticamente) se :

$$x_1(t)=x_2(t)=\dots\dots=x_n(t) \qquad \text{con } t\to\infty \ .$$

Conoscendo le dinamiche di un nodo isolato ed il legame di connessione con gli altri nodi, si ha che lo stato della rete sincronizza con la soluzione $s(t) \in \Re^N$ di un nodo isolato se :

$$s(t)=x_1(t)=x_2(t)=\dots\dots=x_n(t) \qquad \text{con } t\to\infty$$

dove s(t) può essere un punto di equilibrio, un'orbita periodica o un attrattore caotico e la dinamica della rete è determinata dalla dinamica di un nodo isolato:

$$\dot{s}(t)=f(s(t))$$

Lemma 1: consideriamo la rete costituita da N circuiti identici accoppiati linearmente attraverso il primo stato di ogni circuito (nodo). Ogni nodo costituisce un sistema dinamico n dimensionale del tipo:

$$\dot{x}_i=f(x_i)+u_i \quad \text{per } u_i = \sum_{j=1}^{N} a_{ij}\, x_{j1} \quad \text{(vedi paragrafi seguenti).}$$

Supponiamo che λ_2 sia il più grande autovalore diverso da zero della matrice di connessione A, lo stato di sincronizzazione della rete definito come sopra è asintoticamente stabile se:

$$\lambda_2 \leq -\frac{T}{c}$$

[13] Cfr Leon Chua, Makoto Itoh, Ljupco Kocarev e Kevin Eckert , *Chaos Synchronization in Chua's circuit*, Journal of circuits, Systems and Computers, Vol. 3 N° 1(1993)Pag. 103

(1)

dove $c>0$ è il legame di accoppiamento della rete e T è una costante positiva della rete, tale che zero è un punto di stabilità del sistema ad N dimensioni

$$\dot{z}_1=f_1(z)-Tz_1$$

$$\dot{z}_2=f_2(z)$$

……

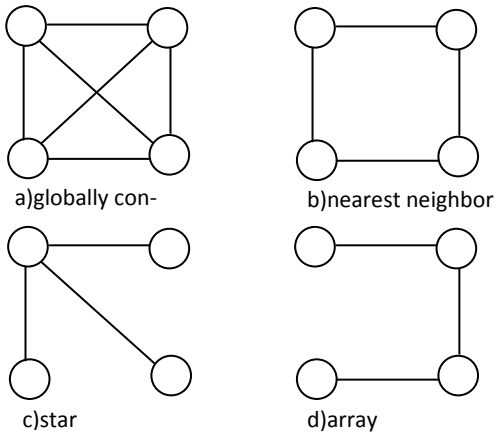
……

$$\dot{z}_N=f_N(z)$$

Tale sistema[14] è il modello di un singolo nodo con autoretroazione $-Tz_1$, dove T è la costante positiva tale da, come detto prima, stabilizzare una singola cella; l'intera rete viene stabilizzata se rispetta la condizione (1).

2.1.3 Topologie regolari di N circuiti di Chua

Mostriamo adesso alcune possibili topologie di connessione[15] fra i nodi di un circuito:



- **Globally-coupled network**

Tale configurazione si ha quando ogni circuito- nodo è direttamente connesso a tutti gli altri nodi della rete.

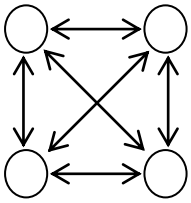


Figura 2.4 globally connected

La corrispondente matrice di connessione è :

[14] Cfr. Posadas- Castillo,Hernandez,Cruz-Hernandez e Lopez-Gutierrez, *Syncronization in a network of Chua's circuits*, forth lasted International conference , CIRCUITS, SIGNALS AND SYSTEMS,novembre 2006,San Francisco , CA,USA. Pag. 237

[15] Cfr. Posadas- Castillo,Hernandez,Cruz-Hernandez e Lopez-Gutierrez op.cit. pag. 237

$$A_{gc} = \begin{bmatrix} -N+1 & 1 & 1 & \cdots & 1 \\ 1 & -N+1 & 1 & \cdots & 1 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 1 & 1 & 1 & \dots & 1 \\ 1 & 1 & 1 & \cdots & -N+1 \end{bmatrix}$$

Questa matrice ha un singolo autovalore uguale a zero e tutti gli altri uguali a $-N$. Il secondo autovalore più grande λ_{2gc}=-N decresce a -∞ per N→∞ :

$$\lim_{N\to\infty} \lambda_{2gc} = 0$$

Esempio con quattro reti Chua:

$$A_{gc}=\begin{bmatrix} -3 & 1 & 1 & 1 \\ 1 & -3 & 1 & 1 \\ 1 & 1 & -3 & 1 \\ 1 & 1 & 1 & -3 \end{bmatrix}$$

Questa matrice presenta un autovalore uguale a zero e tre autovalori pari a -4,per cui anche

λ_2=-4, e chiamando R_c la resistenza di collegamento tra i Chua della rete

$$R_c \leq \frac{C_2 R|\lambda_2|}{K_1 C_1} = 11283\Omega$$

Per i seguenti valori nominali:
C_2=10n,R=1780,λ_2=4,K=6.31,C_1=100n, ed anche con uno scostamento di circa ±10 % di R_c, si avrà sincronizzazione, tanto migliore quanto più bassa è la resistenza R_c. Aumentando, invece la resistenza, si avrà prima un peggioramento, e poi un miglioramento. Per un valore[16] di c=10, con le condizioni iniziali x_{11}=1,, x_{12}=-1, x_{13} =3, x_{14}=-3,ed un collegamento attraverso la prima variabile di stato, la sincronizzazione si raggiunge in circa 5 sec. Per un collegamento attraverso la seconda variabile di stato, per es. x_{11},, x_{22}, x_{32}, x_{42}, la sincronizzazione si raggiungerà in circa 15 sec.,e così pure per un legame attraverso la terza variabile di stato,ad es. x_{13},, x_{23}, x_{33}, x_{43}.

- **Nearest-neighbor coupled network**

Questa configurazione consiste in N circuiti collegati ad anello (vedi figura 2.5) ed ogni cella i è adiacente alle celle vicine $i\pm1$, $i\pm2$, $i\pm3$, ….. , $i\pm k/2$, dove k=costante.

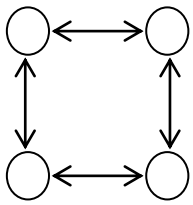


Figura 2.5 Nearest Neighbor coupled network

Le equazioni di stato della rete sono :

[16] Cfr. Posadas Castillo,*op.cit*.,pag.239

$$\dot{x}_i = f(x_i) + c\sum_{j=1}^{k/2} A_{nc}(x_{i+j} + x_{i-j} - 2x_i) \quad \text{per } i=1, 2, \ldots, N$$

La matrice A_{nc} è data da:

$$A_{nc} = \begin{bmatrix} -2 & 1 & 0 & \cdots & 1 \\ 1 & -2 & 1 & 0 & \cdots \\ \vdots & \ddots & \ddots & \cdots & \ddots \\ 0 & \ddots & 1 & -2 & 1 \\ 1 & 0 & \cdots & 1 & -2 \end{bmatrix}$$

e il secondo autovalore più grande può essere trovato in questo modo:

$$\lambda_{2nc} = \sum_{j=1}^{k/2} \operatorname{sen}^2\left(\frac{j\pi}{N}\right)$$

$$\lim_{N\to\infty} \lambda_{2nc} = 0$$

Questa condizione implica che la rete non è sincronizzata, se il numero di circuiti è grande.
Esempio con quattro circuiti di Chua:

$$A_{nc} = \begin{bmatrix} -2 & 1 & 0 & 1 \\ 1 & -2 & 1 & 0 \\ 0 & 1 & -2 & 1 \\ 1 & 0 & 1 & -2 \end{bmatrix}$$

Gli autovalori sono λ_1=-4 , λ_2= λ_3=-2, λ_4=0 con il legame di connessione c=10 e come condizioni iniziali le stesse del caso precedente, il tempo per raggiungere la sincronizzazione con un collegamento di tutti i nodi attraverso a prima variabile sarà di circa 20 sec., maggiore quindi del caso precedente.[17] In tal caso dovrà essere $R_c \leq$ 5641Ω, utilizzando la formula già vista nel precedente paragrafo:

$$R_c \leq \frac{C_2 R|\lambda_2|}{K_1 C_1}$$

e gli stessi parametri nominali di cui sopra.

Tale valore costituirà, come visto sopra, il limite massimo tra sincronizzazione e non-sincronizzazione.

•

[17] Cfr. Posadas Castillo,*op.cit.*,pag.239

- **Star coupled network**

La configurazione di accoppiamento di tipo star è mostrata nella figura seguente:

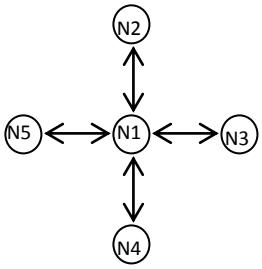


Figura 2.6 Collegamento a stella

la matrice A_{sc} è:

$$A_{sc}=\begin{bmatrix} -N+1 & 1 & 1 & \cdots & 1 \\ 1 & -1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 1 & 0 & 0 & \cdots & \vdots \\ 1 & 0 & 0 & \cdots & -1 \end{bmatrix}$$

Gli autovalori sono {0,-N,-1, … ,-1}.
D'altra parte il secondo più grande auovalore è λ_{2sc}=-1 , che non è legato alla dimensione della rete, poiché:

$$\lim_{N\to\infty} \lambda_{2sc} = -1$$

Per cinque circuiti accoppiati bidirezionalmente a stella la matrice di connessione è:

$$A_{st} = \begin{bmatrix} -4 & 1 & 1 & 1 & 1 \\ 1 & -1 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 & -1 \end{bmatrix}$$

Gli autovalori sono: {0, -5, -1 , -1 , -1} con λ_2=-1

- **Circuiti di otto Chua**

Per una connessione di tipo **nearest neighbor** di otto Chua (vedi Figura 2.7),

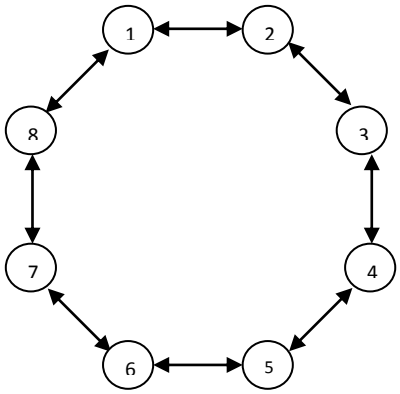


Figura 2.7 Collegamento di tipo nearest neighbor, con otto circuiti

la matrice di collegamento è:

$$A_{nn} = \begin{bmatrix} -2 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & -2 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -2 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -2 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -2 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & -2 \end{bmatrix}$$

I cui autovalori sono: -4.0000 -3.4142 -3.4142 -2.0000 -2.0000 -0.5858 -0.5858 0.0000,come si può vedere tutti interi, poiché la matrice è simmetrica, e negativi; λ_2 =-0,5858 e la resistenza di collegamento sarà allora, per K_{lim} = 6.31, (valore trovato sperimentalmente in laboratorio) e $|\lambda_2|=0.58$

$$R_c \leq \left| \frac{\lambda_2 C_2 R}{K C_1} \right| = 1.474 K\Omega$$

che possiamo considerare il limite massimo per ottenere una buona sincronizzazione.

Per una connessione del tipo **globally connected degli** otto Chua,(figura2.8)

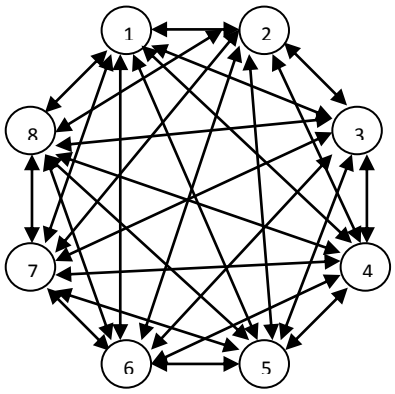


Figura 2.8 collegamento di tipo globally connected, con otto circuiti di Chua

La matrice di collegamento è:

$$A_{gc} = \begin{bmatrix} -7 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -7 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & -7 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & -7 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -7 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & -7 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & -7 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & -7 \end{bmatrix}$$

I suoi autovalori sono -8.0000 -8.0000 -8.000 -8.000 -8.0000 -8.0000 -8.0000 0.0000, come si può vedere tutti interi, poiché la matrice è simmetrica, e negativi; λ_2 = -8 e la resistenza di collegamento sarà allora, per questi valori trovati di K e λ_2 :

$$R_c \leq \left| \frac{\lambda_2 C_2 R}{K C_1} \right| = 20.056\ K\Omega$$

che possiamo considerare il limite massimo per ottenere una buona sincronizzazione.

Per una connessione di tipo **Hub** degli otto Chua (vedi figura 2.9)

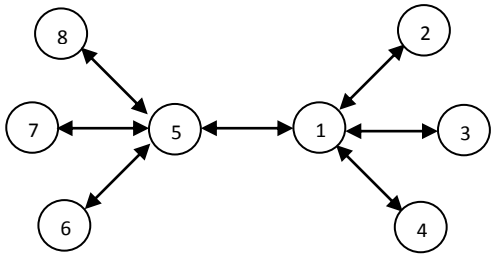


Figura 2.9 Collegamento di tipo Hub di otto circuiti

La matrice di connessione è:

$$A_{hub} = \begin{bmatrix} -4 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & -4 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 \end{bmatrix}$$

I suoi autovalori sono -5.6458 -4.0000 -1.0000 -1.0000 -1.0000 -1.0000 -0.3542 0.0000

λ_2=-0.3542 $R_c \leq 999\Omega$.

Per una connessione di tipo **second nearest neigbor** degli otto Chua(vedi Figura 2.10):

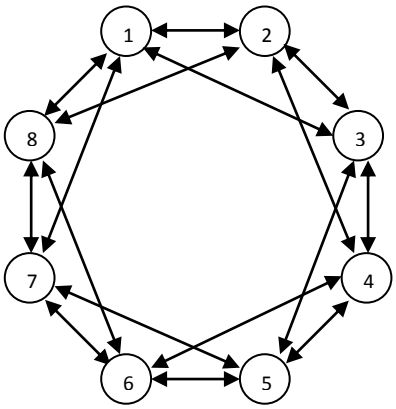


Figura 2.10 Configurazione second nearest neighbor

la matrice di connessione è:

$$A_{snn}=\begin{bmatrix} -4 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & -4 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & -4 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & -4 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & -4 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & -4 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & -4 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & -4 \end{bmatrix}$$

Gli autovalori di tale matrice sono:

0 , -2.59 , -5.41 , -4 , -6 , -2.59 , -6 , -5.41

$\lambda_2 =$ -2.59 $R_c = 7.306K\Omega$

Mostreremo in seguito alcune verifiche eseguite con il programma SwitcherCad della sincronizzazione di 4 e 8 circuiti, utilizzando per le resistenze di collegamento tra i vari circuiti il valore della resistenza di soglia, cioè della resistenza ottenuta con il programma Matlab per valori nominali dei circuiti di Chua, la quale, come detto precedentemente, costituisce il limite tra sincronizzazione e non sincronizzazione. Per verificare sperimentalmente i circuiti con SwitcherCad ed avvicinarsi il più possibile alla realtà, sono state introdotte le tolleranze (0,01% per le capacità, 0.5% per le resistenze). Infatti a livello industriale è pressoché impossibile realizzare resistori o condensatori identici, ed inoltre è importante conoscere le condizioni iniziali di un sistema caotico per verificarne il comportamento degli istanti successivi.

2.1.4 Simulazioni con SwitcherCad

- **Quattro circuiti di Chua, connessi in modalità globally connected**

Nella figura seguente è mostrato il comportamento di quattro circuiti di Chua x_1 , x_2 , x_3 , x_4 inizialmente non accoppiati. I parametri di questi circuiti sono gli stessi che abbiamo usato precedentemente, ma con una tolleranza random di max 0.5% per le resistenze e di max 0.01% per le capacità. Nella figura successiva mostriamo la simulazione SwitcherCad del comportamento degli stessi quattro circuiti x_1 , x_2 , x_3 , x_4 collegati tra loro secondo la modalità globally connected attraverso la variabile x:

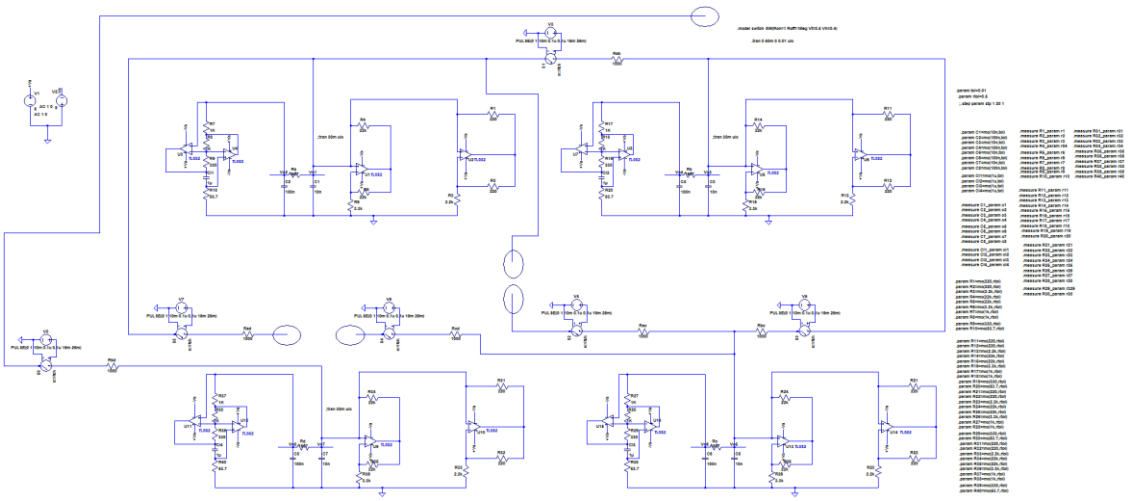


Figura 2.11a Schema di quattro circuiti di Chua, non collegati tra loro

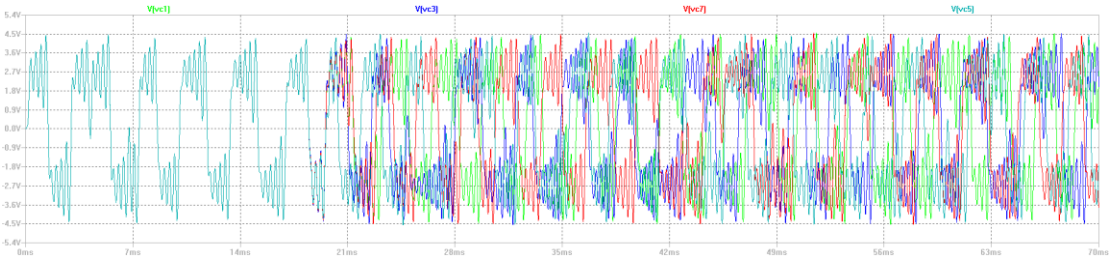


Figura 2.11b Simulazione SwitcherCad nel piano delle fasi, del comportamento di quattro circuiti di Chua con parametri e tolleranze definiti precedentemente, non collegati tra loro,

Nella figura successiva mostriamo, con una simulazione SwitcherCad, il comportamento degli stessi quattro circuiti collegati tra loro, secondo la modalità globally-connected, attraverso la variabile x, con una resistenza di collegamento R_c=11000Ω. La resistenza di soglia per un rete di questo tipo, con quattro circuiti di Chua come nodi, in modalità globally connected, trovata con il Matlab, è R_{th}=11284Ω, per parametri nominali di questi circuiti, Klim=6.31, λ_2=-4

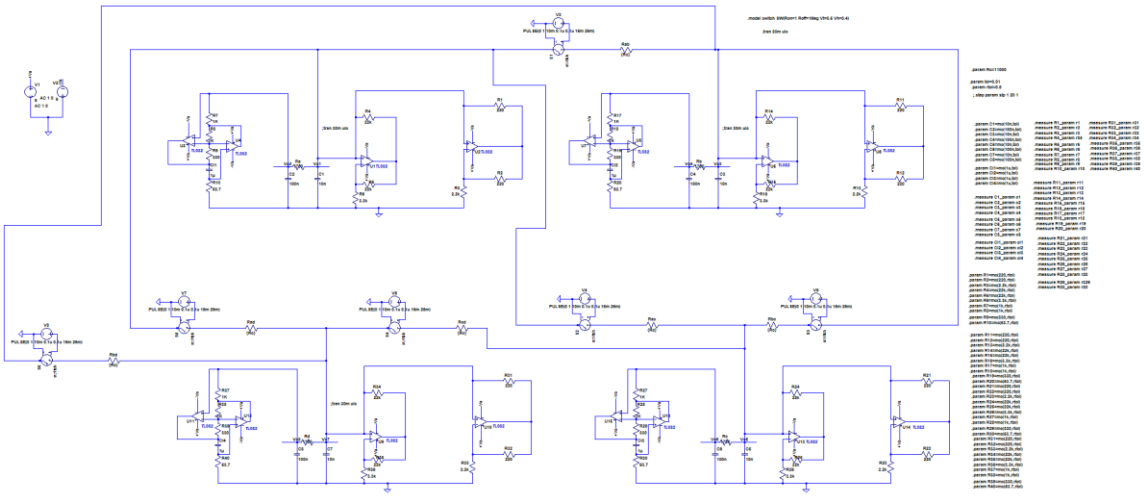


Figura 2.12a schema di quattro circuiti di Chua, collegati tra loro secondo la modalità globally-connected attraverso il parametro x, con R_c=11000

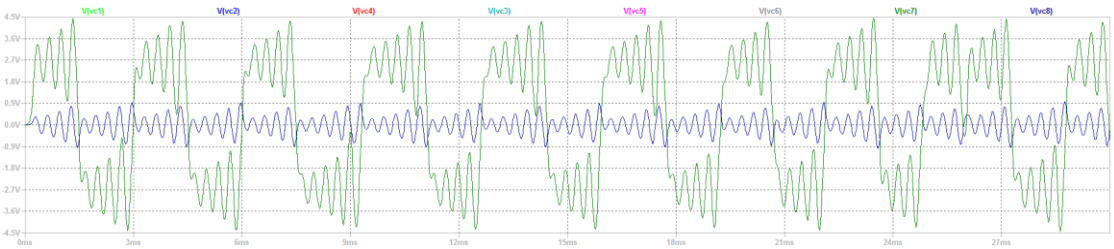


Figura 2.12 b Simulazione SwitcherCad, nel piano delle fasi , del comportamento di quattro circuiti di Chua, con parametri e tolleranze precedentemente citati, collegati tra loro attraverso il parametro x

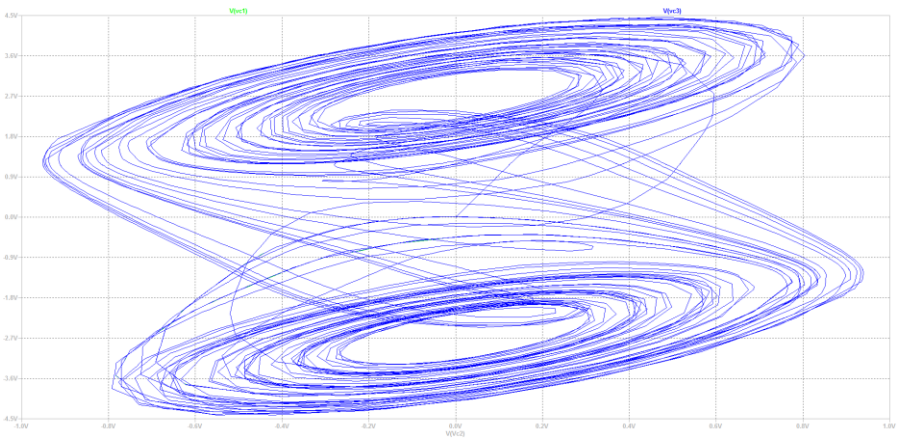


Figura2.12c Simulazione SwitcherCad, nello spazio di stato,del comportamento dei precedenti quattro circuiti di Chua

Gli interruttori utilizzati nella figura 2.12a permettono di avere una fase iniziale in cui i circuiti lavorano autonomamente; con la chiusura degli interruttori si ha interazione tra i nodi e si possono valutare le condizioni di sincronizzazione.

- **Otto circuiti di Chua connessi in modalità globally connected**

Per le prossime simulazioni, a causa del grande numero di nodi e di connessioni, utilizzeremo il subcircuito mostrato in figura 2.13, che comprende anche le tolleranze random per resistenze e capacità, fatta eccezione per R, per poter rappresentare i circuiti di Chua con il new symbol rappresentato in figura 2.14:

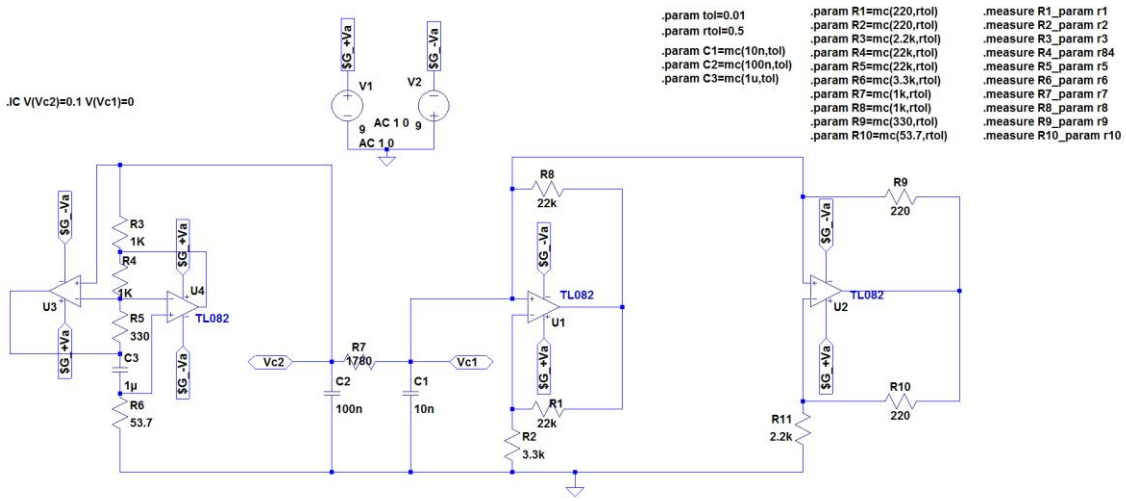


Figura 2.13 Subcircuito che comprende anche le tolleranze random per i parametri del circuito

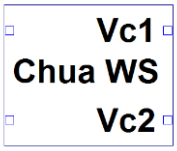


Figura 2.14 new symbol del circuito di Chua

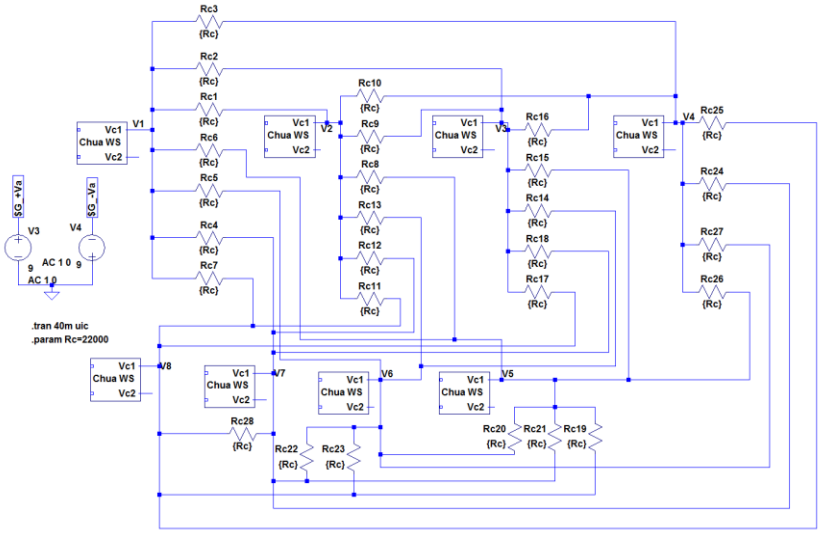


Figura 2.15a Schema di otto circuiti di Chua, collegati tra loro secondo la modalità globally-connected attraverso il parametro x, con R_c=22000Ω

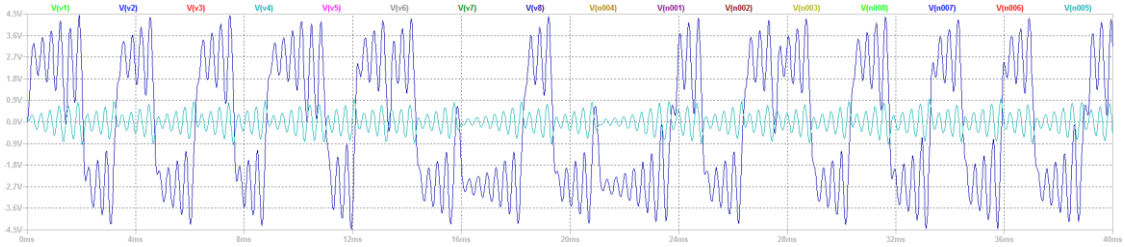


Figura 2.15b Simulazione SwitcherCad, nel piano delle fasi del comportamento di otto circuiti di Chua, con parametri e tolleranze precedentemente citati, collegati tra loro attraverso il parametro x

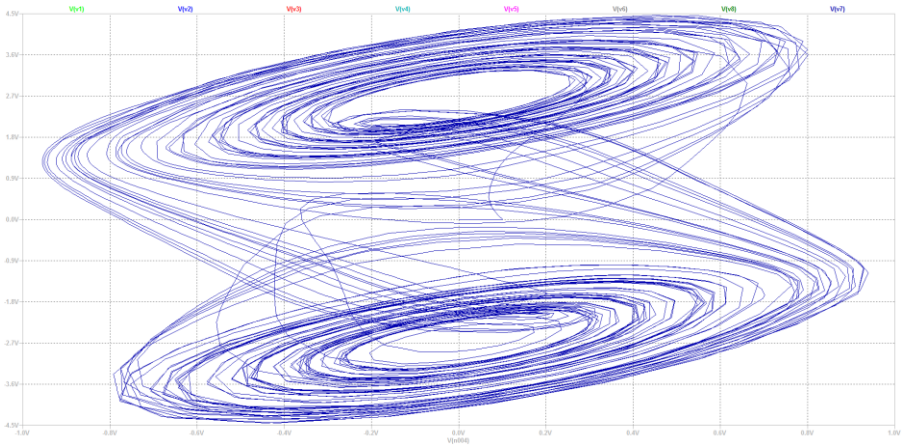


Figura 2.15c Simulazione SwitcherCad, nello spazio di stato del comportamento degli otto circuiti precedenti

2.2 Controllo

2.2.1 I sistemi drive – response[18]

Come abbiamo già visto nel paragrafo 2.1, si parla di connessione direzionale, (o drive-response), quando il circuito driver controlla il circuito slave (o response).Pecora e Carrol hanno dimostrato che i sistemi saranno sincronizzati se i CLE del sistema response sono tutti negativi. I CLE si trovano calcolando gli esponenti di Ljapunov per l'intero $2n\text{-}m$ sistema dimensionale e comparando questi agli esponenti di Ljapunov del sistema drive n-dimensionale. I rimanenti n-m esponenti di Ljapunov saranno i CLE del sistema response. C'è un metodo relativamente semplice per vedere se i CLE saranno negativi, supponendo che il sottosistema sia lineare. Sia $\xi(t)=w(t)-w'(t)$, chiamando ξ il sistema differenza, avremo: $\xi(t)=\dot{w}(t)-\dot{w}'(t) = h(v,w)-h(v,w')$. Se il sottosistema è lineare abbiamo: $\xi(t)=A\dot{\xi}(t)$, dove A è una $(n\text{-}m)x(n\text{-}m)$ matrice costante. Consideriamo come autovalori di A λ_1 , λ_2 , ... , $\lambda_{n\text{-}m}$; le parti reali di questi autovalori sono per definizione i CLE cercati; se tutti i CLE sono negativi allora $\lim_{t\to\infty}\xi(t) = 0$ e i sottosistemi saranno sincronizzati; se c'è un CLE positivo, i sistemi si svilupperanno separatamente più lontano, per $t\to\infty$ e di conseguenza non saranno mai sincronizzati. Un caso intermedio occorre se uno o più CLE sono zero, ma nessuno è positivo; per $t\to\infty$ i sottosistemi saranno separati da una distanza fissa B, dipendente dalla condizioni iniziali. Se i sottosistemi sono sottocircuiti lineari con elementi passivi, è banale calcolare i CLE, se i sottosistemi sono non lineari, i CLE non sono così facilmente determinati, quindi è necessario ricorrere ad un software. In seguito ci occuperemo di tre configurazioni di sistemi drive- response.

Configurazioni x-drive, y-drive, z-drive

Nella configurazione x-drive, come nelle altre due configurazioni prese in esame, il circuito master è collegato al resto del sistema, (nella figura 2.16 ad un altro circuito), attraverso un amplificatore operazionale di tipo buffer, che, a causa della sua resistenza di ingresso infinita, e della bassa resistenza di uscita, permette il collegamento monodirezionale del driver al master.

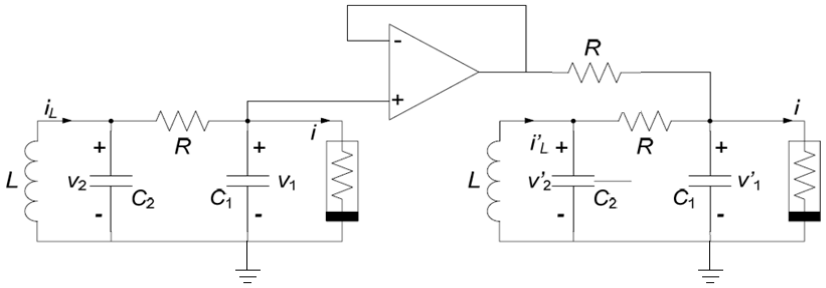


Figura 2.16 Configurazione x-drive

Le equazioni di stato diventano:

$$\dot{x}=a(y-x-g(x)) \qquad \dot{y}'=x-y'+z'$$

$$\dot{y}=x-y+z \qquad \dot{z}'=-\beta y'$$

$$\dot{z}=-\beta y$$

[18] Cfr Leon Chua, Makoto Itoh, Ljupco Kocarev e Kevin Eckert , *Chaos Synchronization in Chua's circuit*, Journal of circuits, Systems and Computers, Vol. 3 N° 1(1993)Pag. 98 e sgg.

Il sistema differenza in forma matriciale sarà allora:

$$\begin{bmatrix}\dot{q}\\ \dot{r}\end{bmatrix} = \begin{bmatrix}-1 & 1\\ -\beta & 0\end{bmatrix}\begin{bmatrix}q\\ r\end{bmatrix}$$

I CLE sono -0.5 , -0.5 e i sistemi sincronizzano.

Nella configurazione y-drive, rappresentata nella figura seguente:

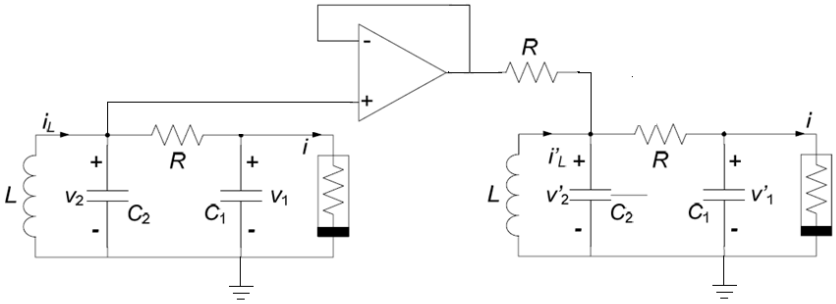


Figura 2.17 Configurazione y-drive

Le equazioni di stato sono:

$$\dot{x}=a(y-x-g(x)) \qquad \dot{x}'=\alpha(y-x'-g(x'))$$

$$\dot{y}=x-y+z \qquad \dot{z}'=-\beta y'$$

$$\dot{z}=-\beta y$$

I CLE sono -2.5 , ±0.05 , 0; come da aspettativa il sistema sincronizza, ma $z(\tau)$ e $z'(\tau)$ rimarranno a una distanza costante B tra di loro.
Nella configurazione z-drive, le equazioni di stato del circuito sono:

$$\dot{x}=a(y-x-g(x)) \qquad \dot{x}'=\alpha(y-x'-g(x'))$$

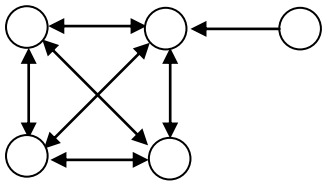
$$\dot{y}=x-y+z \qquad \dot{y}'=x-y'+z'$$

$$\dot{z}=-\beta y$$

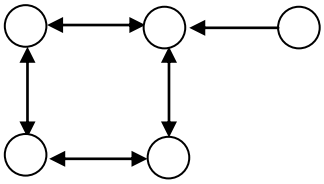
In questo caso sperimentalmente si è visto che è impossibile avere sincronizzazione, poiché i CLE non sono tutti negativi.

2.2 Il pinning

Il pinning è la tecnica che consente di pilotare una rete di dispositivi attraverso un dispositivo leader.



Globally connected



Nearest neighbor coupled

Il leader è collegato in modo unidirezionale tramite un amplificatore operazionale ad uno o più dispositivi della rete e tende a forzare quest'ultima a seguire la propria traiettoria.
Consideriamo la sincronizzazione multipla di circuiti di Chua accoppiati, in comportamento caotico; imponiamo particolari dinamiche da un circuito master ai circuiti slaves della rete complessa, quando i circuiti sono accoppiati con diverse topologie, come per esempio accoppiamenti di tipo globally coupled, nearest neighbor, star.

I sistemi naturali ed artificiali sono composti da migliaia di unità (celle), delle quali noi possiamo conoscere o non conoscere il comportamento individuale.
Consideriamo una rete complessa di N celle identiche, accoppiate linearmente attraverso la variabile x; ogni cella costituisce un sistema dinamico ad N dimensoni come segue:

$$\dot{x}_i = f(x_i) + u_i$$

oppure in modo esplicito:

$$\begin{aligned}
&\dot{x}_{i1} = f_1(x_i) + u_{i\,1} \\
&\dot{x}_{i2} = f_2(x_i) \\
&\ldots\ldots\ldots\ldots \\
&\ldots\ldots\ldots\ldots \\
&\dot{x}_{iN} = f_N(x_i)
\end{aligned}$$

dove $x_i = (x_{i1}, x_{i2}, \ldots\ldots, x_{iN})^T \in \mathcal{R}^N$ sono gli stati della cella i; $u_i \in \mathcal{R}$, $u_{i1} = c\sum_{j=1}^{N} a_{ij}\, x_{j1} \in \mathcal{R}$ sono gli ingressi della cella, $A = (a_{ij})_{NxN}$ è la matrice di accoppiamento che rappresenta la configurazione di accoppiamento nella rete, e $c > 0$ rappresenta il legame di connessione tra i nodi della rete.
Poiché $c\lambda_2 \leq \bar{d}$, ed essendo $\lambda_2 < 0$, $\bar{d} < 0$, la disuguaglianza è equivalente a $c \geq \left|\frac{\bar{d}}{\lambda_2}\right|$ e nel caso di collegamento attraverso la sola variabile x, possiamo mettere $\bar{d} = a = 10$. Un grande valore del modulo di λ_2 implica che la rete può essere sincronizzata per un piccolo valore del parametro di connessione c.

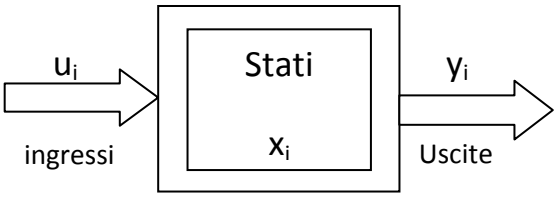


Figura 2.18 cella isolata con ingressi u_{i1} stati x_i ed uscite y_{i1}

La figura mostra la struttura di una cella isolata con un singolo ingresso ed una singola uscita; l'output può essere definito come $y_{i1} = g_i(x_i)$ dove g_i è una funzione non lineare degli stati, d'altra parte in molti casi l'uscita che ci interessa coincide con uno degli stati, per esempio $y_{i1} = x_{i1}$ cioè il primo stato di ogni cella.

- **Star coupled network con un nodo isolato(master)**

Nei casi precedenti abbiamo considerato una rete di circuiti di Chua senza nodo isolato,'master'; adesso consideriamo il caso in cui N_1 è isolato e questo nodo imporrà le sue dinamiche agli altri (N_2, N_3 , N_4) come da figura 2.19:

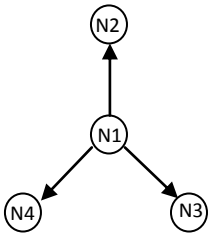


Figura 2.19 Network di quattro celle accoppiate in configurazione star con il nodo isolato N_1 che impone le sue dinamiche caotiche agli altri (N_2 , N_3 , N_4)

La corrispondente matrice di connessione è la seguente:

$$A_{sc}=\begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 1 & 0 & -1 & 0 \\ 1 & 0 & 0 & -1 \end{bmatrix}$$

I corrispondenti autovalori sono $\lambda_1= \lambda_2= \lambda_3= \lambda_4=-1$, con il fattore di accoppiamento c=10 e con condizioni iniziali le stesse del caso precedente, con questi valori il lemma 1 garantisce la completa sincronizzazione del network della rete.
Noi possiamo osservare che il tempo di sincronizzazione[19] per il il collegamento attraverso la variabile x dei circuiti di Chua (x_{11} , x_{21} , x_{31} ,x_{41}) è approssimativamente 20s.

- **Globally coupled network con nodo isolato(master)**

Consideriamo un network di circuiti di Chua in configurazione globally coupled (fig 2.10.3) dove il nodo N_1 è isolato ed imporrà le sue dinamiche caotiche agli altri nodi della rete (N_2 , N_3 , N_4) .

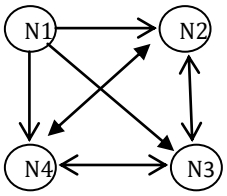


Figura 2.20 Network di quattro celle accoppiate in configurazione globally coupled con il nodo isolato N_1 che impone le sue dinamiche caotiche agli altri (N_2 , N_3 , N_4)

La matrice A_{gc} sarà data da:

$$A_{gc}\begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & -3 & 1 & 1 \\ 1 & 1 & -3 & 1 \\ 1 & 1 & 1 & -3 \end{bmatrix}$$

I corrispondenti autovalori sono (λ_1=-4 , λ_2=-1 , λ_3=-4 , λ_4=0) .

[19] Cfr.Posadas-Castllo ed altri,*op. cit.*,pag.240

2.2.3 Collegamenti master- slave

Il circuito s funge da master per la rete di otto circuiti collegati con la configurazione globally connected, cioè impone le sue dinamiche agli altri nodi del circuito(x_1 , x_2 , x_3 , x_4 , x_5 , x_6 , x_7 , x_8)

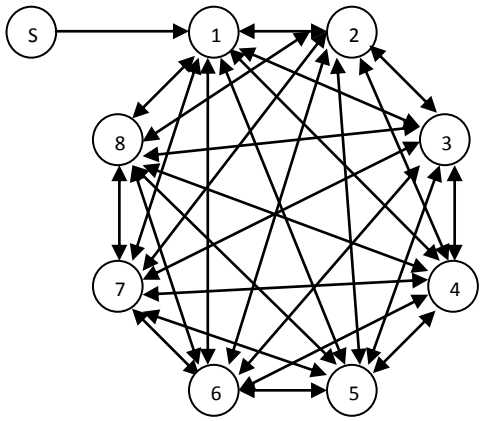


Figura 2.21 configurazione di tipo master slave di una rete di otto Chua con collegamento di tipo globally connected. Il circuito s funge da master ed è collegato ad un solo nodo.

La matrice di connessione per tale rete è la seguente, se il circuito master è collegato ad un solo nodo:

$$A_p = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -8 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & -7 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & -7 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & -7 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & -7 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & -7 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & -7 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & -7 \end{bmatrix}$$

Gli autovalori sono:

-0.1125 -8.0000 -8.8875 -8.0000 -8.0000 -8.0000 -8.0000 -8.0000 0

$\lambda_2 =$ -0.1125

$R_c = 317\Omega$

Se i nodi pinnati sono N, cioè tutti i nodi del circuito, λ_2 è sempre uguale a -1 ,altrimenti ha un valore maggiore (minore in valore assoluto), per cui anche R_c diminuisce con il diminuire del numero dei nodi pinnati.

2.2.4 Simulazioni con SwitcherCad

- **Pinning di otto Chua globally connected**

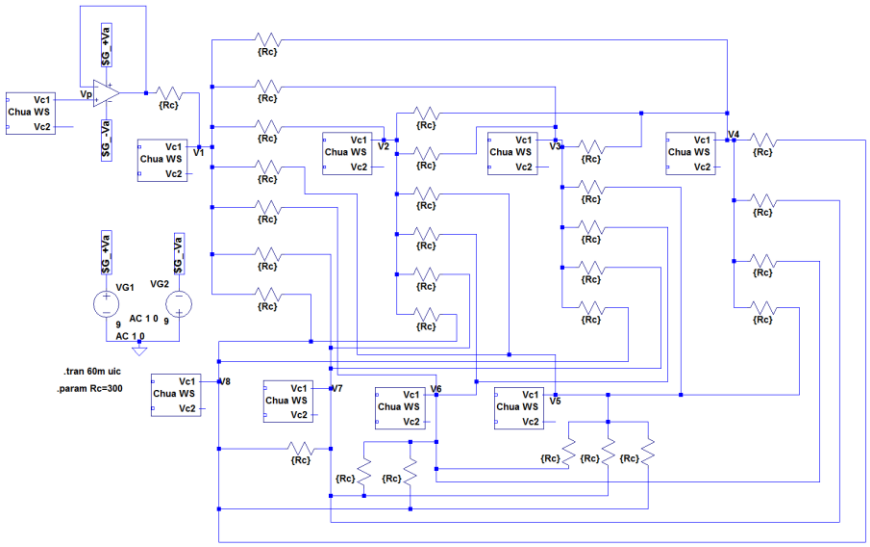


Figura 2.22a Schema SwitcherCad del pinning di otto Chua con connessione globally connected , con 1 nodo pinnato ed R_c=300Ω R_{th}=317Ω

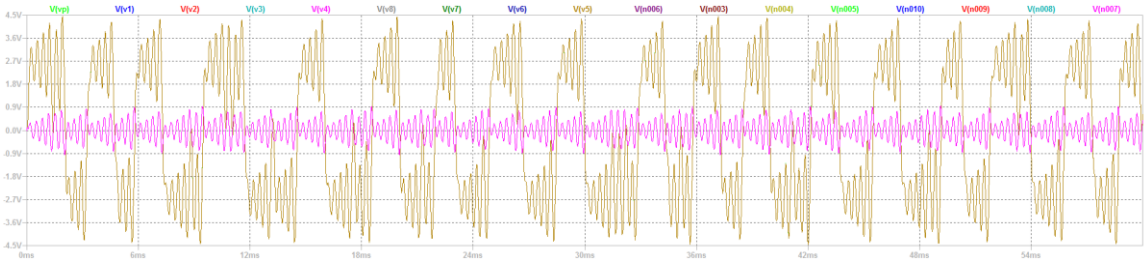


Figura 2.22b Simulazione SwitcherCad, nel piano delle fasi, del pinning di otto circuiti di Chua globaly connected ,con un nodo pinnato, con R_c=300Ω, R_{th}=317Ω

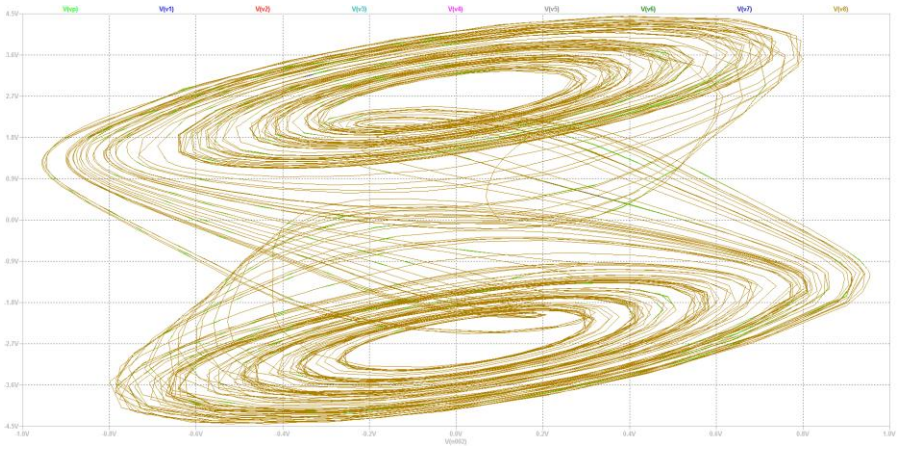


Figura 2.22c Simulazione SwitcherCad, nello spazio di stato, del pinning otto circuiti di Chua globaly connected ,con un nodo pinnato, con R_c=300Ω, R_{th}=317Ω

- **Pinning di otto Chua nearest-neighbor coupled**

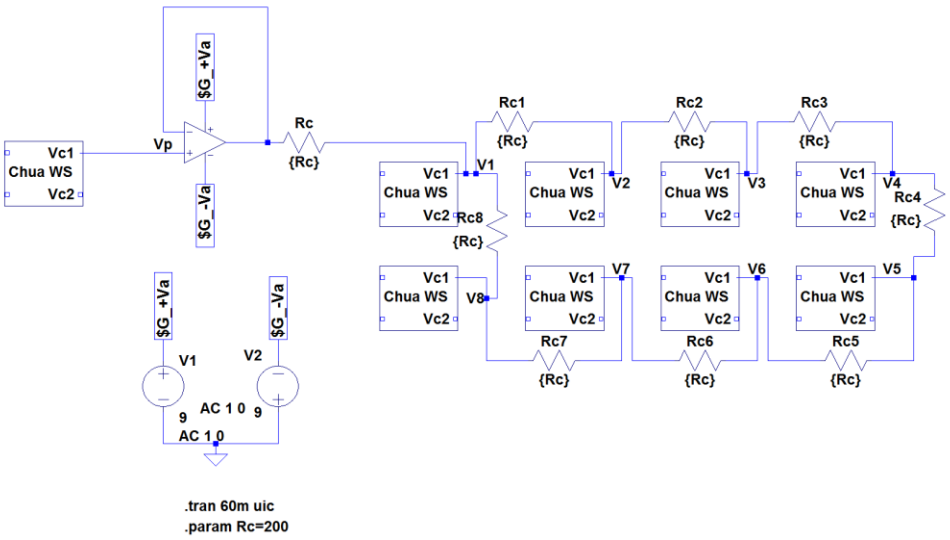


Figura 2.23a Schema pinning di otto Chua con connessione nearest neighbor per simulazione SwitcherCad, con un nodo pinnato, R_c=200Ω R_{th}=205Ω

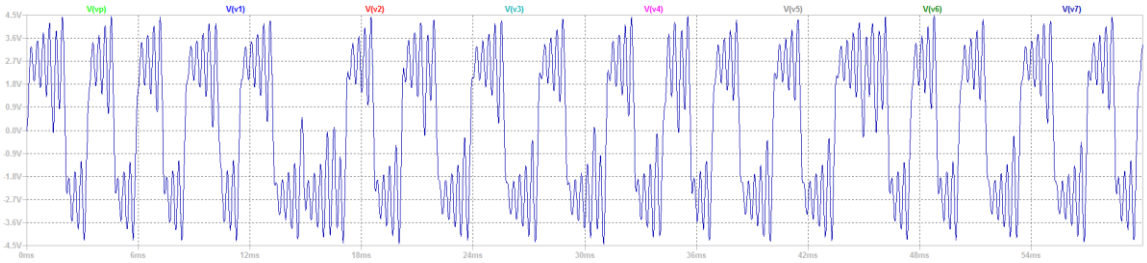


Figura 2.23b Simulazione SwitcherCad, nel piano delle fasi, del pinning di otto circuiti di Chua nearest-neighbor coupled, con un nodo pinnato, con R_c=200Ω, R_{th}=205Ω

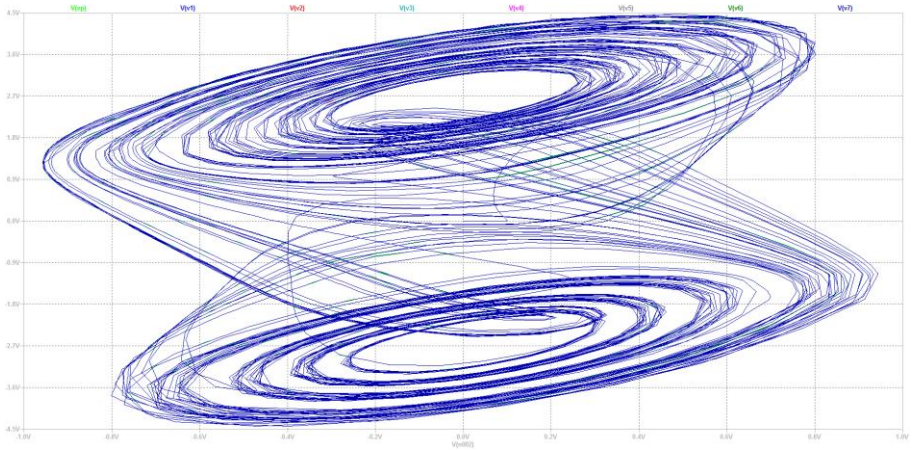


Figura 2.23c Simulazione SwitcherCad, nello spazio di stato, del pinning otto circuiti di Chua nearest-neighbor coupled ,con un nodo pinnato, con R_c=300Ω, R_{th}=317Ω

Come si può notare, in entrambe le simulazioni, le R_c utilizzate sono molto basse(rispettivamente 300Ω e 200Ω) ed aumentano per ogni nodo pinnato in più, fino ad arrivare entrambe a R_{th}=2820Ω, per λ_2=-1, se tutti i nodi sono pinnati, come si vedrà in seguito nelle simulazioni Matlab.

Cap.3 Caratterizzazione topologica di reti complesse

3.1 introduzione

Una rete [20] è un insieme di elementi, chiamati vertici o nodi, connessi tra di loro; nel mondo reale abbondano i sistemi che prendono la forma di reti, per esempio Internet, reti neuronali e metaboliche e molti altri.

Un esempio di rete con otto vertici e dieci connessioni è mostrato nella figura 3.1

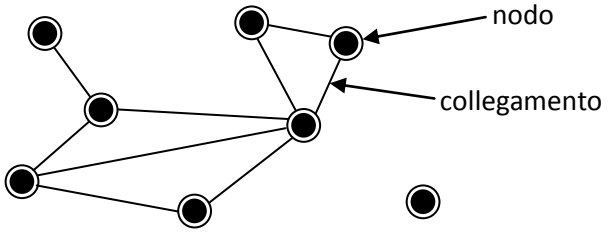


Figura 3.1 Esempio di rete

Una rete può essere rappresentata da un grafo; la teoria dei grafi è una branca della matematica discreta. Spesso si cita come primo approfondimento della teoria delle reti la soluzione del problema del ponte di Könisberg del 1735.

Ci sono varie topologie di reti, per esempio con differenti tipi di nodi, o con differenti tipi di connessioni, e sia i nodi, sia le connessioni possono avere differenti proprietà, come per esempio nei social network.

Anche nella scienza sociologica le reti sono state studiate intensivamente. Già nel 1939 i sociologi dimostrarono l'importanza delle connessioni tra le persone; questi studi consistettero soprattutto nel fare circolare questionari, nei quali si chiedeva di rispondere in dettaglio sulle relazioni con gli altri.

Negli ultimi anni, d'altra parte, c'è stato uno spostamento dell'interesse dall'analisi dei singoli grafi con pochi nodi e poche connessioni alle proprietà statistiche dei grafi di larga scala. Questo nuovo approccio è stato in gran parte dovuto alla disponibilità dei computer e delle reti di comunicazione, che ci permettono di analizzare dati su scala più estesa rispetto al passato, quando gli studi riguardavano reti di dieci o al più cento nodi, mentre adesso non è difficile incontrare reti con milioni o anche miliardi di nodi. Questo cambio di scala modifica anche l'approccio analitico. Molte problematiche, che precedentemente erano state poste negli studi di piccole reti, non sono utilizzabili in molte reti di più larga scala; molte domande alle quali si poteva dare risposta negli studi di piccole reti, non sono interessanti in molte reti più grandi. Una problematica nell'analisi di un social network poteva riguardare la rimozione di quale nodo potesse essere cruciale per la connettività del sistema, ma tale problema è poco significativo nella maggior parte delle reti aventi milioni

[20] Cfr. M.E.J.Newman, *The Structure and Function of Complex Networks*, Siam Review, vol. 45 N°2, pag. 168 e sgg.

di nodi, poiché nessun singolo nodo una volta rimosso ha un grande effetto in tali reti. In questo caso una domanda ragionevole potrebbe riguardare solo quale percentuale di nodi deve essere rimossa per compromettere sostanzialmente la connettività della rete; questo tipo di quesito ha un significato reale nelle reti a larga scala. D'altra parte c'è un'altra ragione molto importante che giustifica il cambio di approccio allo studio delle reti. Per le reti comprese tra dieci e cento nodi, è relativamente facile ottenere un grafo delle reti, con i suoi nodi e le sue connessioni e dare una risposta alle specifiche domande circa la sua struttura esaminando il grafo . Questo è stato uno dei primi metodi di analisi fin da quando è cominciato lo studio delle reti; con una rete di oltre un milione di nodi quest'approccio è inutile. Nel ventesimo secolo il recente sviluppo dei calcoli statistici è stato molto utile per studiare il comportamento dei network complessi, cioè delle reti che hanno un numero diverso di connessioni per ogni nodo (a differenza di quanto avviene per le reti regolari).

Il modello "random" ha dominato per circa quaranta anni sugli studi dei sistemi complessi e ciò è stato dovuto in gran parte all'assenza di un'informazione topologica dettagliata sulle reti reali a larga scala.

I più recenti studi sui circuiti reali complessi[21], supportati dalla possibilità di accedere ad un grande numero di dati sulla topologia complessa delle reti reali, fanno ritenere invece che molti di tali circuiti non sono né completamente regolari né completamente random.

Su questo argomento le due scoperte più recenti sono l'effetto "small-world " e il modello "scale-free" della maggior parte delle reti complesse.

Watts e Strogatz (WS) nel 1998 introdussero il concetto della rete small-world, prendendo un esempio riguardante le relazioni umane; se due persone fanno conoscenza e scoprono di avere amici in comune, entrambi esclamano: " Come è piccolo il mondo!". Nella realtà tutti siamo uniti attraverso una catena di conoscenti.

Lo psicologo Milgram nel lontano 1960, introdusse il concetto dei "sei gradi di separazione", cioè sostenne che negli USA il numero medio dei conoscenti di due persone estranee tra di loro è sei, egli congetturò che questo avviene in tutto il mondo.

Negli ultimi anni è stato dimostrato che questo modello "small-world" è onnipresente in molte reti reali e nei circuiti elettronici.

La caratteristica predominante del grafo ER (modello random) e del modello "small-world" è che la distribuzione della connessione della rete raggiunge un picco ad un valore medio e poi decade esponenzialmente, per cui queste reti sono chiamate "exponential networks" ; esse non sono omogenee, cioè ogni nodo non ha lo stesso numero di connessioni.

Un'altra recente scoperta significativa nel campo delle reti complesse, è l'osservazione che un gran numero di reti a larga scala, incluso Internet, sono "scale-free", cioè la maggior parte dei nodi hanno pochissime connessioni e solo pochi ne hanno molte; è molto importante caratterizzare la

---

[21] Xiao Fan Wang, *Complex networks: topology, dynamics and synchronization*, international journal of bifurcation and Chaos, Vol. 12, N°5 (2002), pag. 885-916

struttura delle reti complesse, poiché questa influenza il comportamento del sistema dinamico, per esempio la connessione "scale-free" rende la rete più tollerante agli errori, ma più vulnerabile agli attacchi esterni, come hanno dimostrato tra gli altri Albert, Callaway, Tu e Choen[22].

Spesso la topologia di legame fra i nodi della rete gioca un ruolo cruciale nel determinare il suo comportamento dinamico; per esempio, a dispetto dell'evidenza, non solo un accoppiamento forte porterà alla sincronizzazione una rete di circuiti identici, come sostenuto da Wu e Chua nel 1995, ma anche molte reti con accoppiamenti relativamente deboli di tipo "real-world" portano i nodi della rete alla sincronizzazione.

3.2 Proprietà dei grafi

Tra tutte le quantità e le misure studiate negli ultimi anni, riguardanti i networks complessi, sono tre i parametri che giocano un ruolo chiave nella teoria e nella modellizzazione di tali reti: la lunghezza media del percorso, il coefficiente di clustering, e la probabilità che la rete abbia un determinato grado medio (distribuzione del grado).

- **Lunghezza media del percorso**

In una rete la distanza d_{ij} tra il nodo i ed il nodo j, è definita come il numero di connessioni lungo il più breve percorso, che connette i due nodi

Il diametro D di una rete è la massima distanza tra un paio di questi nodi.

La lunghezza media si definisce come la media delle distanze tra due nodi; tale distanza determina l'effettiva misura della rete. Nel network friendship, L è il numero medio di legami nella catena più piccola che connette due nodi. Chiaramente $L/D \leq 1$; se $L/D=1$ la rete è globally-coupled, cioè, come abbiamo visto nel capitolo precedente, ogni nodo è connesso con tutti gli altri.

Una delle più importanti scoperte nel campo delle reti complesse, è che le lunghezze medie del percorso della maggior parte delle reti reali a larga scala sono relativamente piccole, anche se hanno un numero di connessioni inferiore rispetto alle reti di tipo globally coupled con lo stesso numero di nodi; si verifica così il cosiddetto small-world-effect .

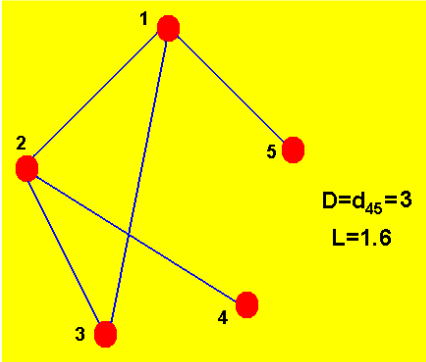


Figura 3.2 Diametro e lunghezza media di path per una rete con cinque nodi e cinque collegamenti

[22] Albert & Barabasi A.-L. [2000], *Topology of complex networks: Local events and universality,* Phys. Rev. Let. 85(24), 5234, 5237
Callaway, Newman, Strogatz, Watts[2000], *Network, robustness and fragility:* Percolation on random graphs, Phys. Rev. Let. 85(25), 5468, 5471
Tu Y [2000], *How robust is internet?,* Nature 406, 353-354
Cohen e altri, 2000-2001

- **Il coefficiente di clustering**

In una rete di amicizie è possibile che gli amici degli amici siano anche amici diretti, o per dirla in un altro modo, due degli amici di una persona, siano anche amici l'un l'altro. Questa proprietà è chiamata clustering.

Supponiamo che un nodo i della rete ha K_i legami che lo connettono agli altri nodi k_i . Al massimo possono esistere $N(N\text{-}1)/2$ legami fra tutti i nodi, e questo accade solo se ogni nodo è connesso a tutti gli altri. Il coefficiente di clustering C_i è il rapporto tra il numero k_i di connessioni di un nodo e il numero totale $N(N\text{-}1)/2$ di connessioni possibili:

$$C_i = \frac{2ki}{N(N-1)}$$

Il coefficiente di clustering C dell'intera rete è la media di tutti i C_i . Chiaramente $C \leq 1$; se $C=1$ la rete è globally-coupled.

In un network completamente random $C \sim N^{-1}$, ed è stato provato che le reti reali a larga scala hanno una tendenza ad avere $C > N^{-1}$, anche se significativamente minore di uno.

Un altro modo[23] per definire il coefficiente di clustering riguarda la proprietà transitiva del network; si è trovato che se il nodo A è connesso con il nodo B ed il nodo B al nodo C, c'è un'alta probabilità che il nodo A si connetta al nodo C; nel linguaggio dei social networks, l'amico del tuo amico è come se fosse anche tuo amico. In termini di topologia delle reti, la transitività implica la presenza di un alto numero di triangoli nel network, cioè insiemi di tre vertici, ognuno dei quali è connesso agli altri (vedi figura 3.3).

La definizione di coefficiente di clustering diventa allora:

$$C = \frac{3 \text{ x numero di triangoli del network}}{\text{numero di triple connesse di nodi}} \quad (1)$$

C misura la frazione di triple mancanti di una sola connessione per completare il triangolo. Il fattore tre nel numeratore è utilizzato poiché ogni triangolo contribuisce a tre triple, e assicura che C appartenga ad un intervallo $0 \leq C \leq 1$. In termini semplici C indica la probabilità che due nodi, che sono i vicini di uno stesso nodo, diventeranno essi stessi vicini.

Un'altra definizione di coefficiente di clustering è:

$$C = \frac{6 \text{ x numero di triangoli del network}}{\text{numero di percorsi di lunghezza 2}} \quad (2)$$

[23] Cfr. M.E.J.Newman, *The Structure and Function of Complex Networks*, Siam Review, vol 45 N°2, pag. 183

Questa definizione è stata molto utilizzata negli studi sociologici, fisici e matematici (Barrat e Weight).

Una definizione alternativa molto usata è stata data da Watts e Strogatz:

$$C_i = \frac{\text{numero di triangoli connessi al nodo i}}{\text{numero di triple centrate nel vertice i}} \quad (3)$$

Per i vertici con grado 0 oppure 1, per i quali sia il numeratore che il denominatore sono uguali a zero, poniamo C_i=0. Il coefficiente di clustering per l'intera rete è la media:

$$C = \frac{1}{n}\sum_i C_i \quad (4)$$

Questa definizione tiene in maggior conto i nodi con basso grado.

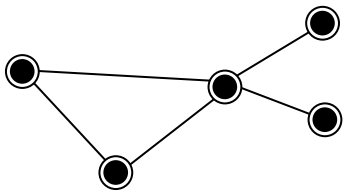


Figura 3.3 Illustrazione della definizione del coefficiente di clustering C, calcolato con la (1). Il network ha un triangolo e otto triple connesse e quindi il coefficiente di clustering (1) è C=3X1/8=3/8. I nodi individuali hanno coefficiente di clustering(3) è C_i= 1, 1, 1/6, 0, 0 , di conseguenza il valore medio del coefficiente di clustering (4) è C=13/30

Come si può vedere il coefficiente di clustering calcolato con la (1) è diverso dal coefficiente calcolato con la (4). Normalmente l'equazione (1) è più facile da calcolare analiticamente, ma la (4) si calcola efficacemente con il computer ed ha trovato un grande utilizzo in studi numerici ed analisi di dati, per cui è importante chiarire quale definizione del coefficiente di clustering è stata usata negli articoli che trattano questo argomento.

- **Il grado medio della rete.**

La caratteristica più semplice e più studiata di ogni nodo è il suo grado.Il grado k_i di un nodo i è il numero totale delle sue connessioni. Il grado medio della rete, $\langle k \rangle$, è la media dei gradi k_i .

- **La funzione di distribuzione del grado**

La probabilità che un nodo, scelto a caso, abbia esattamente k connessioni è caratterizzata dalla funzione di distribuzione $P(k)$. Tutti i nodi di una rete regolare hanno lo stesso numero di collegamenti, e così il diagramma della distribuzione di grado contiene un solo picco; al contrario, ogni

casualità nella rete amplierà il numero di picchi. Nel caso limite di una rete completamente random, la sequenza dei gradi obbedisce alla distribuzione di Poisson, con

$$P(k) \simeq e^{-\langle k \rangle} \frac{\langle k \rangle^k}{k!}$$

La figura della distribuzione di Poisson, decade esponenzialmente dal valore di picco $\langle k \rangle$. A causa di questa decadenza la probabilità di trovare un nodo con k connessioni, diventa molto piccola per $k \gg \langle k \rangle$, dove $\langle k \rangle$ = valore di picco di k.

Negli ultimi anni, un gran numero di risultati empirici, hanno dimostrato che per i network a grande scala, per esempio internet e WWW, la probabilità del grado di connessione si allontana significativamente dalla funzione di probabilità di Poisson, ed è più accuratamente descritta dall'equazione:

$$P(k) \sim k^{-\gamma}$$

Questa funzione di probabilità decade più gradualmente di una funzione esponenziale, permettendo a pochi nodi un grado di connessione molto elevato, per esempio supponendo che c'è una rete con $N=100\ 000$ nodi, che verifica la precedente funzione di probabilità con esponente due, allora un nono dei nodi, circa 11 111 saranno connessi tre volte ed un decimillesimo cioè dieci nodi, avranno cento connessioni; una rete con una distribuzione di tipo power-law, è chiamata scale-free network.

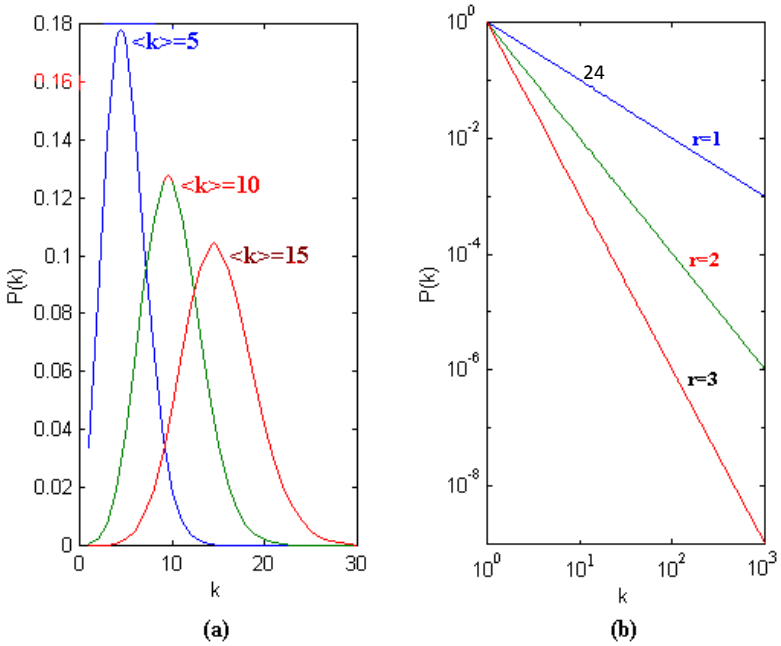


Figura 3.4 (a) Distribuzione del grado gaussiana con $\langle k \rangle$ =5 , 10 e 15

(b) Distribuzione di grado di tipo power-law con γ=1, 2 e 3 rispettivamente

[24] Figura tratta da Xiao Fan Wang, *Complex networks: topology, dynamics and synchronization*, international journal of bifurcation and Chaos, Vol. 12, N°5 (2002), pag.889

3.3 Topologia delle reti complesse

3.3.1Reti regolari

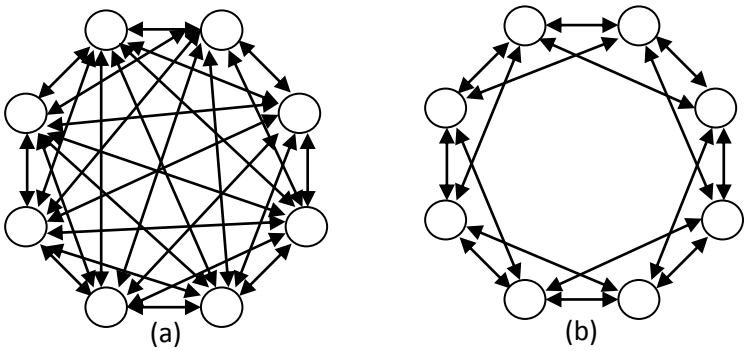


Figura 3.5 Due esempi di network regolari, a)globally coupled b)rete regolare, i cui nodi hanno tutti lo stesso grado.

Per capire la struttura di una rete, misuriamo in primo luogo i tre parametri:

lunghezza media del percorso L, coefficiente di clustering C, funzione di probabilità $P(k)$.

È noto che un network globally coupled, ha la più piccola lunghezza media di percorso $L_{gc}=1$ e il più grande coefficiente di clustering $C_{gc}=1$. Sebbene i modelli di rete globally-coupled comprendano le proprietà di small-world e di clustering di molti network reali, sono evidenti i suoi limiti: una rete di questo tipo con N nodi ha $N(N-1)/2$ connessioni, mentre la maggior parte dei network reali a larga scala, tende ad avere generalmente connessioni di ordine N piuttosto che N^2.

Un grafo accuratamente studiato, regolare e con poche connessioni, è il network nearest-neighbor, in cui come abbiamo visto precedentemente ogni nodo è connesso solo con pochi nodi vicini.

Il termine rete può dare l'impressione di una griglia quadrata bidirezionale, ma attualmente può avere altre geometrie. Una rete minima è una struttura monodimensionale come una riga di persone che si danno la mano; una rete nearest-neighbor consiste in N nodi che formano un anello dove ogni nodo i è adiacente ai nodi vicini, $i\pm1$, $i\pm2$, $i\pm3$, ……. , $i\pm K/2$, con K pari. Per valori di K elevati, la rete nearest-neighbor coupled ha molte connessioni, il coefficiente di clustering è:

$$C_{nc} = \frac{3(K-2)}{4K} \sim \frac{3}{4}$$

Al contrario

$$L_{nc} = \frac{N}{2K} \to \infty \quad \text{con } N \to \infty$$

$$P(k) = \begin{cases} 1, & k = K \\ 0, & k \neq K \end{cases}$$

Ci chiediamo se esista una rete, che contemporaneamente sia di tipo “sparse” e “clustered” ed abbia una piccola lunghezza media del percorso. Un semplice esempio di tale rete è il network di tipo star coupled, nel quale c’è un nodo centrale ed ognuno degli altri N-1 nodi è connesso solo a questo.

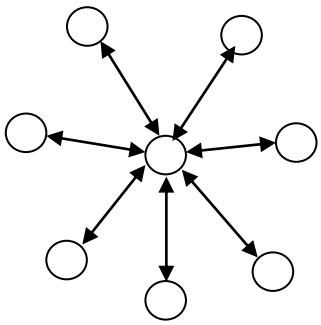


Figura 3.6 Star coupled network

Per questo tipo di collegamento:

$$L_{star} = 2 - \frac{2(N-1)}{N(N-1)} \rightarrow 2 \ \text{ per } \ N \rightarrow \infty$$

$$C_{star} = \frac{N-1}{N} \rightarrow 1 \ \ \text{per } N \rightarrow \infty$$

Il modello star è utilizzabile per reti di tipo sparse, clustering e small- world, oltre che per molti altre reti reali, meglio di quelle regolari.

3.3.2. Reti probabilistiche

Dalla parte opposta rispetto alle reti con strutture completamente regolari, ci sono quelle con strutture completamente random, che furono studiate per la prima volta circa quaranta anni fa da Erdös e Rényi.

Il grado medio di un grafo random è:

$$\langle k \rangle = p(N-1) \simeq pN$$

Sia L_{rand} la lunghezza media del percorso di una rete random. Intuitivamente circa $\langle k \rangle^{L_{rand}}$ nodi della rete, sono ad una distanza L_{rand} o ancora più vicini. Ne consegue che $N \sim \langle k \rangle^{L_{rand}}$ per cui:

$$L_{rand} \sim \frac{In(N)}{In(\langle k \rangle)}$$

Questo incremento logaritmico è tipico dello small word effect.

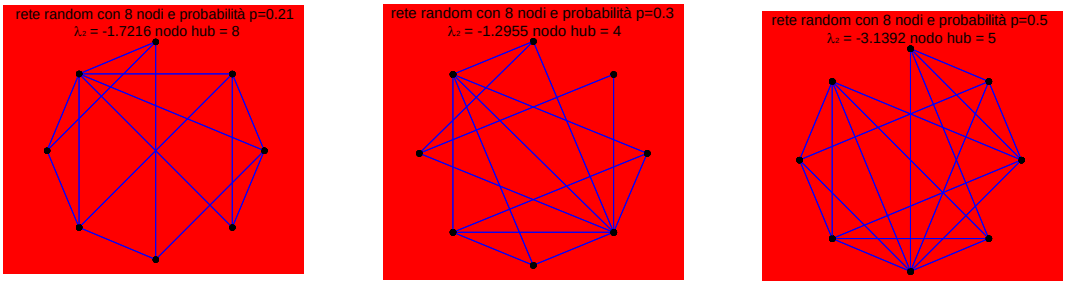
Poiché $\log N$ cresce lentamente al crescere di N, la lunghezza media del path può essere molto piccola anche in un sistema con molti nodi.

Il coefficiente di clustering del modello ER è:

$$C = p = \frac{\langle k \rangle}{N} \ll 1$$

Questo significa che un network o una rete random a larga scala, non può essere "clustered".

Per un grande N l'algoritmo ER genera un network omogeneo, la cui connettività segue approssimativamente la distribuzione di Poisson descritta precedentemente. D'altra parte la probabilità del grado dei networks reali devia significativamente dalla distribuzione di Poisson del grafo random. La distribuzione del grado dei networks reali si può ottenere costruendo grafi random, con determinate probabilità, i cosiddetti Generalized Random Graph:



(a) (b) (c)

Figure 3.7 Evoluzione di un grafo random: dati otto nodi isolati si connette ogni paio di nodi con probabilità a)0.21 b)0.3 c)0.5

- **Reti di tipo small-world**

Una rete regolare è "clustered", ma non può avere l'effetto small-world.

D'altra parte il grafo random, mostra l'effetto small-world, ma non quello di clustering; non sorprende che il modello di reti regolari ed il modello di reti random falliscono entrambi nel riprodurre alcuni importanti aspetti dei network reali, poiché questi non sono né interamente random, né interamente regolari.

Allo scopo di descrivere il passaggio da una rete regolare ad una random, Watts e Strogatz nel 1998 introdussero un interessante modello di network small-world (WS), che descriviamo come segue:

1) Inizio con una rete regolare.

   Si inizia con un network nearest-neighbor coupled che ha N nodi costituenti un anello, dove ogni nodo i è adiacente ai suoi nodi vicini $i \pm 1$, $i \pm 2$, ... , $i \pm K/2$, dove K è un numero pari. Per avere un network "sparse", ma connesso in tutti i modi, consideriamo $N >> K >> \ln(N) >> 1$

2) Rendere random la rete.

   Questo richiede che ogni collegamento della rete abbia probabilità p . Si sostituisce il collegamento tra due nodi della rete nearest-neighbor e si connette l'estremo libero ad un no-

do scelto a caso in tutto il network, con il vincolo che due differenti nodi non possono avere più di una connessione tra di loro, e nessun nodo può essere connesso con se stesso. Questo processo introduce $pNK/2$ collegamenti, che connettono nodi, che erano connessi diversamente. Facendo variare la p, la transizione tra ordine (p=0) e casualità $(p=1)$ può essere strettamente monitorata.

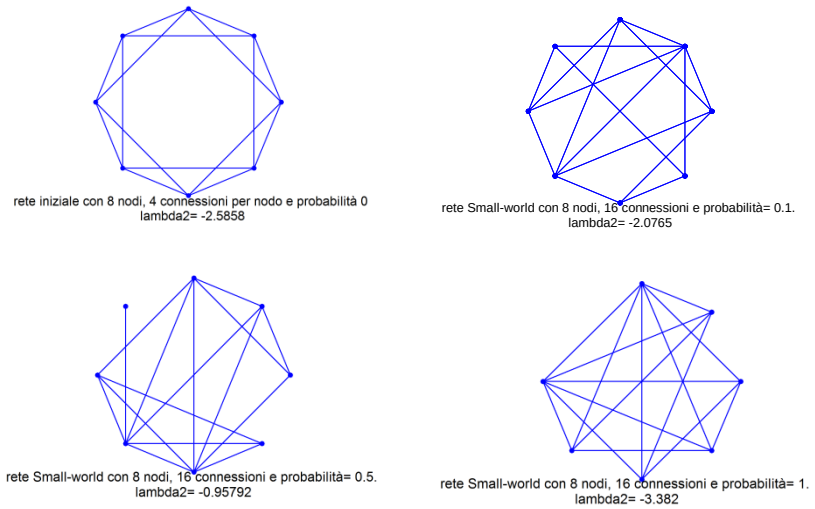


Figure 3.8 Il modello Watts-Strogatz nel quale i collegamenti tra due nodi presi a caso si sostituiscono ad un collegamento precedente

Questa ricerca ha stimolato molti studiosi a trovare delle varianti del modello WS, fra cui quella di Newman e Watts, nota come NW small-world model.

Nel modello NW non si deve eliminare nessuna connessione tra due nearest-neighbor, ma aggiungere una connessione con probabilità p fra due nodi scelti a caso.

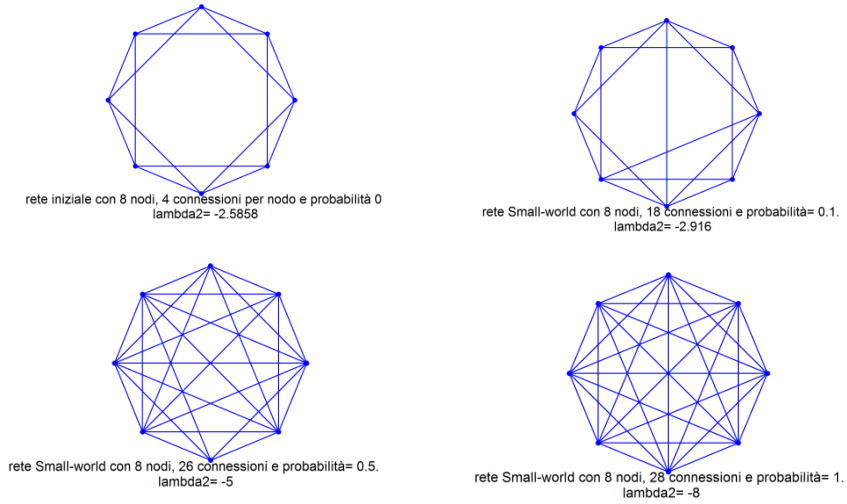


Figure 3.9 Il modello Newmann-Watts nel quale si aggiungono i collegamenti tra due nodi presi a caso.

Comunque, anche in questo caso, nessun nodo può essere collegato a se stesso per più di una volta, o collegato più di una volta ad un altro nodo. Per $p=0$ il modello NW si riduce al network nearest-neighbor coupled; per $p=1$ diventa un globally coupled network.

Il modello NW è più facile da analizzare del modello WS, poiché non permette la formazione di nodi isolati, che invece possono trovarsi nel modello WS. I due modelli sono equivalenti se la probabilità p è sufficientemente piccola e la rete ha un grande numero di nodi; per entrambi i metodi il modello di riferimento è lo small-world.

Nel modello WS per $p=0$ ogni nodo ha lo stesso grado k, se $p\neq 0$ si introduce disordine nel network, ma il grado medio è uguale a k, questo significa che il modello small-world può essere visto come un modello omogeneo in cui ogni nodo ha lo stesso numero di collegamenti.

Watts e Strogatz[25] studiarono in particolare il comportamento del coefficiente di clustering $C(p)$ e della lunghezza media del percorso $L(p)$, parametri entrambi in funzione della probabilità p. Una rete regolare ad anello $(p=0)$ ha un alto coefficiente di clustering $C(0)\simeq 3/4$, ma anche una grande lunghezza media del percorso $L(0)\simeq N/(2K)\gg 1$. I due ricercatori trovarono che per una piccola probabilità, quando le proprietà locali della rete sono ancora quasi le stesse di quelle della rete originale regolare, e quando il coefficiente di clustering non differisce sostanzialmente dal suo iniziale valore $(C(p)\sim C(0))$, la lunghezza media del percorso decresce rapidamente, ed è dello stesso ordine di quella delle reti random $(L(p)\gg L(0))$ (vedi figura 3.10)[26]. Questo risultato è in effetti naturale; intuitivamente è sufficiente aggiungere alcuni collegamenti di tipo random, perché la lunghezza media del percorso decresca significativamente. D'altra parte, l'aggiunta di molte connessioni non può cambiare in modo cruciale le proprietà di clustering locali della rete. La lunghezza media del percorso di una rete small-world obbedisce alla formula generale (Barrat & Weight, 2000):

$$L_{SW}(N,p)\sim\frac{N}{K}f\left(pKN^{d}\right)$$

che soddisfa:

$$f(u)=\begin{cases}\text{costante} & \text{se } u\ll 1\\ \dfrac{\ln(u)}{u} & \text{se } u\gg 1\end{cases}$$

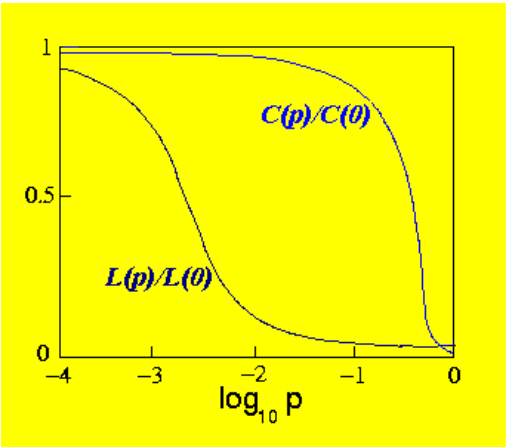


Figura 3.10 lunghezza media del percorso e coefficiente di clustering del modello small-world WS, come funzioni della probabilità di connessione p. entrambi sono normalizzati al loro valore iniziale per p=0

[25] Xiao Fan Wang, *Complex networks: topology, dynamics and synchronization*, international journal of bifurcation and Chaos, Vol. 12, N°5 (2002), pag.897

[26] Figura tratta di Xiao Fan Wang, *op. cit.*

Nel modello WS per p=0 ogni nodo ha lo stesso grado, di modo che la distribuzione di grado è una funzione delta, centrata in k. Una probabilità $p\neq 0$ introduce disordine nella rete, modificando la distribuzione di grado, affinché il grado medio rimanga uguale a k, in tal caso si è trovato che un grande valore di N, la distribuzione di grado dei modelli small-world è:

$$P(k) = \sum_{n=0}^{f(k,K)} C_{\frac{K}{2}}^{n}(1-p)^{N}\, p^{\left(\frac{k}{2}-n\right)} \times \frac{\left(\frac{pK}{2}\right)^{\left(k-\frac{K}{2}-n\right)}}{\left(k-\frac{K}{2}-n\right)!} e^{-\frac{pK}{2}}$$

per $k\geq K/2$ e $f(k,K)=\min(k-K/2,K/2)$. La forma della distribuzione di grado di una rete small-world, è simile a quella della distribuzione di grado di Poisson: ha un picco per $\langle k\rangle = K$ e decade esponenzialmente per un elevato valore di k. Come già detto, questo significa che il modello small-world può anche essere visto come una rete regolare, nella quale tutti i nodi hanno approssimativamente lo stesso numero di connessioni.

- **Reti di tipo scale-free**

Per spiegare l'origine della probabilità di grado del tipo power-law, Barabasi ed Albert proposero un nuovo modello di rete, prendendo spunto dal fatto che in alcuni modelli si trascurano importanti caratteristiche. In primo luogo gran parte delle reti reali sono aperte, e prendono forma aggiungendo in continuazione nuovi nodi al sistema, ma molti altri modelli sono statici: anche se i collegamenti possono essere aggiunti o sostituiti, il numero di nodi non cambia mai. In secondo luogo sia i grafi random, sia i modelli small-world assumono probabilità costante quando si creano nuovi collegamenti, ma questo non è realistico in nessuno dei due casi.

Barabasi ed Albert, ritennero che le reti crescono continuamente con l'aggiunta di nuovi nodi e i nuovi nodi sono di preferenza collegati ai nodi esistenti che hanno già un gran numero di connessioni ("rich gets richer"). L'algoritmo del modello BA scale-free è il seguente, basato su queste due condizioni:

1) Crescita: si comincia con un piccolo numero (m_o) di nodi, in ogni intervallo di tempo si introduce un nuovo nodo e si connette agli m $(\leq m_o)$ esistenti
2) Collegamento preferenziale: la probabilità P_i che il nuovo nodo si connetta al nodo i dipende dal grado k_i di tale nodo, in modo che :

$$P_i = \frac{k_i}{\sum_j k_j}$$

Dopo t intervalli di tempo, da questo algoritmo si ricavano in un network $N=t+m_o$ nodi e mt collegamenti. Se la rete si ingrandisce seguendo queste due leggi, la forma della distribuzione di grado non cambia con il tempo. Tale distribuzione è descritta con una power-law con esponente 3:

$$P(k)\sim 2m^{\frac{1}{\beta}}k^{-\gamma} \quad \text{con } \gamma = \frac{1}{\beta}+1 = 3$$

In altre parole la probabilità di trovare un nodo con k collegamenti è proporzionale al termine k^{-3}. I risultati numerici indicano che, comparato al grafo random con la stessa misura e grado medio, la

lunghezza media del percorso di un modello scale-free è più piccola ed il coefficiente di clustering è molto più alto, questo significa l'esistenza di pochi nodi con numero di connessioni molto alto.

Il BA scale-free model è basato sulla crescita lineare e sul collegamento preferenziale ed ha una distribuzione di grado di tipo potenza con esponente uguale a -3, e più esattamente per le reti reali compreso tra -3 e -1.

Per alcune reti reali la distribuzione di grado può anche non essere una potenza ad esponente negativo, ma un esponenziale. Recentemente sono stati introdotti alcuni nuovi modelli scale-free anche da Albert e Barabasi

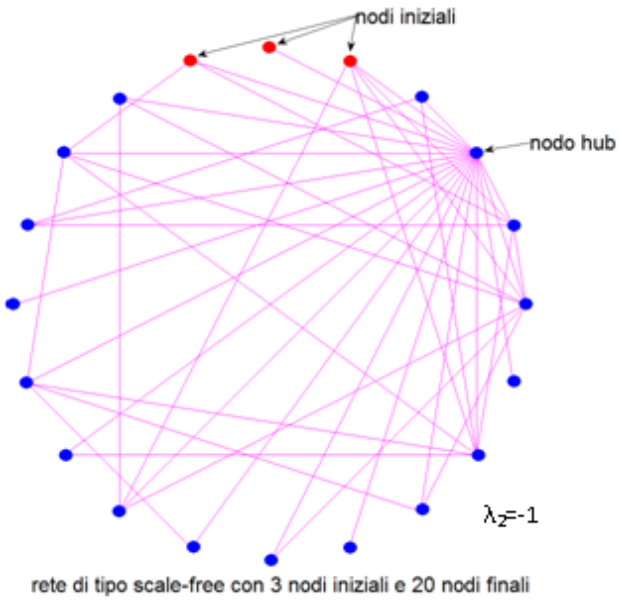


Figura 3.11 evoluzione del modello scale-free di Barabasi-Albert
m_o è il numero di nodi iniziali (in rosso) f è il numero di nodi finali (in blu)

- **Altri modelli di reti**

Modello weighted-evolving

Alcuni sistemi biologici, ecologici ed economici, sono ben descritti da questo tipo di modello[27], introdotto recentemente da Yook ed altri nel 2001. In questo caso i nodi interagiscono l'uno con l'altro attraverso legami di accoppiamento variabili. Si comincia con un piccolo numero (m_o) di nodi, ad ogni intervallo di tempo si introduce un nuovo nodo e lo si connette ad m ($\leq m_o$) nodi già esistenti; la probabilità che un nuovo nodo j sia connesso con un già esistente nodo i è data da:

$$P_i = \frac{k_i}{\sum_j k_j}$$

Assegnando un peso al nuovo legame tra j ed i, si assume che il peso $w_{ji}(=w_{ij})$ sia proporzionale al grado k_i del nodo i, cioè il nodo con maggiori connessioni ha il più grande peso.

Poniamo per ipotesi :

[27] Xiao Fan Wang, *op. cit.* pag. 900

$$\sum_{\{i'\}} w_{ji'} = 1$$

Dove i' rappresenta la somma di tutti i nodi esistenti ai quali è connesso il nuovo nodo j.

Ad ogni legame i ↔ j è assegnato il peso:

$$w_{ji} = \frac{k_i}{\sum_{i'} k_{i'}} \quad (1)$$

In conclusione, il modello WE si definisce come segue: ad ogni intervallo di tempo si aggiunge un nuovo nodo con m legami, il quale è connesso con uguale probabilità ai nodi già esistenti sul sistema; i pesi dei legami sono assegnati dall'equazione (1).

Modelli deterministici

Le reti di tipo small-world e scale-free presentano sempre qualche grado di casualità. Recentemente ha destato molto interesse lo studio di modelli deterministici di tale tipo. Fra gli altri Barabasi e Ravastz nel 2001 hanno presentato un modello semplice che può generare una rete deterministica di tipo scale-free, usando una costruzione gerarchica.

Ad ogni step n, si aggiungono due unità di 3^{n-1} nodi, che sono identici alla rete creata nello step precedente $(n-1)$ ed ogni nodo di queste due unità è connesso alla radice della rete $(n=0)$. La distribuzione di grado per questo network deterministico di tipo scale-free è:[28]

$$P(k) \sim k^{-\frac{\ln 2}{\ln 3}}$$

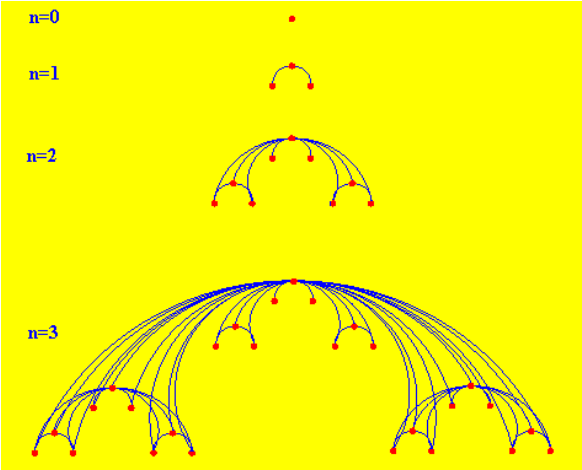


Figura 3.12 Una costruzione gerarchica di un network deterministico scale-free (Barabasi & Ravasz, 2001)

3.4 Connettività delle reti complesse: roustezza e fragilità

Le recenti scoperte nel campo delle reti complesse, indicano che la topologia della rete gioca un ruolo cruciale sui possibili guasti di questa.

[28] Figura tratta da Xiao Fan Wang, op.cit. fig.13 pag. 901

Se ad ogni intervallo di tempo viene rimosso un nodo di una rete completamente connessa, la scomparsa di tale nodo implica la rimozione di tutte le sue connessioni. Se c'erano molte connessioni tra i due nodi i e j, la distruzione di una di quelle fa aumentare la distanza d_{ij} tra di essi, e questo può causare anche un aumento della lunghezza media del percorso dell'intera rete. Nel caso peggiore, quando inizialmente c'era un solo collegamento tra i e j, la distruzione di uno di questi due nodi fa sì che i due nodi diventino isolati l'uno dall'altro. La rete si può considerare robusta, (o tollerante agli errori), se la maggior parte dei nodi resta collegata, anche se una piccola frazione di essi è stata rimossa. Albert e altri nel 2000 paragonarono la robustezza del grafo random ER, che ha una distribuzione di grado esponenziale ed il modello di Barabasi scale-free, che ha una distribuzione di grado power-law. Si è discusso circa due tipi di guasti: quelli dovuti alla casualità furono modellati rimuovendo una frazione di nodi scelti a caso, quelli dovuti ad attacchi intenzionali (hakers), furono descritti rimuovendo i nodi con il più alto numero di connessioni. Gli autori sopracitati sono giunti alla conclusione che i networks scale-free mostrano un'elevata robustezza di fronte ai guasti di tipo casuale, poiché la distribuzione power-law implica che solo pochi nodi hanno un gran numero di connessioni, mentre la maggior parte di essi ha solo poche connessioni, cosicché la probabilità che sia colpito uno di questi ultimi è molto alta, con minori conseguenze negative sulla rete. Al contrario, questo tipo di rete è molto più colpito dagli attacchi intenzionali, poiché quando sono colpiti i nodi con un grande numero di connessioni la rete non funziona più, molto più rapidamente che nel caso random (vedi figura3.13).[29]

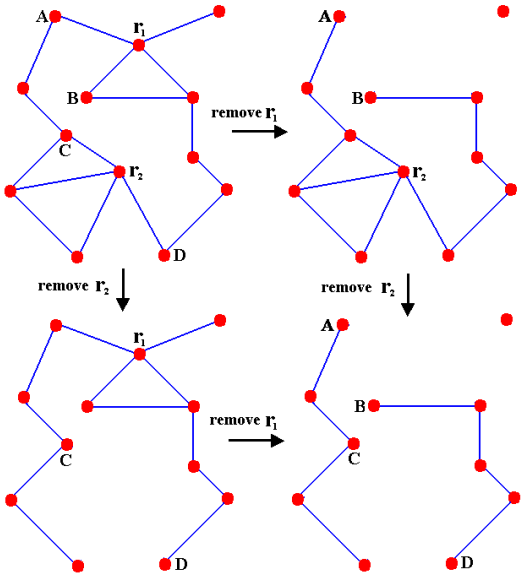


Figura3.13 Effetti della rimozione di un nodo in un network inizialmente connesso:in tale stato d_{AB}= d_{CD}=2. Dopo la rimozione del nodo r_1, d_{AB} =8 , dopo la rimozione del nodo r_2, d_{CD} =7. Dopo la rimozione di entrambi i nodi il network si divide in tre gruppi isolati.

Inoltre consideriamo tutti i nodi, collegati ad altri nodi, suscettibili di infezioni (i cosiddetti virus) attraverso il collegamento. Se chiamiamo ν la probabilità dei nodi suscettibili di infezioni di diventare infetti, e δ la probabilità dei nodi infetti curati di diventare di nuovo suscettibili di infe-

[29] Figura tratta da Xiao Fan Wang, *International Journal of Bifurcation and Chaos*, Vol. 12 N° 5(2002) fig.14 pag. 902

zioni, consideriamo $\lambda = \frac{\nu}{\delta}$ e $\lambda_c > 0$ il valore di soglia di λ ; se nella rete $\lambda > \lambda_c$ l'infezione si propaga e persiste nel tempo, se $\lambda < \lambda_c$ l'infezione decade velocemente in modo esponenziale. Nelle reti scale-free, a differenza che in quelle di tipo random, $\lambda_c = 0$, ma per questo tipo di reti una grande percentuale di nodi ha $\lambda \ll 1$.

3.5 Sincronizzazione di reti complesse

La sincronizzazione, come abbiamo già visto nel primo capitolo, è un processo dove alcuni sistemi (equivalenti o non equivalenti), modificano il loro comportamento sia a causa di una successiva diversa configurazione di accoppiamento, sia a causa di un forzamento esterno. L' origine di tale parola deriva dal greco , e significa "condividere il tempo". Inizialmente l'attenzione degli studiosi è stata rivolta essenzialmente ai sistemi periodici, mentre recentemente si è spostata sui sistemi caotici. Quando questi sono accoppiati, possono avere luogo diversi tipi di sincronizzazione[30]:

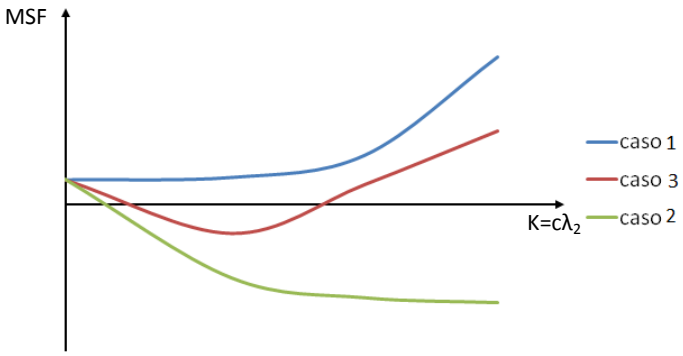
1) Sincronizzazione completa o identica, è la più semplice forma di sincronizzazione e consiste in un perfetto aggancio delle traiettorie di sistemi caotici identici nel corso del tempo.
2) Sincronizzazione di fase, si ha quando coppie di oscillatori non identici a regime raggiungono la sincronizzazione della fase, anche senza la sincronizzazione dell'ampiezza.
3) Sincronizzazione ritardata, implica un ritardo tra l'output di un sistema e l'output di un altro.
4) Sincronizzazione generalizzata, considera sistemi differenti ed associa l'uscita di un sistema all'output di un altro sistema.
5) Ritardo intermittente, cioè il fenomeno della sincronizzazione ritardata può essere interrotto per brevi periodi.
6) Sincronizzazione imperfetta di fase, quando la sincronizzazione di fase viene interrotta per brevi periodi.
7) Altri tipi.

Gli studi più recenti hanno riguardato i sistemi estesi nello spazio o di dimensioni infinite, per studiare meccanismi che riguardano la desincronizzazione e per definire un approccio formale uniforme, che si possa applicare a differenti tipi di sincronizzazione; a tale riguardo è stato studiato il comportamento di reti complesse con diverse topologie di accoppiamento, in particolare di tipo small-world e scale-free. Il metodo chiamato Master Stability Function (vedi cap. 2 par. 2.2) , inizialmente introdotto per reti di oscillatori accoppiati, è stato più tardi esteso al caso di reti complesse di sistemi dinamici, accoppiate con topologie arbitrarie. Ricordando che la MSF nella terminologia dei sistemi dinamici, rappresenta il più grande esponente di Lyapunov, e viene valutata graficamente in funzione del parametro $k=c\lambda_2$ dove c è la costante di accoppiamento, condizione necessaria per ottenere la sincronizzazione è che la MSF sia negativa al crescere del parametro k.

Si possono distinguere tre tipologie di MSF, in cui per $k=0$ la MSF assume valori positivi:

[30] S. Boccaletti, V. Latorab, Y. Moreno, e M. Chavez , D.-U. Hwanga , *Complex networks: Structure and dynamics,* Physics Reports 424 (2006), 175 – 308, pag. 237

1) La MSF è monotonicamente crescente ed inizialmente positiva.
2) La MSF è monotonicamente decrescente, inizialmente positiva e diventa negativa in corrispondenza di un determinato valore di k.
3) La MSF è inizialmente positiva, diventa negativa in corrispondenza di un determinato valore k_a per poi ridiventare positiva per un determinato valore k_b, con $k_b > k_a$.



E' evidente che per gli ultimi due casi la soluzione è banale, cioè si ha sincronizzazione nel secondo caso a partire da un valore di k per cui la funzione si annulla e nel terzo caso per $k \in [k_a, k_b]$. Una soluzione non banale nel secondo caso, che è il più ricorrente, si ha quando la funzione $F(x)$ è periodica, ed è periodico l'intervallo $[k_a, k_b]$ in cui la MSF si annulla, per cui la condizione di stabilità[31] sarà sempre soddisfatta ogniqualvolta $\lambda_N / \lambda_2 < k_b / k_a$.. essendo, come già noto, λ_2 il maggiore autovalore della matrice di connessione, escluso lo zero, e λ_N il minore (e viceversa, se si considerano i valori assoluti dei due autovalori). Nel primo caso,invece,si deve dire che non si avrà mai una sincronizzazione stabile. Si vuole inoltre distinguere il caso dell'accoppiamento simmetrico, in cui la matrice di connessione A_c ha solamente autovalori reali, che possono essere ordinati come $0 = \lambda_1 \geq \lambda_2 \geq \ldots\ldots\ldots \geq \lambda_N$, da una configurazione in cui la matrice di connessione è asimmetrica e diagonalizzabile, nella quale alcuni autovalori possono essere complessi coniugati a coppie.

Nei paragrafi seguenti si tratterà della sincronizzazione di reti non omogenee, in particolare delle reti di tipo scale-free e di quelle di tipo random.

- **Sincronizzazione delle reti di tipo small-world**

Si può costruire la matrice di connessione di un network dinamico con connessioni di tipo NW small-world (Wang & Cheng, 2001). Nella matrice di connessione nearest-neighbor A_{nc}, se $A_{ij} = A_{ji} = 0$ si pone $A_{ij} = A_{ji} = 1$ con probabilità p, poi si utilizza il solito metodo per gli elementi della diagonale principale. Chiamiamo questa matrice A_{SW} e sia λ_{2SW} il suo autovalore maggiore e diverso da zero, per cui se:

$$c \geq \left| \frac{\bar{d}}{\lambda_{2SW}} \right|$$

[31] S. Boccaletti, V. Latorab, Y. Moreno, e M. Chavez , D.-U. Hwanga , *Complex networks: Structure and dynamics,* Physics Reports 424 (2006), 175 – 308, pag. 240

la corrispondente rete small-world sarà sincronizzata (vedi figure 3.14 e 3.15)[32].

Ricordiamo che mentre per le reti di tipo globally connected è sufficiente per ogni N che il coefficiente di accoppiamento c sia maggiore di zero, per le altre reti esiste un fattore di accoppiamento di soglia $\bar{c}=|\bar{d}|>0$ per cui la rete sarà sincronizzata se $c > \bar{c}$

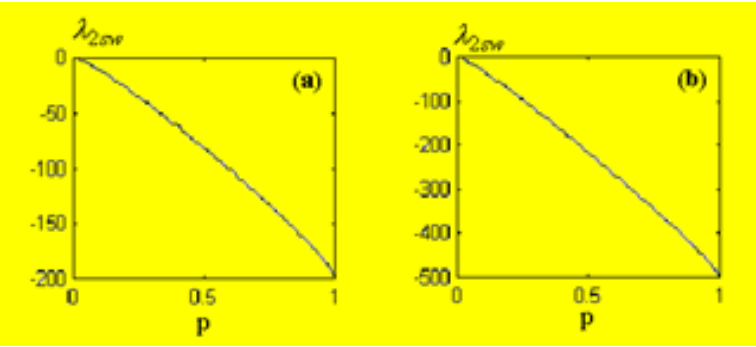


Figura3.14 Valori di λ_{2SW} (p,N) come funzione della probabilità p: (a) N=200 ; (b) N=500

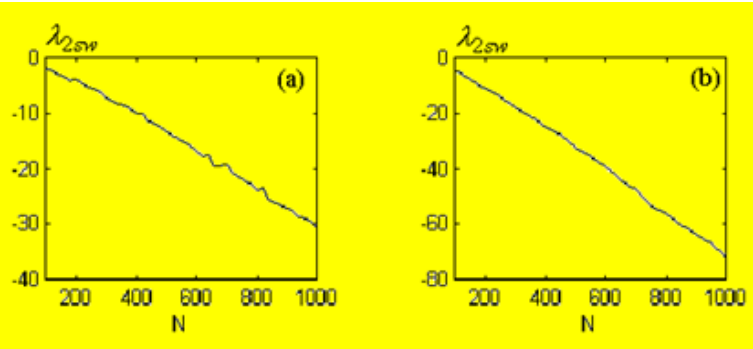


Figura 3.15 Valori di λ_{2SW} (p,N) come funzione di N: (a) p=0.05 ; (b) p=0.1

Si può vedere che:

1)Per ogni valore di N , $\lambda_{2SW}(p,N)$ decresce a -N per p che aumenta da 0 a 1

2) Per ogni valore di p ϵ (0,1], $\lambda_{2SW}(p,N)$ decresce a -∞ per $N \to \infty$

I risultati ottenuti implicano che per ogni tipo di accoppiamento c > 0

3) Per ogni $N > \frac{|\bar{d}|}{c}$, esiste un valore critico, $\bar{p}$, tale che se $\bar{p} \leq p \leq 1$ la rete dinamica di tipo small-world si sincronizzerà.
4) Per ogni $p \in (0\ ,\ 1]$ esiste un valore critico, $\bar{N}$, tale che se $N > \bar{N}$ la rete raggiunge la sincronizzazione.

[32] Figure tratte da Xiao Fan Wang, op.cit. fig.19 pag. 907

Questi risultati implicano che la possibilità di raggiungere la sincronizzazione nella rete nearest-neighbor coupled si rafforza molto aggiungendo una minuscola frazione di connessioni a largo raggio, e questo spiega il vantaggio delle reti di tipo small-world per la sincronizzazione del caos.

- **Sincronizzazione delle reti di tipo scale-free**

Se $A_{sf}(\bar{m},N)$ è la matrice di connessione di una rete dinamica con connessioni di tipo scale-free secondo l'algoritmo di Barabasi, e $m_o = m = \bar{m}$, è stato ricavato[33] che per un fissato valore di $\bar{m}$, il secondo maggiore autovalore della matrice di connessione, $\lambda_{2sf}(\bar{m},N)$, è una funzione decrescente e limitata di N.

D'altra parte

$$\lim_{N\to\infty} \lambda_{2sf}(\bar{m}, N) = \bar{\lambda}_{2sf}(\bar{m}) < 0$$

Il che significa che il secondo autovalore decresce verso una costante negativa, con il tendere di N all'infinito. In particolare per $\bar{m} = 3,\ 5\ ,7$, si ha rispettivamente $\hat{\lambda}_{2sf}(\bar{m})$ ≈ -0.94 , -0.97 , -0.98 .

Questo implica che l'aggiunta di nuovi nodi in uno scale-free network non può ridurre la sincronizzabilità della rete.

A causa del processo di auto-organizzazione di una rete scale-free, la possibilità di sincronizzare tali networks con un gran numero di nodi rimane quasi invariata aggiungendone nuovi.

Ricordando che $\lambda_{2sc} = -1$ per uno star-coupled network, la sincronizzabilità di una rete di tipo scale-free è quasi la stessa di quella di una rete a stella. Questo può essere dovuto alla distribuzione di connettività estremamente disomogenea dello scale-free, nel quale pochi nodi hub si comportano come un singolo centro dello star-network.

Per un numero sufficientemente grande di nodi la sincronizzazione può essere raggiunta se :

$$c \geq \left|\frac{a}{\lambda_{2sf}}\right|$$

per esempio[34] per $\bar{m} = 3\ ,5\ ,7,$ la sincronizzazione può essere raggiunta, rispettivamente, con $c > 1.35$, $c > 1.31$, $c > 1.30$, con a=1,27.

3.6 Robustezza e fragilità in reti scale-free

Mettiamo a confronto la robustezza della sincronizzazione in tali networks, con quelli di tipo random, rimuovendo una piccola frazione di nodi dalla rete.

Chiaramente la rimozione di alcuni nodi dal network, può solo cambiare la matrice di connessione, ma se λ_{2sf} rimane invariato, la sincronizzabilità del network rimarrà invariata dopo la rimozione.

[33] Wang & Cheng, 2001 b in :Xiao Fan Wang, *Complex networks: topology, dynamics and synchronization*, international journal of bifurcation and Chaos, Vol. 12, N°5 (2002), pag.908

[34] Wang & Chen op. cit. pag. 909

Per semplificare poniamo $A_{sf} \in \mathfrak{N}^{NxN}$ e $\overline{A}_{sf} \in \mathfrak{N}^{(N-[fN])x(N-[fN])}$ per indicare rispettivamente le matrici di connessione del network originario con N nodi, e del nuovo network ottenuto rimuovendo [fN] nodi[35]. Consideriamo i valori medi degli autovalori λ_{2sf} e $\overline{\lambda}_{2sf}$ e supponiamo che i nodi rimossi siano i_1 , i_2 , …… ,$[i_{fN}]$; si può costruire la nuova matrice di connessione $\overline{A}_{sf}$ da quella originaria A_{sf} in questo modo:

1) Si forma la matrice minore B$\in \mathfrak{N}^{(N-[fN])x(N-[fN])}$ togliendo le righe-colonne di indice i_1 , i_2 , …… ,$[i_{fN}]$.
2) La matrice $\overline{A}_{sf}$ si ottiene dalla matrice B, ponendo $\bar{a}_{ij} = b_{ij}$ per:

$$i \neq j\,; \qquad \text{e} \qquad \bar{a}_{ii} = -\sum_{j=1, j\neq i}^{N-[fN]} \bar{a}_{ij}$$

Nella simulazione Wang e Chen hanno posto $N=3000$ e $m_o=3$, per cui il network originario conteneva tremila nodi e circa cinquemila connessioni; hanno scoperto che anche quando venivano rimossi casualmente il 5% dei nodi, il nuovo autovalore $\overline{\lambda}_{2sf} \cong \lambda_{2sf}$, questo significa la sincronizzazione è quasi inattaccabile per gli errori di tipo random.

Invece si dimostra particolarmente vulnerabile per gli attacchi intenzionali. Per simulare il comportamento della struttura scale-free rispetto agli attacchi intenzionali, rimuoviamo per primo il nodo con più alto grado, ed in seguito continuiamo a rimuovere i nodi secondo l'ordine decrescente dei loro gradi. Facendo ciò, si è trovato che la grandezza di λ_2 decresceva rapidamente quasi fino a metà del suo valore originario, quando solo l'1% dei nodi connessi erano stati rimossi; ad una soglia critica $f \approx 1.6\%$, $\overline{\lambda}_{2sf}$ improvvisamente scendeva a zero, implicando che l'intero network si scomponeva in clusters isolati.

La tolleranza all'errore random e la vulnerabilità agli attacchi intenzionali da parte degli scale-free networks, dipendono dalla loro estremamente disomogenea distribuzione di connessione. In effetti, se $N \gg 1$ ed $f \ll 1$ non importa se i nodi rimossi sono stati scelti a caso o selezionati specificatamente, si ha sempre :

$$\lambda_2(B) = \lambda_{2sf},$$

dove B è un minore di una matrice A, ricavata come visto sopra; viceversa se sono rimossi la maggior parte dei nodi connessi, allora la nuova matrice di connessione $\bar{A}_{sf}$ si ottiene dalla matrice B aggiungendo una matrice diagonale non negativa, con un numero elevato sulla diagonale principale, per cui si avrà un drastico incremento nel secondo autovalore della matrice, che risulterà così molto maggiore dell'autovalore della matrice originaria.

In conclusione, la connettività e la sincronizzazione dei networks scale-free, sono robuste nei confronti delle infezioni casuali, ma fragili di fronte agli attacchi intenzionali; questi risultati, insieme con alcuni recenti studi circa il robusto ma fragile aspetto nei sistemi complessi[36] indicano che tale proprietà (robust yet fragile) è una caratteristica comune ed essenziale in essi.

[35] [fN] indica il più piccolo e più vicino intero al numero reale fN
[36] Carson & Doyle 1999-2000

3.7 Controllo pinning di reti complesse

Negli ultimi decenni l'analisi ed il controllo dei comportamenti complessi delle reti, composte da un grande numero di circuiti dinamici, ha suscitato un grande interesse. D'altra parte, la maggior parte di questi lavori si sono concentrati sulle strutture con connessioni completamente regolari. Il principale risultato di queste architetture semplici, è aver permesso la focalizzazione della complessità causata dalle dinamiche non lineari dei nodi, anche senza considerare l'ulteriore complessità della stessa struttura della rete. Recentemente è aumentato l'interesse nel comprendere le modalità che caratterizzano la formazione e la topologia delle varie reti complesse, incluso Internet , Word Wide Web, e tanti altri. L'apparente ubiquità di queste reti complesse ci mostra come la struttura di una rete facilita o contrasta i comportamenti di questa. Solo negli ultimi anni, grazie alla grande potenza di calcolo dei moderni calcolatori, si è potuto analizzare l'enorme numero di dati di tali sistemi. Tradizionalmente un network di topologia complessa era descritto da un grafo completamente random, che è esattamente l'opposto del modello regolare. Come già detto nei paragrafi precedenti, la maggior parte delle reti reali non sono né completamente regolari, né completamente random. Per descrivere il passaggio da una rete regolare ad una random, Watts e Strogatz hanno introdotto il network cosiddetto small-world; l'aspetto comune del modello ER e del modello WS, è che la distribuzione di connettività ha un valore medio dei picchi, che decade esponenzialmente, questo significa che ogni nodo ha approssimativamente lo stesso numero di connessioni. Un'altra scoperta significativa nel campo delle reti complesse con grande numero di connessioni è l'osservazione che la gran parte di queste sono di tipo scale-free, ed hanno distribuzioni di connessione power-law, e di conseguenza molti nodi hanno poche connessioni e pochi nodi hanno molte connessioni. Questo aspetto (vedi paragrafo precedente), le rende tolleranti agli errori di tipo random (come per esempio una rimozione casuale di nodi), ma estremamente fragili agli attacchi esterni. Wang e Chen[37] hanno studiato il problema del controllo di una rete dinamica scale-free, applicando ad una piccola frazione di nodi, un "linear feedback". Il feedback pinning è stata una tecnica comune per il controllo del caos spazio-temporale nelle reti dinamiche regolari. I due studiosi hanno ricercato gli effetti di due tipi di pinning: nel primo sono connessi al master un piccolo numero di nodi con molte connessioni, nell'altro i nodi sono scelti a caso. Essi hanno dimostrato che a causa della distribuzione di connettività estremamente disomogenea di un network scale-free, è molto più efficace il pinning su alcuni nodi, che hanno molte connessioni, rispetto al pinning sullo stesso numero di nodi, scelti casualmente.

Supponiamo che noi vogliamo portare una rete di tipo scale-free ad uno stato stazionario definito da:

$$x_1 = x_2 = \ldots = x_n = \bar{x} \tag{1}$$

e che

$$f(\bar{x}) = 0 \tag{2}$$

per realizzare lo scopo di controllo con il pinning effettuato su una piccola frazione $\delta(0<\delta<1)$ di nodi.

[37] Xiao Fan Wang, Guanrong Chen, *Pinning control of scale-free dynamical network*, Phisica A 310(2002) pag. 522

Supponiamo che $i_1, i_2, \ldots i_l$ nodi siano stati selezionati , dove $l=[\delta N]$ sta per il più piccolo e più vicino intero al numero reale δN. Le equazioni del sistema controllato sono:

$$\dot{x}_{i_k} = f(x_{i_k}) + c\sum_{j=1}^{N} a_{i_k j}\, \Gamma x_j - cd(x_{i_k} - \bar{x}) \quad \text{per } k=1,2,\ldots l \tag{3a}$$

$$\dot{x}_{i_k} = f(x_{i_k}) + c\sum_{j=1}^{N} a_{i_k j}\, \Gamma x_j \qquad \text{per } k=l+1, l+2,\ldots\ldots N \tag{3b}$$

dove d>0 è il guadagno di feedback.

Per studiare la stabilità dello stato stazionario $\bar{x}$, linearizziamo la prima equazione intorno ad $\bar{x}$. Questo ci porta a :

$$\dot{\eta} = \eta[Df(\bar{x})] + B\,\eta\Gamma, \tag{4}$$

dove $Df(\bar{x}) \in \mathcal{R}^{nxn}$ è lo jacobiano di f su $\bar{x}$, $\eta = (\eta_1, \eta_2, \ldots . \eta_N)^T \in \mathcal{R}^{Nxn}$

con $\eta_i(t) = x_i(t) - \bar{x}$, per $i = 1,2,\ldots N$, e

$$B = A - D, \quad D=\mathrm{diag}(d_1=d_2= \ldots =d_n) \tag{5}$$

Dove:

$d_{ik} = d$ per $-1 \le k \le l$, e $d_{ik} = 0$ per $l+1 \le k \le N$.

Siano $\lambda_1 \ge \lambda_2 \ge \lambda_3 \ge \ldots\ldots \lambda_N$ gli autovalori della matrice B, e $\Phi = (\Phi_1, \Phi_2, \ldots, \Phi_N) \in \mathcal{R}^{NxN}$ siano i corrispondenti autovettori, che soddisfano la condizione:

$$B\,\Phi_k = \lambda_k \Phi_k, \quad k=1.2\ldots N \tag{6}$$

Espandendo ogni colonna η sulla base Φ,otteniamo:

$$\eta = \Phi\nu \tag{7}$$

dove la matrice ν soddisfa le seguenti condizioni:

$$\dot{\nu} = \nu[D\, f(\bar{x})] + c\lambda\nu\Gamma \tag{8}$$

con $\lambda = \mathrm{diag}(\lambda_1, \lambda_2, \lambda_3, \ldots\ldots \lambda_N)$. Sia ν_k la k_{ma} riga di ν. Allora:

$$\nu_k{}^T = [D\, f(\bar{x}) + c\lambda_k\Gamma]\nu_k{}^T \qquad k=1,2,\ldots N. \tag{9}$$

Abbiamo così trasformato un problema di stabilità di un sistema a $(n \times N)$ dimensioni, in un problema di stabilità di N numeri di sistemi lineari a n dimensioni. Dalla teoria dei sistemi lineari ne consegue che, se la matrice $[D\, f(\bar{x}) + c\lambda_k\Gamma]$ è una matrice di Hurwitz, cioè le parti reali dei suoi autovalori sono tutte negative, i sistemi lineari sono tutti esponenzialmente stabili, la qualcosa implica che lo stato stazionario omogeneo di un sistema controllato è localmente esponenzialmente stabile. D'altra parte, dal momento che $\lambda_1(A) = 0$, lo stato stazionario omogeneo di un sistema controllato è instabile se $\bar{x}$ non è un punto di stabilità di un nodo isolato. Ne consegue il lemma:

LEMMA 1. Consideriamo un network controllato. Supponiamo che esista una costante $\rho<0$ tale che $[D\, f(\bar{x}) + \rho\Gamma]$ sia una matrice di Hurwitz. Sia λ_2 il più grande autovalore della matrice B=A-D.

Se $c\lambda_2 \le \rho$, lo stato omogeneo stazionario $\bar{x}$ del network controllato è localmente ed esponenzialmente stabile. Poiché $\lambda_2 < 0$ e $\rho < 0$, la disuguaglianza è equivalente a:

$$c \ge \left|\frac{\rho}{\lambda_2}\right|$$

Ad un piccolo valore di λ_2 corrisponde un grande valore di $|\lambda_2|$ per cui il network controllato si può stabilizzare in uno stato stazionario con un piccolo coefficiente di accoppiamento.

D'altra parte, la stabilità del network rispetto ad uno specifico schema di pinning, può essere caratterizzata da un grande autovalore λ_2 della matrice B. Si possono scegliere gli l nodi pinnati, e il guadagno di feedback d tali che λ_2 sia il più piccolo possibile.

Per un dato schema di pinning, λ_2 è funzione decrescente di d. Supponiamo che $i_1, i_2, \dots, i_l$ siano scelti come nodi per il pinning, e sia $\widetilde{A} \in \mathcal{R}^{(N-l)x(N-l)}$ la matrice minore di A, che si ottiene da questa rimuovendo le $i_1, i_2, \dots, i_l$, righe colonne di A. Allora abbiamo:

$$\lim_{d\to\infty} \lambda_2(B) = \lambda_2(\widetilde{A})$$

In effetti nel limite $d\to\infty$ gli stati dei nodi controllati $i_1, i_2, \dots, i_l$, possono essere sicuramente collegati tramite pinning allo stato $\bar{x}$ che si vuole raggiungere. In seguito dobbiamo solo ricercare la stabilità del seguente sistema:

$$x_{i_k} = \bar{x}$$

$$\dot{x}_{i_k} = f(x_{i_k}) + c\sum_{j=1}^{N} a_{i_k j}\, \Gamma x_j \qquad \text{per } k=l+1, l+2, \dots, N$$

- **Controllo di un network scale-free con differenti schemi di pinning**

Wang e Cheng [38] hanno fatto una simulazione sulla stabilità di un network dinamico scale-free con diversi schemi di pinning. In tale simulazione si pone il numero dei nodi $N=3000$ e $m_o=m=3$; nominativamente, il network originale contiene tremila nodi con circa novemila connessioni. Nello schema di pinning di tipo random, si collega il master a una frazione δ di nodi scelti a caso. Nello schema specifico dello scale free, si collega al master il nodo con più alto grado, ed in seguito agli altri nodi in ordine decrescente di gradi. Le matrici di connessione dei due schemi, sono denotate rispettivamente come D_r e D_s, i più grandi autovalori sono denotati rispettivamente come λ_{2r} e λ_{2s} e sono funzioni decrescenti della frazione δ del guadagno di feedback d.

Si può dire che:

$$\lim_{d\to\infty} \lambda_{2r}(\delta, d) = \tilde{\lambda}_{2r}(\delta), \quad \lim_{d\to\infty} \lambda_{2s}(\delta, d) = \tilde{\lambda}_{2s}(\delta)$$

Dove $\tilde{\lambda}_{2r}(\delta)$ e $\tilde{\lambda}_{2s}(\delta)$ sono i più grandi autovalori delle matrici minori della matrice di accoppiamento A, rispetto allo schema random ed allo schema specifico dello scale-free.

come si può vedere nella figura seguente[39]:

[38] Wang e Chen,op. cit. pag.526

[39] Figura tratta da Wang , Chen, *Pinning controll of scale-free dynamical networks*, Physica, A 310(2002) pag.527

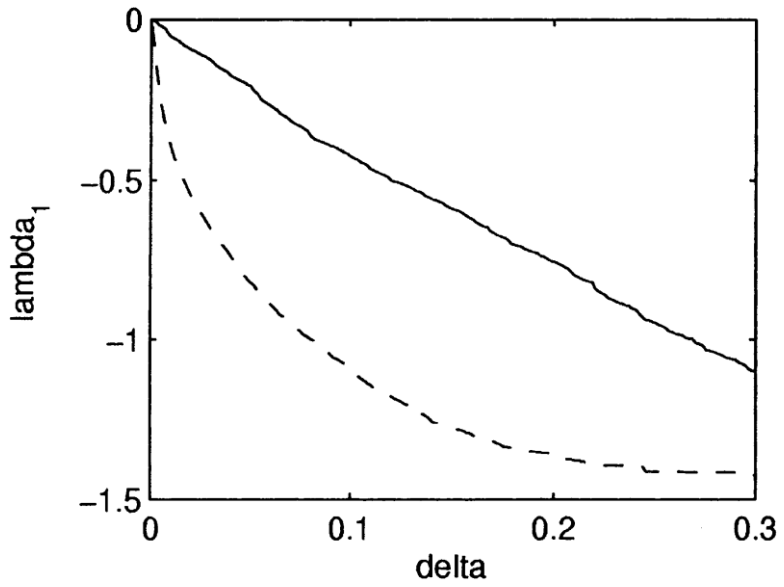


Figura 3.11 Valori di $\tilde{\lambda}_{2s}$ e $\tilde{\lambda}_{2r}$ in funzione della percentuale di nodi pinnati δ

$\tilde{\lambda}_{2s}(\delta) \ll \tilde{\lambda}_{2r}(\delta)$, specialmente per piccoli valori di δ, questo implica che per stabilizzare un network dinamico scale-free, lo schema di pinning specifico richiede un numero molto minore di controllori locali dello schema di pinning random; questo aspetto dei networks dinamici scale-free è determinato dalla loro estremamente disomogenea distribuzione di connettività, dal momento che la maggior parte dei nodi hanno poche connessioni, questi saranno selezionati in uno schema di tipo random, con una maggiore probabilità se $N \gg 1$ e $\delta \ll 1$. Non sorprende vedere che per la stabilità dell'intero network scale-free, la stabilità di pochi nodi con molte connessioni, è molto più importante della stabilità dello stesso numero dei nodi con poche connessioni, inoltre per stabilizzare un network scale-free con un numero minimo di controllori locali, questi si aggiungono ai pochi nodi con gradi molto alti.

- **Esempio: controllo pinning di circuiti di Chua**

Wang e Chen[40] hanno fatto un esperimento, usando un network scale-free, avente per nodi oscillatori di Chua.

Come visto nella prima parte, le equazioni adimensionali del circuito di Chua sono:

$$\begin{cases} \dot{x}_1 = \alpha(x_2 - x_1 + f(x_1)) \\ \dot{x}_2 = x_1 - x_2 + x_3 \\ \dot{x}_3 = \beta\, \dot{x}_2 - \gamma \dot{x}_3 \end{cases} \qquad \text{per } i = 1,2,\ldots N.$$

[40] Wang e Chen,op. cit. pag.528

e la funzione lineare a tratti f(x) è:

$$f(x)=\begin{cases} -b\,x_1-b+a & \text{se } x_1<-1 \\ a\,x_1 & \text{se} |x_1|\leq 1 \\ -b\,x_1-a+b & \text{se } x_1>1 \end{cases}$$

con $\alpha > 0, \beta > 0, \gamma > 0$, e $a<b<0$.

Supposto che due oscillatori di Chua siano collegati attraverso la prima variabile di stato x_1, per cui Γ=diag(1,0,0), le equazioni della rete sono allora:

$$\dot{x}_{i1}=\alpha(x_{i2}-x_{i1}+f(x_{i1}))+c\sum_{j=1}^{N} a_{ij}\,x_{j1}$$

$$\dot{x}_{i2}=x_{i1}-x_{i2}+x_{i3}$$

$$\dot{x}_{i3}=-\beta\,\dot{x}_{i1}-\gamma\dot{x}_{i3}$$

Se i parametri del sistema sono:

α =10.0000, β = 15.0000, γ=0.0385, a=-1.2700, b=-0.6800, l'oscillatore di Chua ha un attrattore caotico, con tre punti di equilibrio instabile:

Se vogliamo stabilizzare il network portandolo ad uno stadio omogeneo $\bar{x}$+by, applicando un controllo di tipo pinning alla prima variabile di stato di una piccola frazione δ di nodi,le equazioni del network controllato sono:

$$\dot{x}_{i1}=\alpha(x_{i2}-x_{i1}+f(x_{i1}))+c\sum_{j=1}^{N} a_{ij}\,x_{j1}+u_i$$

$$\dot{x}_{i2}=x_{i1}-x_{i2}+x_{i3}$$

$$\dot{x}_{i3}=-\beta\,\dot{x}_{i1}-\gamma\dot{x}_{i3}$$

per i = 1,2,…N, con:

$u_i = cd(x_1{}^{+} - x_{i1}\ ),\quad \text{per}\quad i = i_{1,},\quad i_{12},\ldots.;\quad 0\ \text{negli altri casi.}$

Poiché $\rho = -4{,}71$, per guadagno di feedback sufficientemente grande d,

la stabilizzazione può essere raggiunta se:

$c\lambda_2\,(\delta)<4.71.$

Consideriamo un determinato fattore di accoppiamento c, e siano δ_r e δ_s i più piccoli valori valori di δ con i quali può raggiungere la sincronizzazione rispettivamente una rete di tipo random e una rete di tipo scale free.

δ_r e δ_s sono funzioni del coefficiente di accoppiamento c. E' evidente che, per un grande range del coefficiente di accoppiamento, δ_s è molto più piccolo di δ_r, la qualcosa dimostra che lo schema scale-free richiede un numero molto inferiore di controllori locali dello schema random.

In conclusione, è stato dimostrato che il metodo più efficace per collegare i 'controllori' ai nodi della rete, è collegarli ai nodi con un numero più alto di connessioni, principalmente a causa dell'estrema disomogeneità dei networks di tipo scale-free.

Questo risultato spiega anche perché reti complesse, come ad esempio Internet e le reti metaboliche, rimangano stabili, nonostante i frequenti esempi di instabilità di alcuni nodi locali.

D'altra parte,come già visto nel paragrafo 8, la instabilità di una piccola frazione dei nodi che presentano il maggior numero di connessioni, può avere una grande influenza sulla stabilità dell'intera rete scale-free.

Questo indica, ancora una volta, che la stabilità di questo tipo di rete è "robust yet fragile".

3.8 Analisi numerica della topologia delle reti complesse

3.8.1 confronto tra varie topologie

Per scegliere la migliore configurazione tra le varie topologie di reti occorre tenere conto del maggiore autovalore non nullo λ_2 di ogni rete.
Il sistema di N Chua è asintoticamente stabile se l'esponente maggiore di Lyapunov è negativo, la qualcosa si verifica se k_{lim} è maggiore o uguale ad un valore che abbiamo trovato essere uguale a 6.31 per i parametri nominali dei circuiti di Chua, data la condizione di sincronizzazione $c\lambda_2 \geq k$, da cui si ricava:

$$\frac{C_2 R}{C_1 R_c}\lambda_2 \geq k \rightarrow R_c \leq \frac{C_2 R}{k C_1}\lambda_2 = R_{th}$$

Per gli esempi seguenti realizzati con il Matlab sono stati considerati i valori nominali delle capacità $C_1 = 10\ nF$, $C_2 = 100nF$, $R = 1.78K\Omega$; essendo questi dati di ingresso della funzione utilizzata, possono essere tranquillamente sostituiti.

- **Confronto tra sei reti con otto nodi, realizzato con il Matlab**

Per confrontare le sei reti seguenti utilizziamo le funzioni:

```
1) function[Ac,λ2,Rc] = MetodoCartaPenna2(x,C1,C2,R,Klim)
```

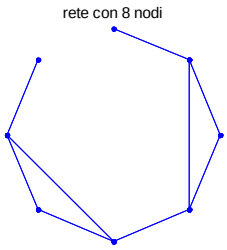
Questa funzione restituisce una rete di qualsiasi tipo e per qualsiasi numero N di nodi, se nel parametro x sono indicate tutte le connessioni dei nodi, in modo unilaterale. Nel parametro y, presente nella funzione,le connessioni vengono prese in senso inverso.
PARAMETRI di uscita:
Ac=matrice di connessione della rete iniziale.
λ2=il più grande autovalore delle matrici di connessione, escluso lo zero
Rc=valore massimo della resistenza di collegamento tra i circuiti di Chua
PARAMETRI DI INGRESSO:
x=tutte le connessioni fra i nodi i,j in un solo verso; un segnale di errore avverte se una connessione è ripetuta, in senso diretto o in senso inverso.
C1, C2, R= parametri del circuito di Chua
Klim =K limite
OUTPUT
Ci=gradi di ogni nodo
C=grado medio della rete
Gradi=gradi in ordine decrescente
Nodi=rispettivi nodi
grado_max=grado massimo
nodi_hub=nodi aventi il grado massimo

esempio per la configurazione N° 1:

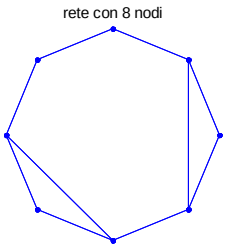
```
[Ac,λ2,Rc] =MetodoCartaPenna2 ([1 2 2 3 3 4 2 4 4 5 5 6 5 7 6 7 7 8],10e-9,100e-9,1780,6.31);
run3_MetodoCartaPenna :
```

questo programma utilizza la precedente funzione per confrontare le sei configurazioni, trovarne la migliore e creare una tabella riassuntiva dei valori di λ_2 ed R_{th} , prese entrambe in ordine decrescente.

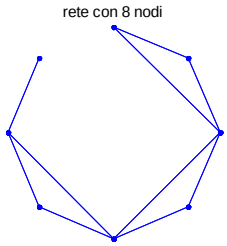
configurazione 1
Ac =
-1 1 0 0 0 0 0 0
1 -3 1 1 0 0 0 0
0 1 -2 1 0 0 0 0
0 1 1 -3 1 0 0 0
0 0 0 1 -3 1 1 0
0 0 0 0 1 -2 1 0
0 0 0 0 1 1 -3 1
0 0 0 0 0 0 1 -1
autovaloridiAc = -4.7321 -4.0000 -3.7321 -3.0000 -1.2679 -1.0000 -0.2679 0.0000
lambda2 = -0.2679
Rc = 755.8630
Ci = 1 3 2 3 3 2 3 1
C = 2.2500
Rc = 755.8630



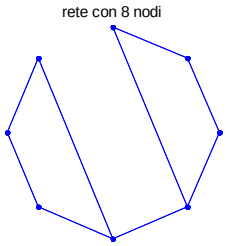
configurazione 2
Ac =
-2 1 0 0 0 0 0 1
1 -3 1 1 0 0 0 0
0 1 -2 1 0 0 0 0
0 1 1 -3 1 0 0 0
0 0 0 1 -3 1 1 0
0 0 0 0 1 -2 1 0
0 0 0 0 1 1 -3 1
1 0 0 0 0 0 1 -2
autovaloridiAc = -4.8136 -4.0000 -4.0000 -3.0000 -2.5293 -1.0000 -0.6571 0.0000
lambda2 = -0.6571
Rc =1.8536e+003
Ci = 2 3 2 3 3 2 3 2
C = 2.5000
Rc=1.8536e+003



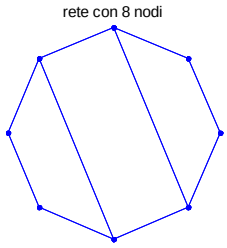
configurazione 3
Ac =
-2 1 1 0 0 0 0 0
1 -2 1 0 0 0 0 0
1 1 -4 1 1 0 0 0
0 0 1 -2 1 0 0 0
0 0 1 1 -4 1 1 0
0 0 0 0 1 -2 1 0
0 0 0 0 1 1 -3 1
0 0 0 0 0 0 1 -1
autovaloridiAc = -5.4954 -4.4950 -3.7714 -3.0000 -1.7277 -1.1060 -0.4045 -0.0000
lambda2 = -0.4045
Rc =1.1409e+003
Ci = 2 2 4 2 4 2 3 1
C = 2.5000
Rc = 1.1409e+003



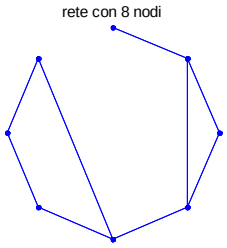
confugurazione 4
Ac =
-2 1 0 1 0 0 0 0
1 -2 1 0 0 0 0 0
0 1 -2 1 0 0 0 0
1 0 1 -3 1 0 0 0
0 0 0 1 -3 1 0 1
0 0 0 0 1 -2 1 0
0 0 0 0 0 1 -2 1
0 0 0 0 1 0 1 -2
autovaloridiAc = -4.9032 -4.0000 -2.8061 -2.0000 -2.0000 -2.0000 -0.2907 -0.0000
lambda2 = -0.2907
Rc = 820.1107
Ci = 2 2 2 3 3 2 2 2
C = 2.2500
Rc = 820.1107



configurazione 5
Ac =
-3 1 0 1 0 0 0 1
1 -2 1 0 0 0 0 0
0 1 -2 1 0 0 0 0
1 0 1 -3 1 0 0 0
0 0 0 1 -3 1 0 1
0 0 0 0 1 -2 1 0
0 0 0 0 0 1 -2 1
1 0 0 0 1 0 1 -3
autovaloridiAc =-5.4142 -4.0000 -3.4142 -2.5858 -2.0000 -2.0000 -0.5858 -0.0000
lambda2 = -0.5858
Rc = 1.6525e+003
Ci = 3 2 2 3 3 2 2 3
C = 2.5000
Rc =1.6525e+003



configurazione 6
Ac =
-1 1 0 0 0 0 0 0
1 -3 1 1 0 0 0 0
0 1 -2 1 0 0 0 0
0 1 1 -3 1 0 0 0
0 0 0 1 -3 1 0 1
0 0 0 0 1 -2 1 0
0 0 0 0 0 1 -2 1
0 0 0 0 1 0 1 -2
autovaloridiAc = -4.8280 -4.0000 -3.4664 -2.3154 -2.0000 -1.1116 -0.2786 -0.0000
lambda2 = -0.2786
Rc = 785.8652
Ci = 1 3 2 3 3 2 2 2
C = 2.5000
Rc = 785.8652



λ_2 per tutte le configurazioni esaminate

λ_2 = -0.2679 -0.6571 -0.4045 -0.2907 -0.5858 -0.2786

resistenze di soglia per tutte le configurazioni esaminate in ordine decrescente

R_{th} = 1.0e+003 *

1.8536 1.6525 1.1409 0.8201 0.7859 0.7559

λ_2 = -0.6571 -0.5858 -0.4045 -0.2907 -0.2786 -0.2679

tutti gli autovalori λ2, in valore assoluto, delle configurazioni, in ordine crescente e le relative configurazioni

λ_2 =0.6571 0.5858 0.4045 0.2907 0.2786 0.2679

CONFIGURAZIONI CORRISPONDENTI= 2 5 3 4 6 1

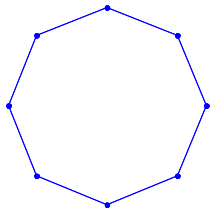
La configurazione migliore con il maggiore valore assoluto di λ_2, quindi con la maggiore Rth è la n° 2

TABELLA RIASSUNTIVA

CONFIGURAZIONI	λ_2	R_{th} (Ω)
CONFIGURAZIONE 2	0.65708	1854
CONFIGURAZIONE 5	0.58579	1652
CONFIGURAZIONE 3	0.40446	1141
CONFIGURAZIONE 4	0.29072	820
CONFIGURAZIONE 6	0.27858	786
CONFIGURAZIONE 1	0.26795	756

- **Confronto tra sei reti con otto nodi, realizzato con il Matlab**

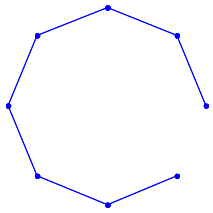
configurazione 1



```
Ac =
   -2    1    0    0    0    0    0    1
    1   -2    1    0    0    0    0    0
    0    1   -2    1    0    0    0    0
    0    0    1   -2    1    0    0    0
    0    0    0    1   -2    1    0    0
    0    0    0    0    1   -2    1    0
    0    0    0    0    0    1   -2    1
    1    0    0    0    0    0    1   -2

autovaloridiAc = -4.0000 -3.4142 -3.4142
-2.0000
  -2.0000   -0.5858   -0.5858   -0.0000
lambda2 =   -0.5858
Rc = 1.6525e+003
Ci =   2    2    2    2    2    2    2    2
```

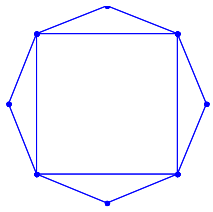
configurazione 2



```
Ac =
   -2    1    0    0    0    0    0    1
    1   -2    1    0    0    0    0    0
    0    1   -1    0    0    0    0    0
    0    0    0   -1    1    0    0    0
    0    0    0    1   -2    1    0    0
    0    0    0    0    1   -2    1    0
    0    0    0    0    0    1   -2    1
    1    0    0    0    0    0    1   -2

autovaloridiAc = -3.8478 -3.4142 -2.7654 -2.0000
  -1.2346   -0.5858   -0.1522    0.0000
lambda2 =   -0.1522
Rc =  429.4594
Ci =   2    2    1    1    2    2    2    2
```

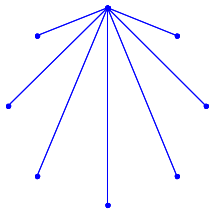
configurazione 3



```
Ac =
   -2    1    0    0    0    0    0    1
    1   -4    1    1    0    0    0    1
    0    1   -2    1    0    0    0    0
    0    1    1   -4    1    1    0    0
    0    0    0    1   -2    1    0    0
    0    0    0    1    1   -4    1    1
    0    0    0    0    0    1   -2    1
    1    1    0    0    0    1    1   -4

autovaloridiAc = -6.0000 -4.7321 -4.7321
-4.0000 -2.0000   -1.2679   -1.2679    0.0000
lambda2 =   -1.2679
Rc =  3.5768e+003
Ci =   2    4    2    4    2    4    2    4
C =   3
```

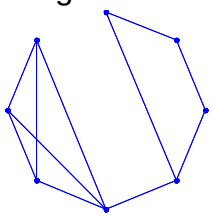
configurazione 4



```
Ac =
   -7    1    1    1    1    1    1    1
    1   -1    0    0    0    0    0    0
    1    0   -1    0    0    0    0    0
    1    0    0   -1    0    0    0    0
    1    0    0    0   -1    0    0    0
    1    0    0    0    0   -1    0    0
    1    0    0    0    0    0   -1    0
    1    0    0    0    0    0    0   -1

autovaloridiAc = -8.0000 -1.0000 -1.0000 -1.0000
-1.0000
  -1.0000   -1.0000   -0.0000
lambda2 =   -1
Rc =  2.8209e+003
Ci =   7    1    1    1    1    1    1    1
C =   1.7500
```

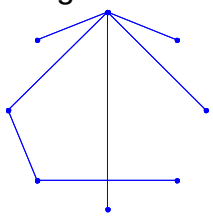
configurazione 5



```
Ac =
   -2    1    0    1    0    0    0    0
    1   -2    1    0    0    0    0    0
    0    1   -2    1    0    0    0    0
    1    0    1   -3    1    0    0    0
    0    0    0    1   -4    1    1    1
    0    0    0    0    1   -3    1    1
    0    0    0    0    1    1   -3    1
    0    0    0    0    1    1    1   -3

autovaloridiAc = -5.3234 -4.0000 -4.0000
-4.0000 -2.3579 -2.0000 -0.3187 0.0000
lambda2 =  -0.3187
Rc =  898.9405
Ci =   2    2    2    3    4    3    3    3
C =   2.7500
```

configurazione 6



```
Ac =
   -5    1    1    0    1    0    1    1
    1   -1    0    0    0    0    0    0
    1    0   -1    0    0    0    0    0
    0    0    0   -1    0    1    0    0
    1    0    0    0   -1    0    0    0
    0    0    0    1    0   -2    1    0
    1    0    0    0    0    1   -2    0
    1    0    0    0    0    0    0   -1

autovaloridiAc =  -6.0548   -3.1610   -1.5068   -1.0000
  -1.0000   -1.0000   -0.2774    0.0000
lambda2 =   -0.2774
Rc =  782.5424
Ci =   5    1    1    1    1    2    2    1
C =   1.7500
```

λ_2 per tutte le configurazioni esaminate

λ_2 = -0.5858 -0.1522 -1.2679 -1.0000 -0.3187 -0.2774

resistenze di soglia per tutte le configurazioni esaminate in ordine decrescente

R_{th} = 1.0e+003 *

3.5768 2.8209 1.6525 0.8989 0.7825 0.4295

λ_2 = -1.2679 -1.0000 -0.5858 -0.3187 -0.2774 -0.1522

tutti gli autovalori in valore assoluto delle configurazioni in ordine crescente:

λ_2 = 1.2679 1.0000 0.5858 0.3187 0.2774 0.1522

configurazioni corrispondenti = 3 4 1 5 6 2

la configurazione migliore con il maggiore valore assoluto di $\lambda 2$, quindi con la maggiore Rth è la n° 3

TABELLA RIASSUNTIVA

CONFIGURAZIONI	**λ_2**	**R_{th} (Ω)**	**R_{th} (Ω) effettiva**
CONFIGURAZIONE 3	1.27	3576.7822	3648
CONFIGURAZIONE 4	1	2820.9192	2877
CONFIGURAZIONE 1	0.58	1652.4562	1685
CONFIGURAZIONE 5	0.32	898.9405	917
CONFIGURAZIONE 6	0.28	782.5424	798
CONFIGURAZIONE 2	0.15	429.4594	438

Mentre la Rth è ricavata dai parametri nominali dei circuiti di Chua, la Rth effettiva è ricavata dai parametri misurati in laboratorio, C2=103 nF,e C1=10.1nF.

- **Confronto tra sei reti con 16 nodi, realizzato con il Matlab**

autovaloridiAc =-4.0000 -3.8478 -3.8478 -3.4142
-3.4142 -2.7654 -2.7654 -2.0000 -2.0000 -1.2346
-1.2346 -0.5858 -0.5858 -0.1522 -0.1522 0.0000

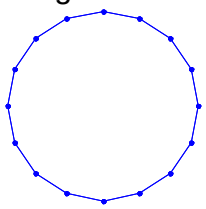
lambda2 = -0.1522
Rc = 429.4594
Ci =2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

C = 2

gradi in ordine decrescente e relativi nodi
gradi =2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
nodi =1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Rc = 429.4594

configurazione 1



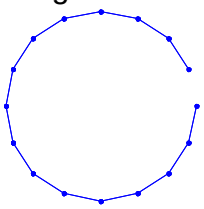
autovaloridiAc =-3.9616 -3.8478 -3.6629
-3.4142 -3.1111 -2.7654 -2.3902 -2.0000
-1.6098 -1.2346 -0.8889 -0.5858 -0.3371
-0.1522 -0.0384 0.0000

lambda2 = -0.1522
Rc = 429.4594
Ci =2 2 2 1 1 2 2 2 2 2 2 2 2 2 2 2
C = 1.8750

gradi in ordine decrescente e relativi nodi
gradi =2 2 2 2 2 2 2 2 2 2 2 2 2 1 1
nodi =1 2 3 6 7 8 9 10 11 12 13 14 15 16
4 5

Rc = 429.4594

configurazione 2

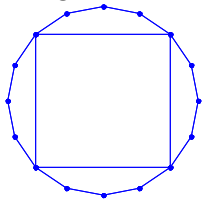


autovaloridiAc = -6.4495 -4.8260 -4.8260 -4.0000
-3.4142 -3.1524 -3.1524 -2.0000 -2.0000 -1.6145
-1.6145 -1.5505 -0.5858 -0.4071 -0.4071 -0.0000

lambda2 = -0.4071
Rc = 1.1485e+003
Ci = 2 2 4 2 2 2 4 2 2 2
4 2 2 2 4 2
C = 2.5000

gradi in ordine decrescente e relativi nodi
gradi = 4 4 4 4 2 2 2 2 2
2 2 2 2 2 2 2
nodi = 3 7 11 15 1 2 4 5
6 8 9 10 12 13 14 16
Rc = 1.1485e+003

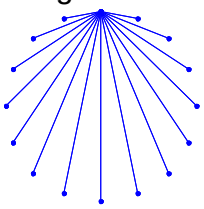
configurazione 3



autovaloridiAc =-16.0000 -1.0000 -1.0000
-1.0000
-1.0000 -1.0000 -1.0000 -1.0000 -1.0000
-1.0000
-1.0000 -1.0000 -1.0000 -1.0000 -1.0000
-0.0000
lambda2 = -1.0000
Rc = 2.8209e+003
Ci =15 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
C = 1.8750

gradi in ordine decrescente e relativi nodi
gradi =15 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
nodi = 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Rc = 2.8209e+003

configurazione 4



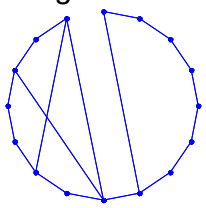
autovaloridiAc = -5.8681 -4.4173 -4.0000 -3.7652
-3.4507-3.4142 -3.1258 -2.2505 -2.0000 -2.0000
-1.1902-1.0760 -0.7383 -0.5858 -0.1179 0.0000

lambda2 = -0.1179
Rc = 332.4469
Ci = 2 2 2 2 2 2 2 3 4 2
3 2 2 3 2 3
C = 2.3750

gradi in ordine decrescente e relativi nodi
gradi = 4 3 3 3 3 2 2 2 2
2 2 2 2 2 2 2
nodi = 9 8 11 14 16 1 2 3 4
5 6 7 10 12 13 15

Rc = 332.4469

configurazione 5



autovaloridiAc = -13.0077 -3.2222
-1.5373-1.0000
-1.0000-1.0000 -1.0000 -1.0000-1.0000 -1.0000
-1.0000-1.0000-1.0000 -0.2328 -0.0000 0.0000

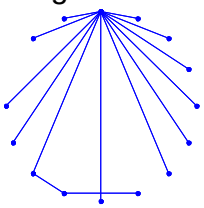
lambda2 = -0.2328

Rc = 656.6952
Ci =12 1 1 1 1 1 1 1 1 2 2 1 1 0 1 1
C =1.7500

gradi in ordine decrescente e relativi nodi
gradi =
12 2 2 1 1 1 1 1 1 1 1 1 1 1 0
nodi =1 10 11 2 3 4 5 6 7 8 9 12 13 15 16 14

Rc = 656.6952

configurazione 6



λ_2 per tutte le configurazioni esaminate

λ_2 = -0.1522 -0.1522 -0.4071 -1.0000 -0.1179 -0.2328

resistenze di soglia per tutte le configurazioni esaminate in ordine decrescente

Rth = 1.0e+003 *

2.8209 1.1485 0.6567 0.4295 0.4295 0.3324

λ_2 = -1.0000 -0.4071 -0.2328 -0.1522 -0.1522 -0.1179

tutti gli autovalori λ_2, in valore assoluto, delle configurazioni, in ordine decrescente e le relative configurazioni;

λ_2 = 1.0000 0.4071 0.2328 0.1522 0.1522 0.1179

CONFIGURAZIONI CORRISPONDENTI = 4 3 6 2 1 5

La configurazione migliore con il maggiore valore assoluto di λ_2, quindi con la maggiore Rth è la n° 4

TABELLA RIASSUNTIVA

CONFIGURAZIONI	**λ_2**	**R_{th} (Ω)**
CONFIGURAZIONE 4	1	2821
CONFIGURAZIONE 3	0.40714	1149
CONFIGURAZIONE 6	0.23279	657
CONFIGURAZIONE 2	0.15224	429
CONFIGURAZIONE 1	0.15224	429
CONFIGURAZIONE 5	0.11785	332

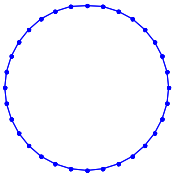
Il valore di Rth è ottenuto peri valori nominali dei parametri dei circuiti

Come si può notare, confrontando questa tabella con quella precedente, aumentando il numero dei nodi, cambia l'ordine delle configurazioni migliori.

- **Confronto tra sei reti aventi 32 nodi ognuna, realizzato con il Mat-lab**

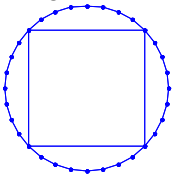
autovalori di Ac = -4.0000 -3.9616 -3.9616 -3.8478 -3.8478
-3.6629 -3.6629 -3.4142 -3.4142 -3.1111 -3.1111 -2.7654
-2.7654 -2.3902 -2.3902 -2.0000 -2.0000 -1.6098 -1.6098
-1.2346 -1.2346 -0.8889 -0.8889 -0.5858 -0.5858 -0.3371
-0.3371 -0.1522 -0.1522 -0.0384 -0.0384 0.0000
lambda2 = -0.1522
Rc =429.4594
Ci = 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
C =2
grado max = 2
gradi in ordine decrescente e relativi nodi
gradi = 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
nodi = 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22
23 24 25 26 27 28 29 30 31 32
rete regolare

configurazione 1



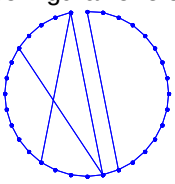
autovaloridiAc = -6.4721 -4.8284 -4.8284 -4.0000 -3.8478
-3.8066 -3.8066 -3.4142 -3.4142 -3.2945 -3.2945 -3.2787
-2.7654 -2.5859 -2.5859 -2.0000 -2.0000 -1.8046 -1.8046
-1.7700 -1.2346 -1.0706 -1.0706 -0.5858 -0.5858 -0.4872
-0.4872 -0.4792 -0.1522 -0.1221 -0.1221 0.0000
lambda2 = -0.1221
Rc = 344.5176
Ci = 2 2 2 2 4 2 2 2 2 2 2 2 4 2 2 2 2 2 2 2 4 2 2 2
2 2 2 2 4 2 2 2
C = 2.2500
grado max =4
gradi in ordine decrescente e relativi nodi
gradi = 4 4 4 4 2 2 2 2 2 2 2 2 2 2
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
nodi = 5 13 21 29 1 2 3 4 6 7 8 9 10 11 12 14
15 16 17 18 19 20 22 23 24 25 26 27 28 30 31 32
nodo hub = 5 13 21 29

configurazione 3



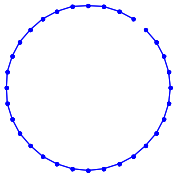
autovaloridiAc = -5.7399 -4.5743 -4.2320 -3.9563 -3.8426
-3.7787 -3.6702 -3.6180 -3.6180 -3.3075 -3.0828 -3.0000
-3.0000 -2.7584 -2.3307 -2.2091 -1.9418 -1.8714 -1.5963
-1.3820 -1.3820 -1.1554 -1.0000 -0.7461 -0.6617 -0.5349
-0.3226 -0.2793 -0.1947 -0.1729 -0.0404 -0.0000
lambda2 = -0.1729
Rc = 487.7626
Ci = 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 4 2 2 2 3 2 2 2 2 2 2 2 3 2 2 2 3
C = 2.1875
grado max = 4
gradi in ordine decrescente e relativi nodi
gradi = 4 3 3 3 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
nodi = 16 15 20 28 32 1 2 3 4 5 6 7 8 9 10 11 12 13 14 17 18 19 21
22 23 24 25 26 27 29 30 31
nodo hub = 16

configurazione 5



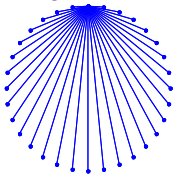
autovalori di Ac = -3.9904 -3.9616 -3.9139 -3.8478 -3.7638
-3.6629 -3.5460 -3.4142 -3.2688 -3.1111 -2.9428 -2.7654
-2.5806 -2.3902 -2.1960 -2.0000 -1.8040 -1.6098 -1.4194
-1.2346 -1.0572 -0.8889 -0.7312 -0.5858 -0.4540 -0.3371
-0.2362 -0.1522 -0.0861 -0.0384 -0.0096 0.0000
lambda2 = -0.0861
Rc = 242.9357
Ci = 2 2 2 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2 2 2 2 2 2 2
C = 1.9375
grado max = 2
gradi in ordine decrescente e relativi nodi
gradi = 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2 2 2 2 2 2 2 2 2 2 1 1
nodi = 1 2 3 6 7 8 9 10 11 12 13 14 15 16 17 18
19 20 21 22 23 24 25 26 27 28 29 30 31 32 4 5
nodi hub = 1 2 3 6 7 8 9 10 11 12 13 14 15 16 17 18
19 20 21 22 23 24 25 26 27 28 29 30 31 32

configurazione 2



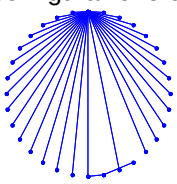
autovaloridiAc =-32.0000 -1.0000 -1.0000 -1.0000 -1.0000
-1.0000 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000
-1.0000 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000
-1.0000 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000
-1.0000 -1.0000 -1.0000 -1.0000 -1.0000 -0.0000
lambda2 = -1.0000
Rc = 2.8209e+003
Ci = 31 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
C =1.9375
grado max = 31
gradi in ordine decrescente e relativi nodi
gradi = 31 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
nodi =1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17
18 19 20 21 22 23 24 25 26 27 28 29 30 31 32
nodo hub =1

configurazione 4



autovaloridiAc = -30.0012 -3.2379 -1.5480 -1.0000 -1.0000
-1.0000 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000
-1.0000 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000
-1.0000 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000 -1.0000
-1.0000 -1.0000 -1.0000 -1.0000 -0.2128 -0.0000
lambda2 =-0.2128
Rc =600.2788
Ci=29 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
C =1.9375
grado max = 29
gradi in ordine decrescente e relativi nodi
gradi = 29 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1
nodi = 1 16 17 2 3 4 5 6 7 8 9 10 11 12
13 14 15 18 19 20 21 22 23 24 25 26 27 28
29 30 31 32
nodo hub=1

configurazione 6



λ_2 per tutte le configurazioni esaminate:

λ_2 = -0.1522 -0.0861 -0.1221 -1.0000 -0.1729 -0.2128
resistenze di soglia per tutte le configurazioni esaminate in ordine decrescente:
Rth = 1.0e+003 *
2.8209 0.6003 0.4878 0.4295 0.3445 0.2429

λ_2 in ordine crescente:
λ_2 = -1.0000 -0.2128 -0.1729 -0.1522 -0.1221 -0.0861

Tutti gli autovalori λ_2, in valore assoluto, delle configurazioni, in ordine decrescente e le relative configurazioni
λ_2 = 1.0000 0.2128 0.1729 0.1522 0.1221 0.0861
CONFIGURAZIONI = 4 6 5 1 3 2
la configurazione migliore con il maggiore valore assoluto di λ2, quindi con la maggiore Rth è la n° 4

TABELLA RIASSUNTIVA

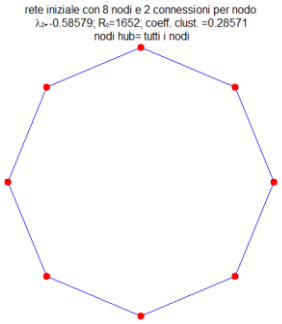
CONFIGURAZIONI	**λ_2**	**$R_{th}(\Omega)$**
CONFIGURAZIONE 4	1	2821
CONFIGURAZIONE 6	0.2128	600
CONFIGURAZIONE 5	0.17291	488
CONFIGURAZIONE 1	0.15224	429
CONFIGURAZIONE 3	0.12213	345
CONFIGURAZIONE 2	0.086119	243

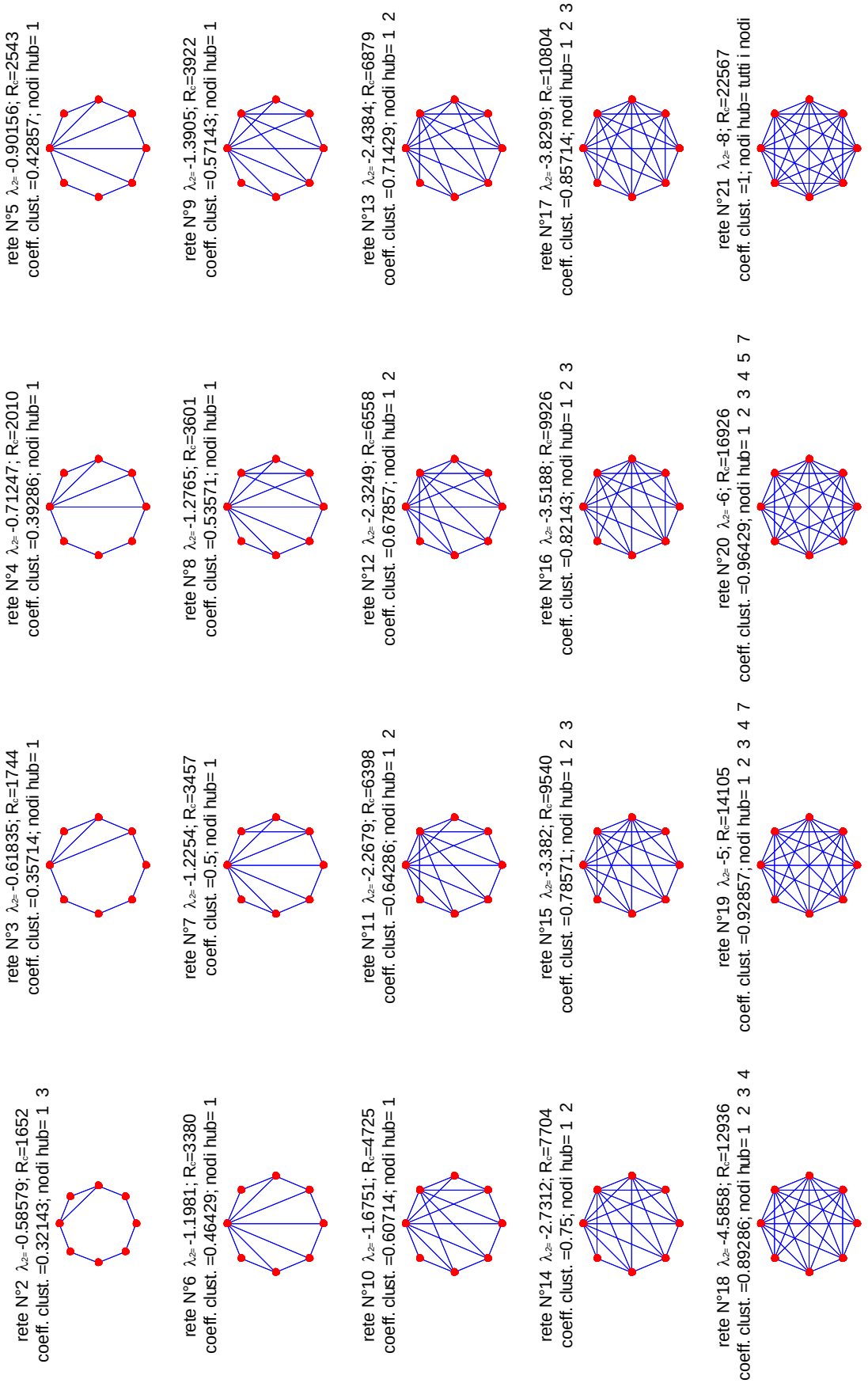
- **Esempi di tutte le connessioni possibili a partire da una configurazione tipo nearest-neighbor e ricerca delle configurazioni più convenienti.**

La rete di partenza è una rete di tipo nearest-neighbor con otto nodi, le reti seguenti si ottengono aggiungendo una connessione alla volta,fino ad arrivare alla 'globally connected'.Si è utilizzata la seguente funzione:

```
[A,λ2,Rth]=TutteLeConnessioni2(8,2,10e-9,100e-9,1780,6.31,'NN'),
```

dove : 8=numero di nodi, 2=numero di connessioni per nodo, seguono i parametri nominali dei circuiti di Chua,C1,C2,R,Klim,'NN'è il tipo di rete da cui si parte,la 'nearest neighbor'.





Tutti gli autovalori λ_2, in valore assoluto, in ordine decrescente e le relative configurazioni:

λ_2 =

8.0000 6.0000 5.0000 4.5858 3.8299 3.5188 3.3820 2.7312 2.4384 2.3249 2.2679 1.6751 1.3905 1.2765 1.2254 1.1981 0.9016 0.7125 0.6183 0.5858 0.5858

CONFIGURAZIONE =

21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 1 2

La configurazione migliore con il maggiore valore assoluto di λ_2, quindi con la maggiore Rth è la n° 21

TABELLA RIASSUNTIVA

CONFIGURAZIONI	**λ_2**	**$R_{th}(\Omega)$**	**Coefficiente di clustering**	**nodi hub**
CONFIGURAZIONE 21	8.00	22567	1.00	Rete regolare
CONFIGURAZIONE 20	6.00	16926	0.96	1 2 3 4 5 7
CONFIGURAZIONE 19	5.00	14105	0.93	1 2 3 4 7
CONFIGURAZIONE 18	4.59	12936	0.89	1 2 3 4
CONFIGURAZIONE 17	3.83	10804	0.86	1 2 3
CONFIGURAZIONE 16	3.52	9926	0.82	1 2 3
CONFIGURAZIONE 15	3.38	9540	0.79	1 2 3
CONFIGURAZIONE 14	2.73	7704	0.75	1 2
CONFIGURAZIONE 13	2.44	6879	0.71	1 2
CONFIGURAZIONE 12	2.32	6558	0.68	1 2
CONFIGURAZIONE 11	2.27	6398	0.64	1 2
CONFIGURAZIONE 10	1.68	4725	0.61	1
CONFIGURAZIONE 9	1.39	3922	0.57	1
CONFIGURAZIONE 8	1.28	3601	0.54	1
CONFIGURAZIONE 7	1.23	3457	0.50	1
CONFIGURAZIONE 6	1.20	3380	0.46	1
CONFIGURAZIONE 5	0.90	2543	0.43	1
CONFIGURAZIONE 4	0.71	2010	0.39	1
CONFIGURAZIONE 3	0.62	1744	0.36	1
CONFIGURAZIONE 1	0.59	1652	0.29	1 3
CONFIGURAZIONE 2	0.59	1652	0.32	Rete regolare

Come si può osservare, in questo tipo di configurazioni λ_2 aumenta con l'aumentare del numero di connessioni, e quindi le configurazioni più efficienti sono quelle che più si avvicinano alla globally-connected. Ciò avviene anche se si cosiderano reti con un maggior numero di nodi. Qui di seguito consideriamo le tabelle relative a reti di 16 e 32 nodi:

Reti di 16 nodi

La configurazione migliore con il maggiore valore assoluto di λ_2, quindi con la maggiore Rth è la n° 105

TABELLA RIASSUNTIVA per reti di 16 nodi

CONFIGURAZIONI	λ_2	$R_{th}(\Omega)$	coefficiente di clustering
CONFIGURAZIONE 105	16	45135	1
CONFIGURAZIONE 104	14	39493	0.99167
CONFIGURAZIONE 103	13	36672	0.98333
CONFIGURAZIONE 102	12.5858	35503	0.975
CONFIGURAZIONE 101	11.8299	3337	0.96667
CONFIGURAZIONE 100	11.5188	32494	0.95833
CONFIGURAZIONE 99	11.382	32108	0.95
CONFIGURAZIONE 98	10.7312	30272	0.94167
CONFIGURAZIONE 97	10.4384	29446	0.93333
CONFIGURAZIONE 96	10.3249	29126	0.925
CONFIGURAZIONE 95	10.2679	28965	0.91667
CONFIGURAZIONE 94	9.6751	27293	0.90833
CONFIGURAZIONE 93	9.3905	26490	0.9
CONFIGURAZIONE 92	9.2765	26168	0.89167
CONFIGURAZIONE 91	9.2254	26024	0.88333
CONFIGURAZIONE 90	9.1981	25947	0.875
CONFIGURAZIONE 89	8.6405	24374	0.86667
CONFIGURAZIONE 88	8.3611	23586	0.85833
CONFIGURAZIONE 87	8.2467	23263	0.85
CONFIGURAZIONE 86	8.193	23112	0.84167
CONFIGURAZIONE 85	8.1667	23038	0.83333
CONFIGURAZIONE 84	8.1522	22997	0.825
CONFIGURAZIONE 83	7.6175	21488	0.81667
CONFIGURAZIONE 82	7.3416	20710	0.80833
CONFIGURAZIONE 81	7.2275	20388	0.8
CONFIGURAZIONE 80	7.1724	20233	0.79167
CONFIGURAZIONE 79	7.1437	20152	0.78333
CONFIGURAZIONE 78	7.1289	20110	0.775
CONFIGURAZIONE 77	7.1206	20087	0.76667
CONFIGURAZIONE 76	6.601	18621	0.75833
CONFIGURAZIONE 75	6.3277	17850	0.75
CONFIGURAZIONE 74	6.2143	17530	0.74167
CONFIGURAZIONE 73	6.1587	17373	0.73333
CONFIGURAZIONE 72	6.1286	17288	0.725
CONFIGURAZIONE 71	6.1118	17241	0.71667
CONFIGURAZIONE 70	6.1029	17216	0.70833
CONFIGURAZIONE 69	6.0979	17202	0.7
CONFIGURAZIONE 68	5.5887	15765	0.69167
CONFIGURAZIONE 67	5.3174	15000	0.68333
CONFIGURAZIONE 66	5.2047	14682	0.675
CONFIGURAZIONE 65	5.149	14525	0.66667

CONFIGURAZIONE 64	5.1182	14438	0.65833
CONFIGURAZIONE 63	5.1003	14388	0.65
CONFIGURAZIONE 62	5.0899	14358	0.64167
CONFIGURAZIONE 61	5.0842	14342	0.63333
CONFIGURAZIONE 60	5.081	14333	0.625
CONFIGURAZIONE 59	4.5791	12917	0.61667
CONFIGURAZIONE 58	4.3094	12157	0.60833
CONFIGURAZIONE 57	4.1974	11841	0.6
CONFIGURAZIONE 56	4.1418	11684	0.59167
CONFIGURAZIONE 55	4.1108	11596	0.58333
CONFIGURAZIONE 54	4.0922	11544	0.575
CONFIGURAZIONE 53	4.0808	11512	0.56667
CONFIGURAZIONE 52	4.074	11492	0.55833
CONFIGURAZIONE 51	4.0702	11482	0.55
CONFIGURAZIONE 50	4.0681	11476	0.54167
CONFIGURAZIONE 49	3.5714	10075	0.53333
CONFIGURAZIONE 48	3.303	9318	0.525
CONFIGURAZIONE 47	3.1917	9003	0.51667
CONFIGURAZIONE 46	3.1363	8847	0.50833
CONFIGURAZIONE 45	3.1052	8759	0.5
CONFIGURAZIONE 44	3.0863	8706	0.49167
CONFIGURAZIONE 43	3.0743	8672	0.48333
CONFIGURAZIONE 42	3.0667	8651	0.475
CONFIGURAZIONE 41	3.0621	8638	0.46667
CONFIGURAZIONE 40	3.0595	8631	0.45833
CONFIGURAZIONE 39	3.0581	8627	0.45
CONFIGURAZIONE 38	2.5651	7236	0.44167
CONFIGURAZIONE 37	2.2978	6482	0.43333
CONFIGURAZIONE 36	2.187	6169	0.425
CONFIGURAZIONE 35	2.1319	6014	0.41667
CONFIGURAZIONE 34	2.1008	5926	0.40833
CONFIGURAZIONE 33	2.0817	5872	0.4
CONFIGURAZIONE 32	2.0695	5838	0.39167
CONFIGURAZIONE 31	2.0614	5815	0.38333
CONFIGURAZIONE 30	2.0562	5800	0.375
CONFIGURAZIONE 29	2.0529	5791	0.36667
CONFIGURAZIONE 28	2.0511	5786	0.35833
CONFIGURAZIONE 27	2.0501	5783	0.35
CONFIGURAZIONE 26	1.5599	4400	0.34167
CONFIGURAZIONE 25	1.2934	3649	0.33333
CONFIGURAZIONE 24	1.1832	3338	0.325
CONFIGURAZIONE 23	1.1283	3183	0.31667
CONFIGURAZIONE 22	1.0973	3095	0.30833
CONFIGURAZIONE 21	1.0782	3041	0.3
CONFIGURAZIONE 20	1.0658	3006	0.29167
CONFIGURAZIONE 19	1.0574	2983	0.28333
CONFIGURAZIONE 18	1.0518	2967	0.275
CONFIGURAZIONE 17	1.048	2956	0.26667
CONFIGURAZIONE 16	1.0457	2950	0.25833
CONFIGURAZIONE 15	1.0444	2946	0.25
CONFIGURAZIONE 14	1.0437	2944	0.24167

CONFIGURAZIONE 13	0.75321	2125	0.23333
CONFIGURAZIONE 12	0.52804	1490	0.225
CONFIGURAZIONE 11	0.39435	1112	0.21667
CONFIGURAZIONE 10	0.31126	878	0.20833
CONFIGURAZIONE 9	0.25693	725	0.2
CONFIGURAZIONE 8	0.22007	621	0.19167
CONFIGURAZIONE 7	0.19455	549	0.18333
CONFIGURAZIONE 6	0.17686	499	0.175
CONFIGURAZIONE 5	0.16491	465	0.16667
CONFIGURAZIONE 4	0.15744	444	0.15833
CONFIGURAZIONE 3	0.15352	433	0.15
CONFIGURAZIONE 2	0.15224	429	0.14167
CONFIGURAZIONE 1	0.15224	429	0.13333

RETI DI 32 nodi

la configurazione migliore con il maggiore valore assoluto di λ_2, quindi con la maggiore R_{th} è la n° 465

TABELLA RIASSUNTIVA

CONFIGURAZIONI	**λ_2**	**$R_{th}(\Omega)$**	**coefficiente di clustering**
CONFIGURAZIONE 465	32	90269	1
CONFIGURAZIONE 464	30	84628	0.99798
CONFIGURAZIONE 463	29	81807	0.99597
CONFIGURAZIONE 462	28.5858	80638	0.99395
CONFIGURAZIONE 461	27.8299	78506	0.99194
CONFIGURAZIONE 460	27.5188	77628	0.98992
CONFIGURAZIONE 459	27.382	77242	0.9879
CONFIGURAZIONE 458	26.7312	75407	0.98589
CONFIGURAZIONE 457	26.4384	74581	0.98387
CONFIGURAZIONE 456	26.3249	74260	0.98185
CONFIGURAZIONE 455	26.2679	74100	0.97984
…… …… in ordine decrescente fino alla configurazione 24 …….. ……..			
CONFIGURAZIONE 24	0.16391	462	0.11089
CONFIGURAZIONE 4	0.15477	437	0.070565
CONFIGURAZIONE 3	0.15288	431	0.068548
CONFIGURAZIONE 2	0.15224	387	0.066532
CONFIGURAZIONE 1	0.15224	387	0.064516
CONFIGURAZIONE 23	0.13659	347	0.10887
CONFIGURAZIONE 22	0.11631	295	0.10685
CONFIGURAZIONE 13	0.10164	258	0.08871

CONFIGURAZIONE 21	0.10085	256	0.10484
CONFIGURAZIONE 12	0.092105	234	0.086694
CONFIGURAZIONE 20	0.088805	225	0.10282
CONFIGURAZIONE 11	0.083847	213	0.084677
CONFIGURAZIONE 19	0.079255	201	0.10081
CONFIGURAZIONE 10	0.076649	195	0.082661
CONFIGURAZIONE 18	0.071572	182	0.09879
CONFIGURAZIONE 9	0.070338	179	0.080645
CONFIGURAZIONE 17	0.065316	166	0.096774
CONFIGURAZIONE 8	0.064773	164	0.078629
CONFIGURAZIONE 16	0.060176	153	0.094758
CONFIGURAZIONE 7	0.059843	152	0.076613
CONFIGURAZIONE 15	0.055923	142	0.092742
CONFIGURAZIONE 6	0.055454	141	0.074597
CONFIGURAZIONE 14	0.052386	133	0.090726
CONFIGURAZIONE 5	0.051531	131	0.072581

Come si può notare, nel caso di 32 nodi,il valore di $|\lambda_2|$ aumenta con l'aumentare del numero di connessioni, ma solo a partire a partire dalla rete N° 24; ciò non si verifica per le prime ventiquattro configurazioni.

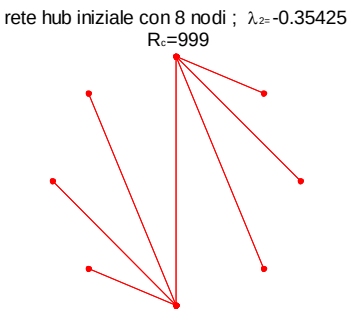
Confronto tra tre reti hub con otto nodi

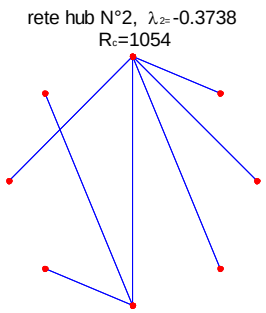
Con lo stesso programma utilizzato precedentemente si possono confrontare tre reti hub, la seconda e la terza ottenute dalla prima, mediante la traslazione di una connessione da un nodo hub all'altro.

Si è utilizzata la seguente funzione:

```
[A,λ2,Rth]=TutteLeConnessioni2(8,2,10e-9,100e-9,1780,6.31,'hub'),
```

dove: 8=numero di nodi, 2=numero di connessioni per nodo, seguono i parametri nominali dei circuiti di Chua,C1,C2,R,Klim,'hub'è il tipo di rete da cui si parte.





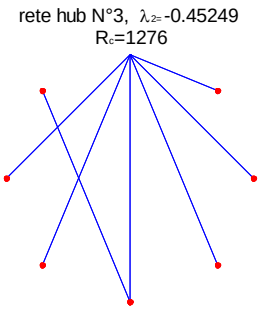
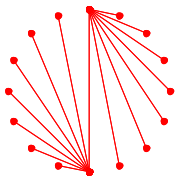


TABELLA RIASSUNTIVA

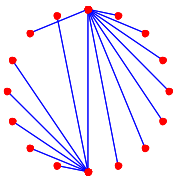
CONFIGURAZIONI	**λ_2**	**R_{th} (Ω)**
CONFIGURAZIONE 3	0.45	1276
CONFIGURAZIONE 2	0.37	1054
CONFIGURAZIONE 1	0.35	999

- **Confronto tra sette reti hub con sedici nodi**

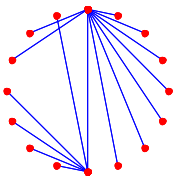
rete hub iniziale con 16 nodi ; λ_2= -0.20417
R_c=576



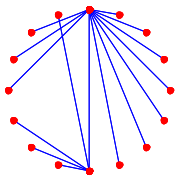
rete hub N°2, λ_2= -0.20694
R_c=584



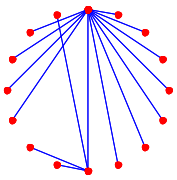
rete hub N°3, λ_2= -0.21574
R_c=609



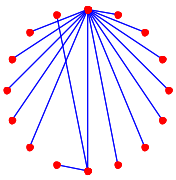
rete hub N°4, λ_2= -0.23231
R_c=655



rete hub N°5, λ_2= -0.26069
R_c=735



rete hub N°6, λ_2= -0.31065
R_c=876



rete hub N°7, λ_2= -0.41306
R_c=1165

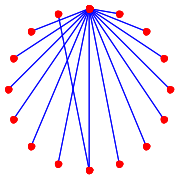
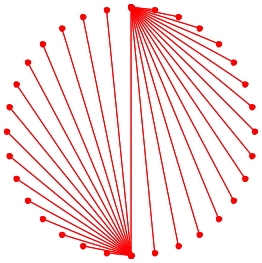


TABELLA RIASSUNTIVA

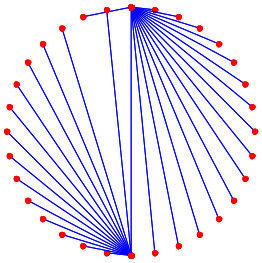
CONFIGURAZIONI	**λ_2**	**R_{th} (Ω)**
CONFIGURAZIONE 7	0.41	1165
CONFIGURAZIONE 6	0.31	876
CONFIGURAZIONE 5	0.26	735
CONFIGURAZIONE 4	0.23	655
CONFIGURAZIONE 3	0.22	609
CONFIGURAZIONE 2	0.21	584
CONFIGURAZIONE 1	0.20	576

- **Confronto tra quindici reti hub con trentadue nodi**

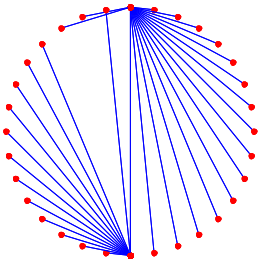
rete hub iniziale con 32 nodi ; $\lambda_{2=}$-0.11181
R_c=315



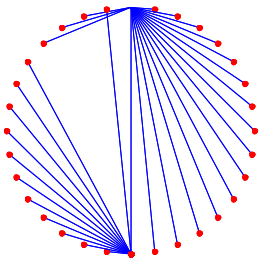
rete hub N°2, $\lambda_{2=}$-0.1122
R_c=317



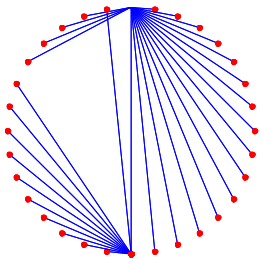
rete hub N°3, $\lambda_{2=}$-0.11341
R_c=320



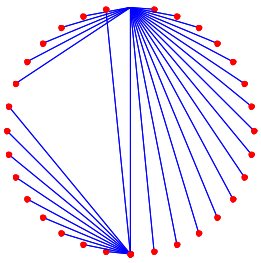
rete hub N°4, $\lambda_{2=}$-0.11549
R_c=326



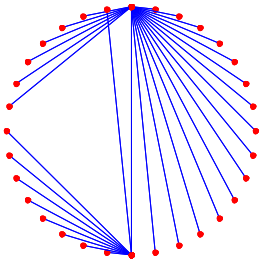
rete hub N°5, $\lambda_{2=}$-0.11853
R_c=334



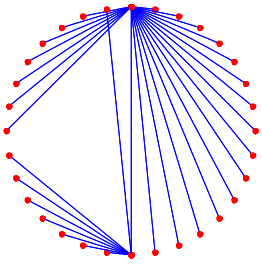
rete hub N°6, $\lambda_{2=}$-0.12268
R_c=346



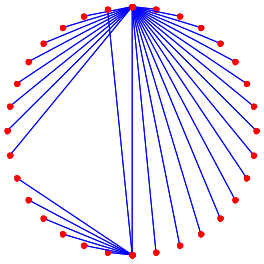
rete hub N°7, $\lambda_{2=}$-0.12818
R_c=362



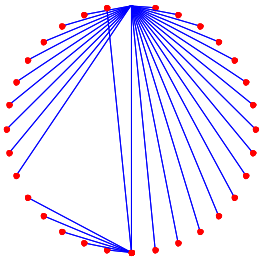
rete hub N°8, $\lambda_{2=}$-0.13535
R_c=382



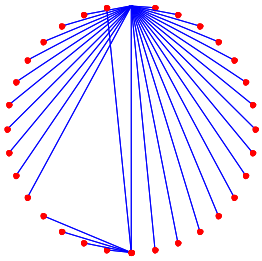
rete hub N°9, $\lambda_{2=}$-0.14473
R_c=408



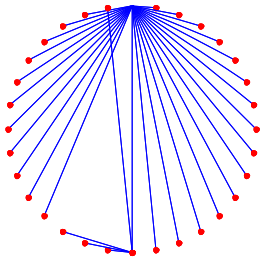
rete hub N°10, $\lambda_{2=}$-0.15711
R_c=443



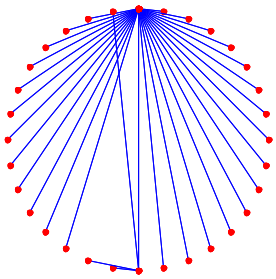
rete hub N°11, $\lambda_{2=}$-0.1738
R_c=490



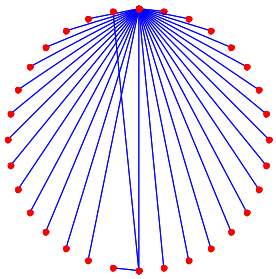
rete hub N°12, $\lambda_{2=}$-0.19712
R_c=556



rete hub N°13, $\lambda_{2=}$-0.23158
R_c=653



rete hub N°14, $\lambda_{2=}$-0.28749
R_c=811



rete hub N°15, $\lambda_{2=}$-0.39668
R_c=1119

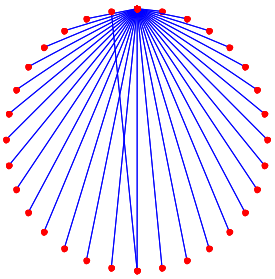


TABELLA RIASSUNTIVA

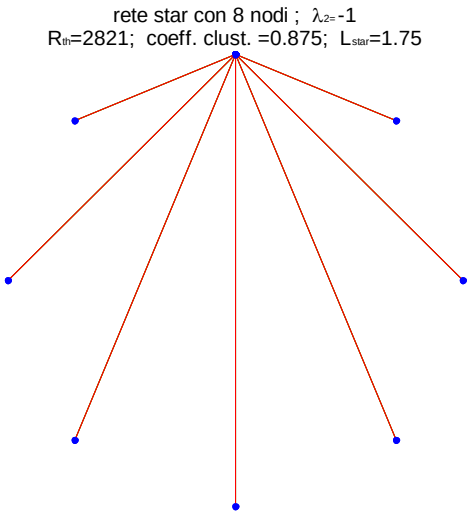
CONFIGURAZIONI	λ_2	Rth (Ω)
CONFIGURAZIONE 15	0.40	1119
CONFIGURAZIONE 14	0.29	811
CONFIGURAZIONE 13	0.23	653
CONFIGURAZIONE 12	0.20	556
CONFIGURAZIONE 11	0.17	490
CONFIGURAZIONE 10	0.16	443
CONFIGURAZIONE 9	0.14	408
CONFIGURAZIONE 8	0.14	382
CONFIGURAZIONE 7	0.13	362
CONFIGURAZIONE 6	0.12	346
CONFIGURAZIONE 5	0.12	334
CONFIGURAZIONE 4	0.12	326
CONFIGURAZIONE 3	0.11	320
CONFIGURAZIONE 2	0.11	317
CONFIGURAZIONE 1	0.11	315

In tutte queste configurazioni, notiamo che λ_2 cresce in valore assoluto man mano che le connessioni passano da un nodo hub all'altro.

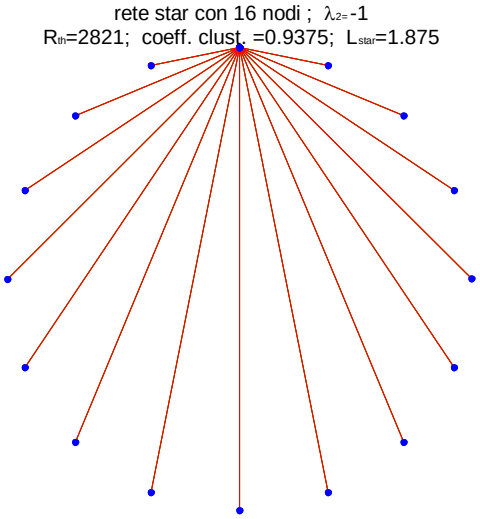
Traslando l'ultima connessione da un nodo hub all'altro, si ottiene una configurazione a stella, che come si vedrà nel paragrafo seguente, ha sempre λ_2=-1.

- **Configurazioni a stellla**

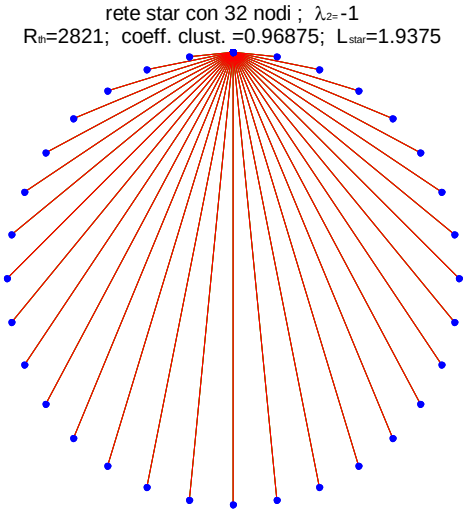
```
[A,λ2,Rth]=TutteLeConnessioni2(8,2,10e-9,100e-9,1780,6.31,'star');
```




```
[A,λ2,Rth]=TutteLeConnessioni2(16,2,10e-9,100e-9,1780,6.31,'star');
```




```
[A,λ2,Rth]=TutteLeConnessioni2(32,2,10e-9,100e-9,1780,6.31,'star');
```



Si può notare che per qualunque numero di nodi λ_2=-1.

Legenda:

λ_2=massimo autovalore non nullo della matrice di connessione.
R_{th}= limite superiore della resistenza di collegamento, per evitare la perdita della sincronizzazione
coeff. clust. = coefficiente di clustering, ottenuto con la formula : C= `(N-1)/N` [41]
L_{star}=lunghezza di path calcolata con la formula : `L=2-2*(N-1)/(N*(N-1))` [42]

[41] Cfr. Xiao Fan Wang, *Complex networks: topology, dynamics and synchronization*, international journal of bifurcation and Chaos, Vol. 12, N°5 (2002), pag. 894
[42] Vedi sopra

3.8.2 Robustezza di reti scale-free e random al distacco casuale di nodi o ad attacchi mirati

- **Esempi di distacco nodi in una rete scale-free**

Nel caso di distacco dei nodi di una rete scale-free o di una rete random, è importante verificare il comportamento del secondo autovalore maggiore, non nullo, della matrice di connessione di una rete sincronizzata, λ_2, poiché tale rete rimane sincronizzata soltanto se non si modifica λ_2, anche se ovviamente sarà diversa la nuova matrice di connessione. Naturalmente il comportamento sarà diverso per una rete di tipo scale-free, che possiede solo pochi nodi con molte connessioni,e la maggioranza di questi con poche connessioni, e la rete random, le cui connessioni sono casuali.
Iniziamo prima con il verificare il comportamento di un rete scale-ree, poiché questa configurazione è la più ricorrente nel mondo reale, a cominciare per esempio da Internet. Come già detto nel precedente capitolo, essa si ottiene partendo da pochi nodi, e gli altri che si aggiungono man mano si collegano con una probabilità molto alta ai nodi precedenti, aventi i gradi maggiori, determinando così pochi nodi con molte connessioni, e la maggior parte di essi con poche. La conseguenza, già vista, è una grande fragilità agli attacchi intenzionali, cioè diretti al distacco dei nodi con i più alti gradi, e all'opposto una grande robustezza nei confronti dei distacchi causali.
Con la funzione DistaccoNodiPerReteScaleFree(Mo,N,z,p_star)si costruisce una rete di tipo scale-free, secondo il modello di Barabasi-Albert, cioè la probabilità che un nodo si colleghi al nodo i dipende dal grado k_i di tale nodo:

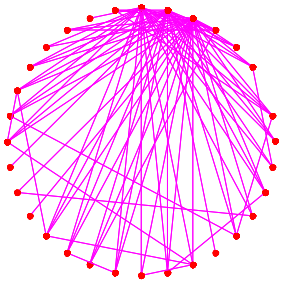
$$P_i = \frac{k_i}{\sum_j k_j}$$

Con questa funzione vediamo come si modifica λ_2 di una rete di tipo scale-free, se si staccano alcuni (z) nodi,sia partendo in ordine decrescente dal nodo di grado più alto, sia in modo casuale.
PARAMETRI DI INGRESSO:
Mo=numero di nodi iniziale
N=numero finale di nodi
z=numero di nodi staccati
p_star= probabilità che si distacchi ognuno degli z nodi
PARAMETRO DI USCITA:
As=matricediconnessione della rete iniziale di tipo scale-free
OUTPUT:
A=Matrice di incidenza;
Ap=Matrice di connessione dopo l'operazione di distacco-nodi
λ2=il più grande autovalore delle matrici di connessione,escluso lo zero
Ci=gradi di ogni nodo;
C=grado medio della rete;
cc=coefficiente di clustering;
Gradi=gradi in ordine decrescente
Nodi=rispettivi nodi
grado_max=grado massimo
nodi_hub=nodi aventi il grado massimo
N.B. I parametri di uscita che terminano con il punto e virgola, escono in output solo dopo la rimozione di questo.

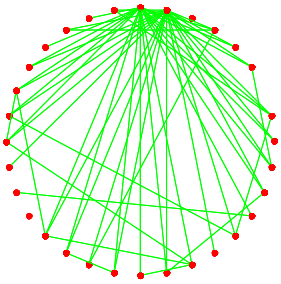
La rete di tipo scale-free dell'esempio seguente parte da tre nodi iniziali per arrivare a 32 nodi finali; si staccano cinque nodi, uno per volta , prima partendo in ordine decrescente dal nodo di grado più alto, ed in seguito si ripete l'operazione in modo casuale

Esempio: function [A] =DistaccoNodiPerReteScaleFree(3,32,5,0.3);

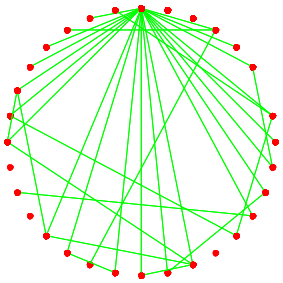
rete di tipo scale-free con 3 nodi iniziali e 32 nodi finali;
λ_2=-0.90492, CC=0.15927, nodo hub=3



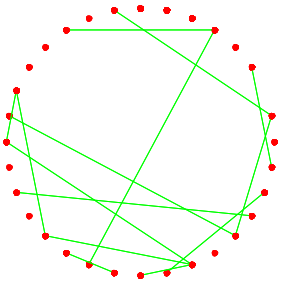
distacco di # 1 nodi in ordine decrescente di grado;
λ_2= -0.39029, CC=0.11089, nodi hub=2



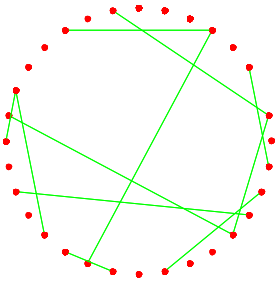
distacco di # 2 nodi in ordine decrescente di grado;
λ_2= -0.28519, CC=0.068548, nodi hub=1



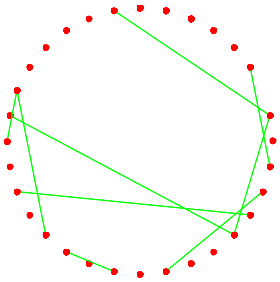
distacco di # 3 nodi in ordine decrescente di grado;
λ_2= -0.58579, CC=0.028226, nodi hub=15



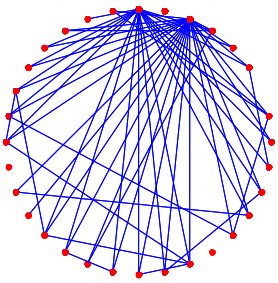
distacco di # 4 nodi in ordine decrescente di grado;
λ_2= -0.58579, CC=0.022177, nodi hub=4 8 13 27



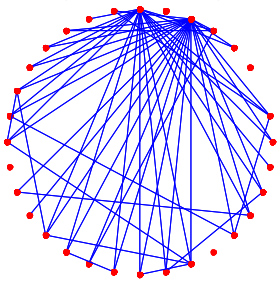
distacco di # 5 nodi in ordine decrescente di grado;
λ_2= -0.58579, CC=0.018145, nodi hub=8 13 27



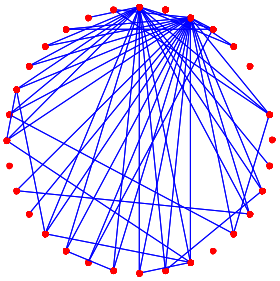
distacco di # 1 nodi in ordine casuale;
λ_2= -0.9372, CC=0.11694, nodi hub=3



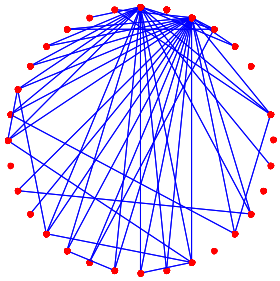
distacco di # 2 nodi in ordine casuale;
λ_2= -0.90377, CC=0.11089, nodi hub=3



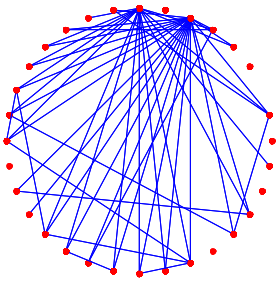
distacco di # 3 nodi in ordine casuale;
λ_2= -0.89776, CC=0.10685, nodi hub=3



distacco di # 4 nodi in ordine casuale;
λ_2= -0.89101, CC=0.10081, nodi hub=3



distacco di # 5 nodi in ordine casuale;
λ_2= -0.89101, CC=0.10081, nodi hub=3



Come si può ben vedere nell'esempio,la rete di tipo scale-free subito perde la sincronizzazione,al primo distacco di un nodo hub, mentre dopo 5 distacchi casuali è ancora sincronizzata, in perfetto accordo con quanto sostenuto da Wang e Chen sulle reti scale-free, 'robust yet fragile', cioè resistenti agli attacchi casuali, ma estremamente fragile rispetto agli attacchi intenzionali.

- **Esempi di distacco di nodi in una rete di configurazione 'random'**

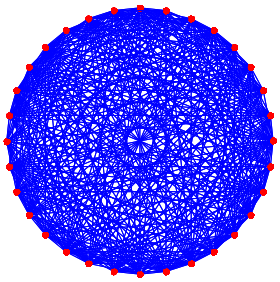
Nella rete di tipo 'random' le connessioni sono più equamente distribuite rispetto alla rete scale-free, e quindi anche il comportamento della rete di fronte al distacco dei nodi può essere alquanto diverso.
Per lo studio di questo tipo di reti si può utilizzare la funzione:
distacco_nodi_rete_random (N,delta,p,ps),
che esamina il comportamento di una rete costituita da N circuiti, in configurazione 'random', (cioè ogni connessione tra i due nodi i e j avviene con probabilità p), quando da essa si stacca una percentuale delta di nodi .Prima compaiono le reti in cui il distacco dei nodi avviene ordinatamente a partire da quello avente il grado più alto di connessioni, ed in un secondo momento i nodi sono staccati in modo casuale, per verificare la differenza tra i due casi,soprattutto in relazione alla persistenza della sincronizzazione. Questa, infatti,permane solo a condizione che non si modifichi λ_2.

Esempio:

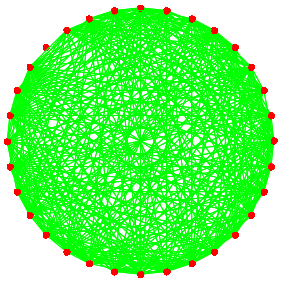
```
function [Ac] = distacco_nodi_rete_random(32,0.2,0.3,0.3)
```

Con questa funzione vediamo come si modifica λ2 in una rete di tipo random con il distacco di una percentuale delta di nodi e delle relative connessioni,sia partendo in ordine decrescente dal nodo più alto,sia in modo casuale.
PARAMETRI DI INGRESSO:
N=numero di nodi
delta=percentuale di nodi da rimuovere
p=probabilità che due nodi i e j siano connessi
ps=probabilità che un nodo sia distaccato
PARAMETRO DI USCITA:
AC=MATRICE DI CONNESSIONE
OUTPUT:
A=Matrice di incidenza;
As=Matrice di connessione dopo il distacco di un nodo
λ2 =il più grande autovalore della matrice Ac,escluso lo zero
Ci=gradi di ogni nodo
C=grado medio della rete
Gradi=gradi di ciascun nodo in ordine decrescente
Nodi=rispettivi nodi
grado_max=grado massimo
nodi_hub=nodi aventi il grado massimo
N.B. I parametri di uscita che terminano con il punto e virgola,escono in output solo dopo la rimozione di questo.

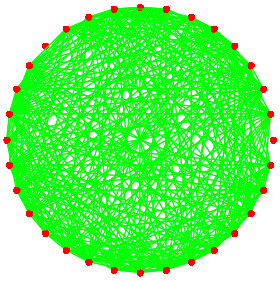
rete random con 32 nodi e probabilità p=0.5
$\lambda 2$ = -17.9962 nodo hub = 29



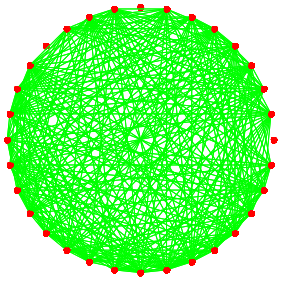
distacco di # 1 nodi in ordine decrescente di grado;
λ_2= -17.0204, CC=0.69556, nodi hub=9 27



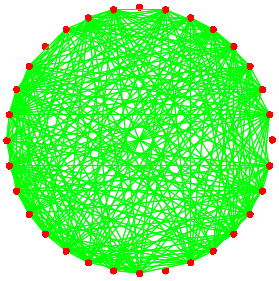
distacco di # 2 nodi in ordine decrescente di grado;
λ_2= -16.0387, CC=0.64315, nodi hub=1 27



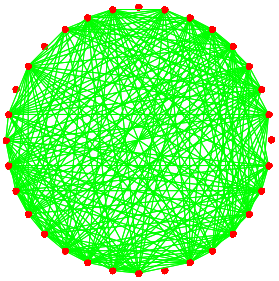
distacco di # 3 nodi in ordine decrescente di grado;
λ_2= -15.1043, CC=0.59274, nodi hub=16 27



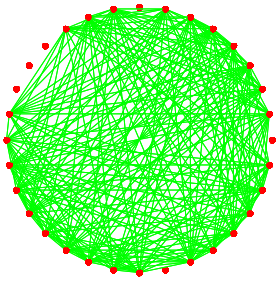
distacco di # 4 nodi in ordine decrescente di grado;
λ_2= -14.7507, CC=0.54435, nodi hub=27



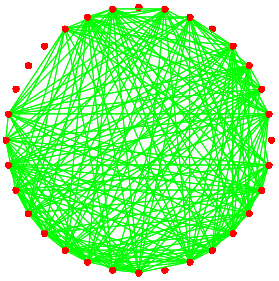
distacco di # 5 nodi in ordine decrescente di grado;
λ_2= -13.875, CC=0.49798, nodi hub=28



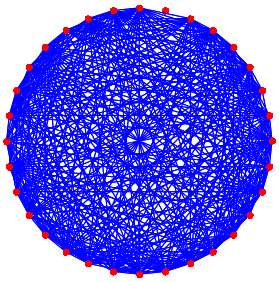
distacco di # 6 nodi in ordine decrescente di grado;
λ_2= -12.947, CC=0.45363, nodi hub=4 15 19



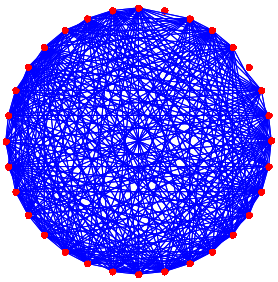
distacco di # 7 nodi in ordine decrescente di grado;
λ_2= -12.0281, CC=0.41331, nodi hub=15 19 31



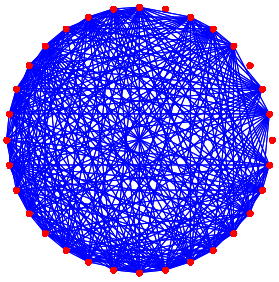
distacco di # 1 nodi in ordine casuale;
λ_2= -17.1604, CC=0.70565, nodi hub=29



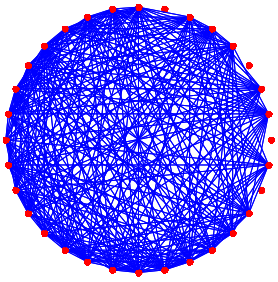
distacco di # 2 nodi in ordine casuale;
λ_2= -16.8969, CC=0.65927, nodi hub=29



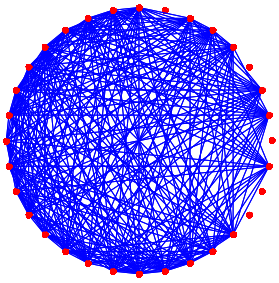
distacco di # 3 nodi in ordine casuale;
λ_2= -15.9212, CC=0.60887, nodi hub=29



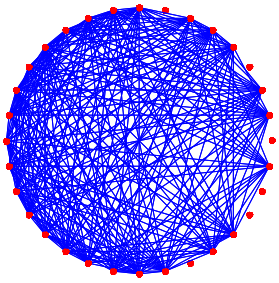
distacco di # 4 nodi in ordine casuale;
λ_2= -15.6872, CC=0.56855, nodi hub=29



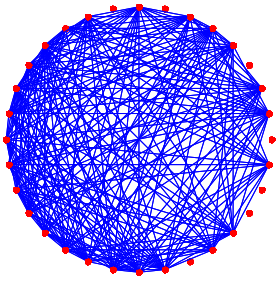
distacco di # 5 nodi in ordine casuale;
λ_2= -15.0243, CC=0.53024, nodi hub=29



distacco di # 6 nodi in ordine casuale;
λ_2= -14.4576, CC=0.48992, nodi hub=27 29



distacco di # 7 nodi in ordine casuale;
λ_2= -13.5652, CC=0.4496, nodi hub=27 29



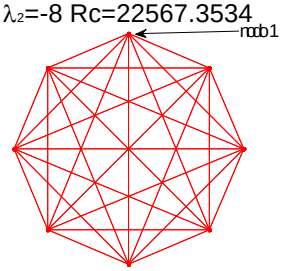
3.11 Esempi di pinning

1) rete globally connected con otto nodi

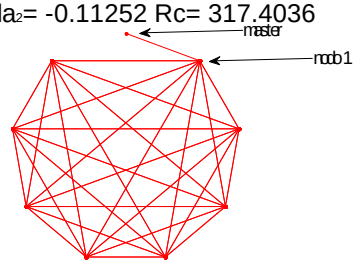
```
A =[ -7     1     1     1     1     1     1     1;
      1    -7     1     1     1     1     1     1;
      1     1    -7     1     1     1     1     1;
      1     1     1    -7     1     1     1     1;
      1     1     1     1    -7     1     1     1;
      1     1     1     1     1    -7     1     1;
      1     1     1     1     1     1    -7     1;
      1     1     1     1     1     1     1    -7];

partendo_dalla_matrice(A,10e-9,100e-9,1780,6.31,8,'pinning');
```

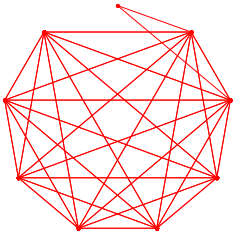
Rete con 8 connessioni
λ_2=-8 Rc=22567.3534



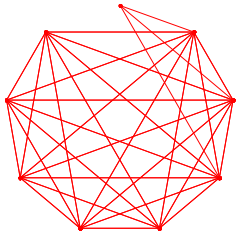
rete con numero 1 di nodi pinnati
lambda$_2$= -0.11252 Rc= 317.4036



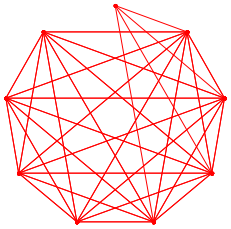
rete con numero 2 di nodi pinnati
lambda$_2$= -0.228 Rc= 643.1643



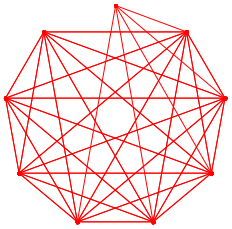
rete con numero 3 di nodi pinnati
lambda$_2$= -0.34669 Rc= 977.979



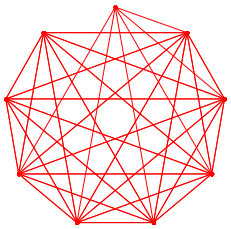
rete con numero 4 di nodi pinnati
lambda$_2$= -0.46887 Rc= 1322.6475



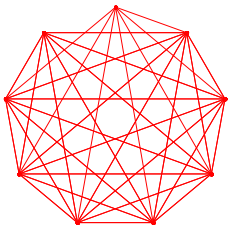
rete con numero 5 di nodi pinnati
lambda$_2$= -0.59488 Rc= 1678.0948



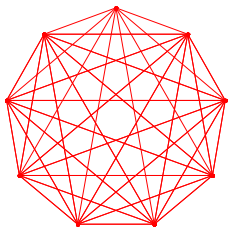
rete con numero 6 di nodi pinnati
lambda$_2$= -0.72508 Rc= 2045.3999



rete con numero 7 di nodi pinnati
lambda$_2$= -0.85995 Rc= 2425.8355



rete con numero 8 di nodi pinnati
lambda$_2$= -1 Rc= 2820.9192

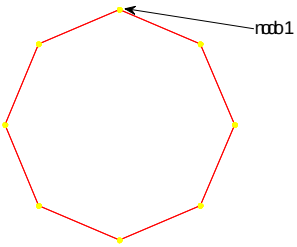


2) Rete nearest-neighbor coupled

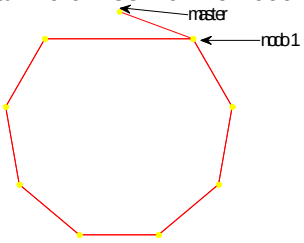
```
A =[  -2     1     0     0     0     0     0     1
       1    -2     1     0     0     0     0     0
       0     1    -2     1     0     0     0     0
       0     0     1    -2     1     0     0     0
       0     0     0     1    -2     1     0     0
       0     0     0     0     1    -2     1     0
       0     0     0     0     0     1    -2     1
       1     0     0     0     0     0     1    -2];

partendo_dalla_matrice(A,10e-9,100e-9,1780,6.31,8,'pinning');
```

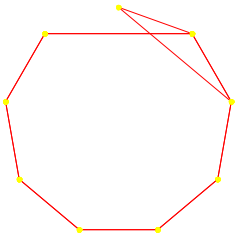
Rete con 8 connessioni
λ_2=-0.58579 Rc=1652.4562



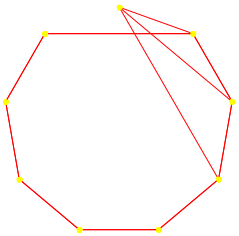
rete con numero 1 di nodi pinnati
lambda$_2$= -0.07253 Rc= 204.6007



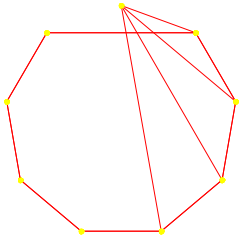
rete con numero 2 di nodi pinnati
lambda$_2$= -0.12061 Rc= 340.2445



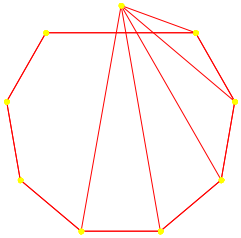
rete con numero 3 di nodi pinnati
lambda$_2$= -0.17056 Rc= 481.1408



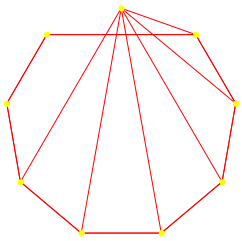
rete con numero 4 di nodi pinnati
lambda$_2$= -0.23599 Rc= 665.6948



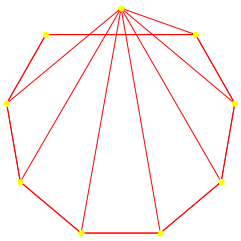
rete con numero 5 di nodi pinnati
lambda$_2$= -0.33356 Rc= 940.9375



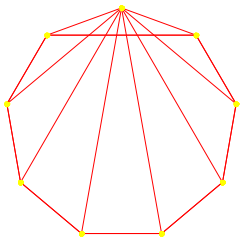
rete con numero 6 di nodi pinnati
lambda$_2$= -0.49152 Rc= 1386.5348



rete con numero 7 di nodi pinnati
lambda$_2$= -0.74673 Rc= 2106.4604



rete con numero 8 di nodi pinnati
lambda$_2$= -1 Rc= 2820.9192



- **Esempi di pinning per una rete di tipo scale-free**

Quando si collega un circuito 'master' ad una rete, per indirizzarne i comportamenti, è preferibile collegarlo ai nodi che hanno un più alto numero di connessioni.[43]Con la seguente funzione è possibile studiare a quali e quanti nodi della rete si deve connettere il circuito 'master' per ottenere il migliore risultato ai fini della sincronizzazione:

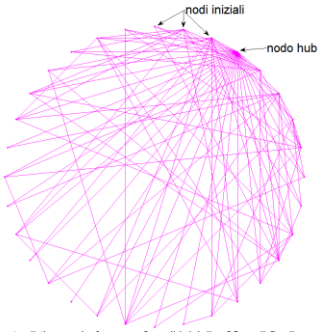
PinningPerReteSCaleFree(3,20,3)
Questa funzione costruisce una rete di tipo scale-free secondo il modello di Barabasi-Albert, cioè la probabilità che il nodo i si colleghi al nodo j è uguale a:

$$P_i = \frac{k_i}{\sum_j k_j}$$

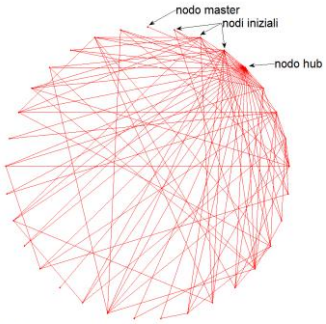
Possiamo vedere anche come si modifica λ_2 di una rete di tipo scale-free con l'aggiunta alla rete iniziale di un circuito 'master', collegato in modo unidirezionale solo ad alcuni nodi,sia partendo in ordine decrescente dal nodo con il grado più alto, sia partendo in ordine crescente dal nodo con il grado più basso della rete, sia in modo casuale. In questa configurazione i nodi iniziali sono 3, i nodi finali sono 30, i nodi da "pinnare" sono tre.

[43] Wang e Chen op.cit.

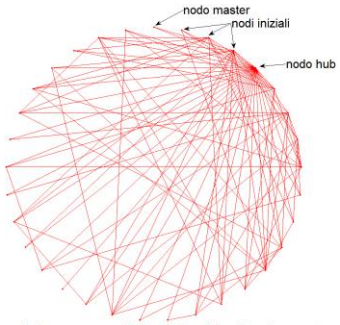
PARAMETRI DI INGRESSO:
Mo=numero di nodi iniziale
N=numero di nodi finale
z=numero di nodi con i quali il'Master' si collega(in modalità unidirezionale)
p= probabilità che due nodi della rete si colleghino tra di loro
PARAMETRO DI USCITA:
As=matrice di connessione della rete iniziale di tipo scale-free
OUTPUT: A=Matrice di incidenza;
Ap=Matrice di connessione dopo l'operazione di pinning
λ2 =il più grande autovalore delle matrici di connessione,escluso lo zero
Ci=gradi di ogni nodo;
C=grado medio della rete;
Gradi=gradi in ordine decrescente
Nodi=rispettivi nodi
grado max=grado massimo
nodi_hub=nodi aventi il grado massimo
N.B. I parametri di uscita che terminano con il punto e virgola,escono in output solo dopo la rimozione di questo
Esempio: function [A] = ReteScaleFree(3,30,3,0.3);
Si ottengono dalla rete iniziale tre reti con tre nodi pinnati secondo i criteri esposti sopra



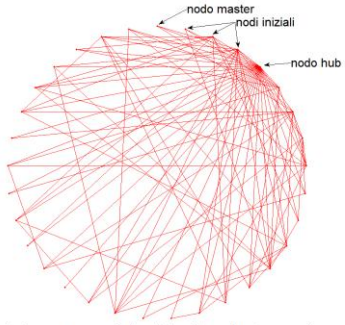
rete di tipo scale-free con 3 nodi iniziali e 32 nodi finali.
Lambda2= -1



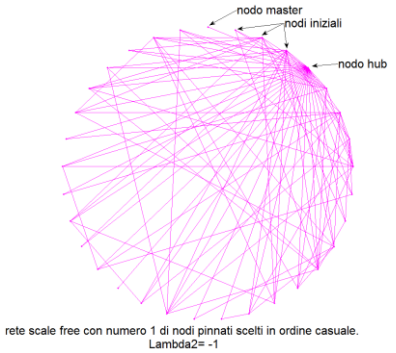
rete scale free con numero 1 di nodi pinnati in ordine decrescente.
Lambda2= -1

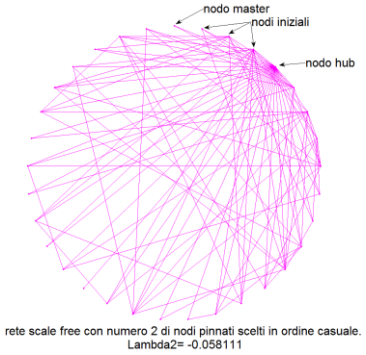


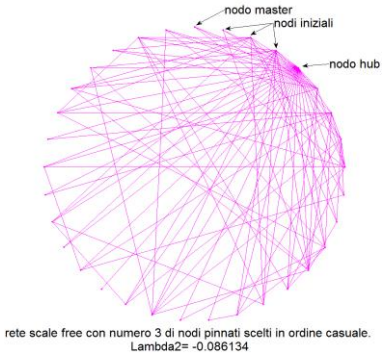
rete scale free con numero 2 di nodi pinnati in ordine decrescente.
Lambda2= -0.059354



rete scale free con numero 3 di nodi pinnati in ordine decrescente.
Lambda2= -0.088202







Esempio2 : DistaccoNodiPerReteScaleFree(3,100,5,0.3)

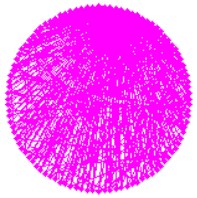
Consideriamo adesso come varia l'autovalore λ_2, se si distaccano cinque nodi, con le relative connessioni, da una rete di tipo scale-free, con tre nodi iniziali e cento finali (vedi esempio pagina seguente).

Anche in questo caso partiamo prima dal distacco dei nodi in ordine decrescente di grado, cominciando da quello con il grado maggiore, e poi dal distacco casuale di questi.

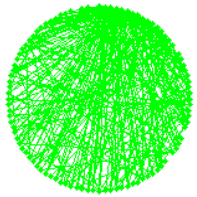
Come nell'esempio precedente, l'autovalore λ_2 si modifica notevolmente nel primo caso considerato (distacco dei nodi in ordine decrescente di grado), mentre nel secondo caso (distacco casuale) si ha una brusca variazione di λ_2 con il distacco del primo nodo, mentre l'autovalore permane quasi identico per gli ultimi quattro nodi staccati.

Si deve ritenere che per un numero N molto più grande (nell'ordine dei milioni o più di nodi), il distacco di una bassa percentuale di questi sia del tutto ininfluente sulla sincronizzazione della rete, sia in un caso che nell'altro.

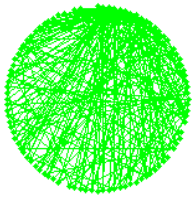
rete di tipo scale-free con 3 nodi iniziali e 100 nodi finali;
lambda2=-1



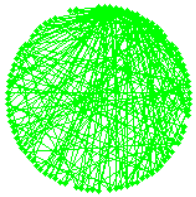
distacco di 1 nodi in ordine decrescente di grado;
lambda2= -0.64655



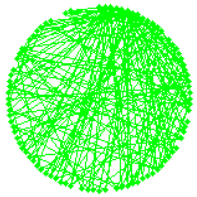
distacco di 2 nodi in ordine decrescente di grado;
lambda2= -0.27353



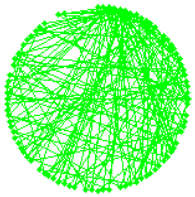
distacco di 3 nodi in ordine decrescente di grado;
lambda2= -0.16862



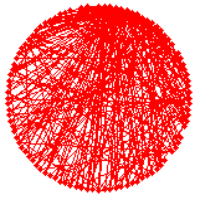
distacco di 4 nodi in ordine decrescente di grado;
lambda2= -0.10942



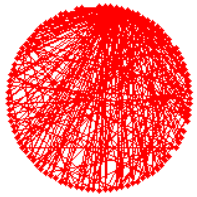
distacco di 5 nodi in ordine decrescente di grado;
lambda2= -0.10947



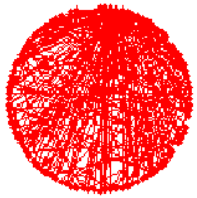
distacco di 1 nodo in ordine casuale;
lambda2= -0.64655



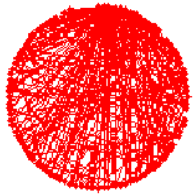
distacco di 2 nodi in ordine casuale;
lambda2= -0.63968



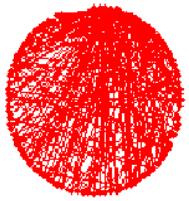
distacco di 3 nodi in ordine casuale;
lambda2= -0.63531



distacco di 4 nodi in ordine casuale;
lambda2= -0.63862



distacco di 5 nodi in ordine casuale;
lambda2= -0.63862



APPENDICE

Listati Matlab

La funzione **MetodoCartaPenna2 (x,C1,C2,R,Klim),** utilizzata dai programmi di confronto tra reti, (cfr. pag.68-73), è stata unificata con la funzione **partendo_dalla_matrice(A,C1,C2,R,Klim,z,tipo)**, nell'unica funzione **metodocartapenna1(V,C1,C2,R,Klim, tipo,tipo2)** in cui con **'tipo'** si sceglie tra la partenza da un 'vettore', e la partenza da una 'matrice' quadrata, sia di incidenza che di connessione, con **'tipo2'** si sceglie tra 'pinning','distacco nodi' oppure '0' , con cui si termina il programma. Ulteriori spiegazioni (parametri di ingresso,uscita,output) si troveranno all'interno della funzione.

La funzione **TutteLeConnessioni (N,k,C1,C2,R,Klim,tipo)** è stata utilizzata alle pag.73-84.

Per quanto riguarda le funzioni utilizzate, riguardanti reti di tipo random e scale-free, sono state tutte unificate nell'unica funzione **reti_irregolari(N, p,p_star,tipo,tipo2).**

Quasi tutte queste funzioni hanno utilizzato a loro volta le funzioni **TrovaLambda2** (V) o **Trova-Lambda2bis(V),** che restituiscono il maggiore autovalore non nullo della matrice di connessione**,** λ_2.Si sono esclusi, in queste funzioni, oltre all'autovalore nullo, anche gli autovalori compresi nell'intervallo [-0.05 , 0.05], che sono stati posti uguali a O.

```
function[A,lambda2,Rth]=TutteLeConnessioni2(N,k,C1,C2,R,Klim,tipo)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%PARAMETRI DI INGRESSO:
%N,numero intero positivo, è il numero dei nodi della rete(N>=3);
%k,numero intero pari,< N-1,è il numero di connessioni per ogni nodo,si
%utilizza solo nel tipo 'NN'.Attenzione:
%La funzione considera errore,e si blocca, per k>=N e per k dispari;
%C1=capacità del primo condensatore del circuito di Chua;
%C2=capacità del secondo condensatore del circuito di Chua;
%R=resistenza variabile del circuito di Chua;
%Klim=parametro ricavato dalla MSF;
%I parametri C1,C2,,R,Klim,sono utilizzati per trovare:
%Rc=abs((lambda2/Klim)*(C2*R/C1))
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%tipo:
%'NN',rete iniziale nearest neighbor con k connessioni per lato ed N
%nodi,alla quale si aggiunge una connessione alla volta, fino ad arrivare
%alla rete di tipo globally connected
%'hub' rete di tipo hub, inizialmente simmetrica con due nodi hub; le
%connessioni di un nodo vengono di volta in volta spostate all'altro nodo
%'star',rete con connessione a stella
%triangoli1:triangoli in serie di un circuito aperto
%triangoli2
%triangoli3
%quadrilateri
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%PARAMETRO DI USCITA:
%A=matrice di connessione
%OUTPUT:
%lambda2=maggiore autovalore diverso da zero della matrice di connessione
%Ci=gradi di ogni nodo della rete
%C=grado medio della rete
%grado_max=grado massimo
%nodi_hub= nodi che hanno il grado massimo
%Rc=valore massimo della resistenza  di collegamento tra i vari circuiti
%della rete
%Il programma,per il tipo 'NN',fa anche un confronto tra le reti,per
%evidenziarne la migliore ai fini della sincronizzazione,cioè quella con
%lambda2 ,in valore assoluto,maggiore.


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
      switch tipo
          case 'NN'

              if k>=N || k<2 ||rem(k,2)~=0
     errordlg('errore:k deve essere un numero pari<N:','File Error');
              return
              end
       q=N*(N-1)/2-(k*N/2-1)
       A=zeros (N,N);
  for i=1:k/2
    diagonale=ones(1,N-i);
    lato=ones(1,i);
    A=A+diag(diagonale,-i)+diag(diagonale,i)+diag(lato,N-i)+diag(lato,-N+i);
  end
  disp('MATRICE DI CONNESSIONE')
   D=diag(sum(A,2));
   A=A-D
  autovaloridiA=eig(A)'
```

```
   B=A;
   V=diag(autovaloridiA);
   lambda2=TrovaLambda2(V);
   Rc=abs((lambda2/Klim)*(C2*R/C1));
   Rc=round(Rc)
   Ci=-diag(A)'  %Ci=gradi di ogni nodo;C=grado medio della rete
   C=sum (Ci)/N
   disp('COEFFICIENTE DI CLUSTERING:')
   cc=C/(N-1)
 disp('GRADI IN ORDINE DECRESCENTE; NODI IN ORDINE DECRESCENTE DI GRADO')
   [Gradi,Nodi]=sort(Ci,'descend')
    grado_max=max(Ci)
    Z=diag(Ci);
    [x,x]=find(Z==grado_max);
   nodihub=x';
            if length(x')==N
                disp('Rete regolare;tutti i nodi hanno lo stesso grado')
                str=('rete regolare');
            else
               disp('nodi_hub=x')
               str=(['nodihub = ', num2str(nodi_hub)]);
            end
 %str=num2str(nodihub);
%plotting
t=0:2*pi/N:2*pi;
xy=[sin(t);cos(t)]';
gplot(A,xy,'-b.');
set(findobj('Type','Line'),'MarkerEdgeColor','r','MarkerSize',15,'LineWidth',0.0
8);
title(['\fontsize{12}rete iniziale con ',num2str(N),' nodi e ',num2str(k),....
' connessioni per nodo',10,' \lambda\fontsize{6}2
\fontsize{12}=',num2str(lambda2),....
'; R\fontsize{6}c\fontsize{12}=',num2str(Rc),'; coeff. clust.
=',num2str(cc),10,' ',str])
axis('equal','off')
hold off
pause
Ci=zeros(1,N)';
C=0;
nuovarete=1;
lambda=zeros(1,q);
lambda(1)=lambda2;
Rth=zeros(1,q);
Rth(1)=Rc;
CC=zeros(1,q);
CC(1)=cc;
for r=2:q-1
     for i=1:N
          for j=1:N
              if (j~=i)&& (A(i,j)~=1)
                          A(i,j)=1;
                          A(j,i)=A(i,j);
                          nuovarete=nuovarete+1
                          D=diag(sum(A,2));
                          Ac=A-D
                          autovaloridiAc=eig(Ac)'
                          V=diag(autovaloridiAc);
                          lambda2=TrovaLambda2(V);
                          Ci=-diag(Ac)'
                          C=sum(Ci)/N
                          disp('COEFFICIENTE DI CLUSTERING:')
                          cc=C/(N-1)
   disp('GRADI IN ORDINE DECRESCENTE; NODI IN ORDINE DECRESCENTE DI GRADO')
```

```
                            [Gradi,Nodi]=sort(Ci,'descend')
                            grado_max=max(Ci)
                            Z=diag(Ci);
                            [x,x]=find(Z==grado_max);
                            %disp('NODO HUB')
                            nodi_hub=x';
                            if length(x)==N
                               disp('Rete regolare;tutti i nodi hanno lo stesso
grado')
                               str=('rete regolare')
                             else
                                disp('nodi_hub=x')
                                str=(['nodihub = ', num2str(nodi_hub)]);
                            end
                            %str=num2str(nodi_hub);
                            Rc=abs((lambda2/Klim)*(C2*R/C1));
                            Rc=round(Rc)
                            lambda(r)=lambda2;
                            Rth(r)=Rc;
                            CC(r)=cc;
                            r=r+1;
                            gplot(Ac,xy,'-b.');

set(findobj('Type','Line'),'MarkerEdgeColor','r','MarkerSize',15,'LineWidth',0.0
8);
                            title(['\fontsize{12}rete N°',num2str(nuovarete),'
\lambda\fontsize{6}2= \fontsize{12}',....
                            num2str(lambda2),';
R\fontsize{6}c\fontsize{12}=',num2str(Rc),10,'coeff. clust. =',num2str(cc),10,'
',str])
                            axis('equal','off')
                            hold off
                            pause

                end
            end
        end
end

disp('lambda2 per tutte le configurazioni esaminate')

    lambda2=lambda
    disp('resistenze di soglia per tutte le configurazioni esaminate in ordine
decrescente:')
    Rth=sort(Rth,'descend');
    Rth=round(Rth)
    disp('tutti gli autovalori lambda2, in valore assoluto, delle configurazio-
ni, in ordine decrescente e le relative configurazioni:')
    [lambda2,CONFIGURAZIONE]=sort(abs(lambda2),'descend')
    disp(['la configurazione migliore con il maggiore valore assoluto di lamb-
da2, quindi con la maggiore Rth è la n° ',num2str(CONFIGURAZIONE(1))])
    disp('TABELLA RIASSUNTIVA ')
    disp('                                                        ')
   disp(['CONFIGURAZIONI           LAMBDA2       Rth        Coefficiente di clu-
stering   '])
    disp('                                                       ')
 for i=1:q
     st=sprintf('CONFIGURAZIONE %1.0f \t %10.2f\t %10.0f\t %10.2f\n
',CONFIGURAZIONE(i),lambda2(i),Rth(i),CC(CONFIGURAZIONE(i)));
     disp(st)
```

```
     %disp(['CONFIGURAZIONE ',num2str(CONFIGURAZIONE(i)),'
',num2str(lambda2(i)),'        ',num2str(Rth(i))])
 end



%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    case 'hub'
 if rem(N,2)==0
    else
    errordlg('errore:N deve essere un numero pari>=2','File Error');
end
     A=zeros(N);
 for i=2:(N/2+1)
          A(i,1)=1;
          A(1,i)=1;
          for i=(N/2+2):N
              A(i,N/2+1)=1;
              A(N/2+1,i)=1;
          end
  end
D=diag(sum(A,2));
Ac=A-D
B=A;
autovaloridiA=eig(Ac)'
V=diag(autovaloridiA)';
lambda2=TrovaLambda2(V);
disp ('Ci=gradi di ogni nodo,C=grado medio della rete')
Ci=-diag(Ac)'
C=sum(Ci)/N
[Gradi,Nodi]=sort(Ci,'descend')
grado_max=max(Ci)
Z=diag(Ci);
[x,x]=find(Z==grado_max);
%disp('NODO HUB')
nodihub=x'  %nodi hub
Rc=abs((lambda2/Klim)*(C2*R/C1));
Rc=round(Rc)
%plotting
t=0:2*pi/N:2*pi;
xy=[sin(t);cos(t)]';
gplot(B,xy,'-r.')
set(findobj('Type','Line'),'MarkerEdgeColor','b','MarkerSize',15,'LineWidth',1);
title(['\fontsize{16}rete hub iniziale con ',num2str(N),' nodi ;
\lambda\fontsize{8}','2 \fontsize{16}=',num2str(lambda2),10,'
R\fontsize{8}c\fontsize{16}=',num2str(Rc)])
%title(['rete hub iniziale con nodi ' int2str(N),'  lambda2=  ',
num2str(lambda2)])
axis('square','off')
hold off
 pause

q=(N-2)/2-1;
nuovarete=1;
lambda=zeros(1,q);
lambda(1)=lambda2;
Rth=zeros(1,q);
Rth(1)=Rc;
     for i=1:q
        B(N/2+1,N-i)=0;
        B(N-i,N/2+1)= 0;
```

```
        B(N-i,1)=1;
        B(1,N-i)=1;
                nuovarete=nuovarete+1
                D=diag(sum(B,2))
                Ac=B-D
                autovaloridiA=eig(Ac)'
                V=autovaloridiA;
                lambda2=TrovaLambda2bis(V);
                lambda(nuovarete)=lambda2
                Rc=abs((lambda2/Klim)*(C2*R/C1));
                Rc=round(Rc)
                Rth(nuovarete)=Rc;
                disp ('Ci sono i gradi di ogni nodo,C è il grado medio della re-
te')
                Ci=-diag(Ac)'
                C=sum (Ci)/N
  disp('GRADI IN ORDINE DECRESCENTE; NODI IN ORDINE DECRESCENTE DI GRADO')
               [Gradi,Nodi]=sort(Ci,'descend')
               grado_max=max(Ci)
               Z=diag(Ci);
               [x,x]=find(Z==grado_max);
                nodihub=x'  %nodi hub
                t=0:2*pi/N:2*pi;
                xy=[sin(t);cos(t)]';
                gplot(Ac,xy,'-b.')

set(findobj('Type','Line'),'MarkerEdgeColor','r','MarkerSize',15,'LineWidth',1);
                title(['\fontsize{16}rete hub N°',num2str(nuovarete),',
\lambda\fontsize{8}2 \fontsize{16}=',num2str(lambda2),10,'
R\fontsize{8}c\fontsize{16}=',num2str(Rc)])
                %title(['rete hub numero ' int2str(nuovarete),'  lambda2=  ',
num2str(lambda2)])
                axis('square','off')
                hold off
                pause

 end
disp('lambda2 per tutte le configurazioni esaminate')
    lambda2=lambda
    disp('resistenze di soglia per tutte le configurazioni esaminate in ordine
decrescente')
    Rth=sort(Rth,'descend');
    Rth=round(Rth)
    %lambda2=sort(lambda)
    disp('tutti gli autovalori lambda2, in valore assoluto, delle configurazio-
ni, in ordine decrescente e le relative configurazioni')
    [lambda2,CONFIGURAZIONE]=sort(abs(lambda2),'descend')
    disp(['la configurazione migliore con il maggiore valore assoluto di lamb-
da2, quindi con la maggiore Rth è la n° ',num2str(CONFIGURAZIONE(1))])
    disp('TABELLA RIASSUNTIVA ')
    disp(['CONFIGURAZIONI           LAMBDA2       Rth'])
    disp('                                                  ')
 for i=1:q+1
     st=sprintf('CONFIGURAZIONE %1.0f \t %10.2f\t
%10.0f\n',CONFIGURAZIONE(i),lambda2(i),Rth(i));
     disp(st)
     %disp(['CONFIGURAZIONE ',num2str(CONFIGURAZIONE(i)),'
',num2str(lambda2(i)),'          ',num2str(Rth(i))])
 end

 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
    case 'star'
 disp('lunghezza media di path')
 L=2-2*(N-1)/(N*(N-1))
 disp('coeficiente di clustering')
 cc=(N-1)/N


 A=zeros(N);
 for i=1:N
            A(1,i)=1;
            A(i,1)=1;
            A(1,1)=0;
 end

D=diag(sum(A,2));
Ac=A-D
t=0:2*pi/N:2*pi;
xy=[sin(t);cos(t)]';
gplot(Ac,xy,'-g.');
hold on
autovaloridiAc=eig(Ac)'
V=diag(autovaloridiAc);
lambda2=TrovaLambda2(V);
Rc=abs((lambda2/Klim)*(C2*R/C1));
Rc=round(Rc)
Rth=Rc;
Ci=-diag(Ac)'   %Ci sono i gradi di ogni nodo;C è il grado medio della rete
C=sum (Ci)/N
disp('GRADI IN ORDINE DECRESCENTE; NODI IN ORDINE DECRESCENTE DI GRADO')
[Gradi,Nodi]=sort(Ci,'descend')
grado_max=max(Ci)
Z=diag(Ci);
[x,x]=find(Z==grado_max);
nodi_hub=x  %nodi hub
%plotting
t=0:2*pi/N:2*pi;
xy=[sin(t);cos(t)]';
gplot(A,xy,'-g.');
%title(['rete star con nodi ' int2str(N)])
title(['\fontsize{24}rete star con ',num2str(N),' nodi ;
\lambda\fontsize{12}',....
    '2=
\fontsize{24}',num2str(lambda2),10,'R\fontsize{12}th\fontsize{24}=',num2str(Rth)
,....
    ';  coeff. clust. =',num2str(cc),';
L\fontsize{12}star\fontsize{24}=',num2str(L)])
set(findobj('Type','Line'),'MarkerEdgeColor','b','MarkerSize',18,'LineWidth',0.0
8);
axis('square','off')
hold off
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

          case 'triangoli1'

%Con il tipo 'triangoli1' si crea una rete aperta di triangoli
%con N nodi se N è dispari, N+1 se N è pari.
%N deve essere un numero intero multiplo di 3.
%Il numero di triangoli che si ottiene è uguale a  N/2-1 se N
%è pari,al più piccolo intero vicino a N/2 se N è dispari
```

```
if rem(N,3)~=0||N==3
    errordlg('errore:N deve essere un multiplo di 3 e N>3','File Error');
    return
end
if rem(N/3,2)==0
    N=N+1
end
z=floor(N/2);
numero_triangoli=z
A=zeros(N,N);
A(1,2)=1;
A(2,1)=1;
A(N,N-1)=1;
A(N-1,N)=1;

for i=3:N-2
     for j=3:N-2
         if i~=j && rem(i,2)~=0
                A(i,i+1)=1;
                A(i+1,i)=1;
                A(i,i-1)=1;
                A(i-1,i)=1;
                A(i,i+2)=1;
                A(i+2,i)=1;
                A(i,i-2)=1;
                A(i-2,i)=1;
         end
      end
 end

D=diag(sum(A,2));
disp('matrice di connessione')
Ac=A-D
autovalori=eig(Ac)'
V=diag(autovalori);
lambda2=TrovaLambda2(V);
Rth=C2*R*abs(lambda2)/(C1*Klim);
Rc=round(Rth)
disp('Ci sono i gradi di ogni nodo')
Ci=-diag(Ac)'
disp('C è il grado medio della rete')
C=sum(Ci)/N
disp('gradi in ordine decrescente e relativi nodi')
[gradi,nodi]=sort(Ci,'descend')
grado_max=max(Ci)
Z=diag(Ci);
[x,x]=find(Z==grado_max);
nodo_hub=x'

%plotting
t=0:2*pi/N:2*pi;
xy=[sin(t);cos(t)]';
gplot(Ac,xy,'-r.')
axis('square','off')
set(findobj('Type','Line'),'MarkerEdgeColor','k','MarkerSize',15,'LineWidth',0.0
8);
title(['\fontsize{16}',num2str(z),' triangoli con ',num2str(N),' nodi
\lambda\fontsize{8}',....
'2\fontsize{16}=',num2str(lambda2),10,'R\fontsize{8}c\fontsize{16}=',num2str(Rc)
,' nodi hub= ',num2str(nodo_hub)])
hold off
```

```
pause


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

          case 'triangoli2'
 %il tipo 'triangoli2' crea una rete aperta di N/3-1 triangoli ed N-1 nodi.
% N deve essere un numero itero multiplo di 3.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if rem(N,3)~=0 || N==3
    errordlg('errore:N deve essere un multiplo di 3 e N>3','File Error');
    return
end
z=N/3-1;
numero_triangoli=z
A=zeros(N-1,N-1);

for k=0:N/2
    for i=1:N-2
     for j=1:N-2

         if i~=j && i== 2+k*3 && 1<=N-3
                A(i,i+1)=1;
                A(i+1,i)=1;
                A(i,i-1)=1;
                A(i-1,i)=1;
                A(i,i+2)=1;
                A(i+2,i)=1;
         else if i~=j && i== 4+k*3 && 1<=N-5

                A(i,i+1)=1;
                A(i+1,i)=1;
                A(i,i-1)=1;
                A(i-1,i)=1;
                A(i,i-2)=1;
                A(i-2,i)=1;
             end
         end
      end
    end
 end

D=diag(sum(A,2));
disp('matrice di connessione')
At=A-D
Ac=At;
autovalori=eig(At)'
V=diag(autovalori);
lambda2=TrovaLambda2(V);
Rc=C2*R*abs(lambda2)/(C1*Klim);
Rth=round(Rc)
disp('Ci=gradi di ogni nodo')
Ci=-diag(At)'
disp('C=grado medio della rete')
C=sum(Ci)/N
disp('gradi in ordine decrescente e relativi nodi')
[gradi,nodi]=sort(Ci,'descend')
grado_max=max(Ci)
Z=diag(Ci);
[x,x]=find(Z==grado_max);
```

```
nodo_hub=x'
%plotting
t=0:2*pi/N:2*pi;
xy=[sin(t);cos(t)]';
gplot(At,xy,'-b.')
set(findobj('Type','Line'),'MarkerEdgeColor','k','MarkerSize',15,'LineWidth',0.0
8);
title(['\fontsize{16}',num2str(z),' triangoli con ',num2str(N-1),' nodi;
\lambda\fontsize{8}',....
'2\fontsize{16}=',num2str(lambda2),10,'R\fontsize{8}c\fontsize{16}=',num2str(Rth
),'; nodi hub= ',num2str(nodo_hub)])
axis('square','off')
hold off

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

          case 'quadrilateri'
%%il tipo 'quadrilateri' crea una rete chiusa di quadrilateri .
%N deve essere un numero positivo multiplo di 4. Il numero di quadrilateri che
si ottiene
%è'uguale a N/4, il numero di nodi è uguale a N+2
%
disp('N deve essere un numero intero multiplo di 4.')
if rem(N,4)~=0
    errordlg('errore: N essere un multiplo di 4','File Error')
    return
end
disp('numero di quadrilateri')

z=N/4;
disp('numero di nodi')
N=N+2

%numero_rettangoli=z

A=zeros(N,N);  %matrice di incidenza
A(N,N-1)=1;
A(N-1,N)=1;
for k=0:floor(N-4/4)
    for i=2:N
     for j=2:N

          if i~=j && i==2+4*k && i<=N-4
                A(i,i-1)=1;
                A(i-1,i)=1;
                A(i,i+1)=1;
                A(i+1,i)=1;
                A(i,i+3)=1;
                A(i+3,i)=1;

         else if i~=j && i==4+k*4 && i<=N-2
                A(i,i-1)=1;
                A(i-1,i)=1;
                A(i,i+1)=1;
                A(i+1,i)=1;

             end
         end
      end
    end
```

```
 end

D=diag(sum(A,2));
disp('matrice di connessione')
At=A-D
autovalori=eig(At)'
V=diag(autovalori);
lambda2=TrovaLambda2(V);
Rth=C2*R*abs(lambda2)/(C1*Klim);
Rc=round(Rth)
disp('Ci = gradi di ogni nodo')
Ci=-diag(At)'
disp('C = grado medio della rete')
C=sum(Ci)/N
disp('gradi in ordine decrescente e relativi nodi')
[gradi,nodi]=sort(Ci,'descend')
grado_max=max(Ci)
Z=diag(Ci);
[x,x]=find(Z==grado_max);
nodo_hub=x'
  %plotting
t=0:2*pi/N:2*pi;
xy=[sin(t);cos(t)]';
gplot(At,xy,'-r.')
axis('square','off')
set(findobj('Type','Line'),'MarkerEdgeColor','k','MarkerSize',15,'LineWidth',0.0
8);
title(['\fontsize{16}',num2str(z),' quadrilateri con ',num2str(N),' nodi;
\lambda\fontsize{8}',....
'2\fontsize{16}=',num2str(lambda2),10,'R\fontsize{8}c\fontsize{16}=',num2str(Rc)
,'; nodi hub= ',num2str(nodo_hub)])
hold off
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
case 'triangoli3'
%%il tipo 'triangoli3' crea una rete chiusa ditriangoli .
%N deve essere un numero positivoedN>=3. Il numero ditriangoli che si ottiene
%è'uguale a N/2, il numero di nodi è uguale a N.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

if rem(N,2)~=0 || N<3
    errordlg('errore:N deve essere un numero pari e N>=3','File Error');
    return
end
k=2;
%matrice di incidenza
 A=zeros (N,N);
  for i=1:k/2
    diagonale=ones(1,N-i);
    lato=ones(1,i);
    A=A+diag(diagonale,-i)+diag(diagonale,i)+diag(lato,N-i)+diag(lato,-N+i);
  end


for i=1:N-3
   %for j=1:N-3
       if  rem(i,2)~=0
           A(i,i+2)=1;
           A(i+2,i)=1;
           A(1,N-1)=1;
           A(N-1,1)=1;
       end
```

```
  % end
end

 disp('MATRICE DI CONNESSIONE')
   D=diag(sum(A,2));
   Ac=A-D
   autovaloridiAc=eig(Ac)'
   V=diag(autovaloridiAc);
   lambda2=TrovaLambda2(V);
   Rc=abs((lambda2/Klim)*(C2*R/C1))
   Rth=round(Rc)
   Ci=-diag(Ac)'  %Ci=gradi di ogni nodo;C=grado medio della rete
   C=sum (Ci)/N
   disp('COEFFICIENTE DI CLUSTERING:')
   cc=C/(N-1)
 disp('GRADI IN ORDINE DECRESCENTE; NODI IN ORDINE DECRESCENTE DI GRADO')
   [Gradi,Nodi]=sort(Ci,'descend')
    grado_max=max(Ci);
    Z=diag(Ci);
    [x,x]=find(Z==grado_max);
   nodihub=x';
            if length(x)==N
                disp('Rete regolare;tutti i nodi hanno lo stesso grado')
                str=('tutti i nodi')
             else
                nodihub=x'
                str=num2str(nodihub)
            end
%plotting
t=0:2*pi/N:2*pi;
xy=[sin(t);cos(t)]';
gplot(Ac,xy,'-b.');
set(findobj('Type','Line'),'MarkerEdgeColor','r','MarkerSize',15,'LineWidth',0.0
8);
title(['\fontsize{16}rete iniziale con ',num2str(N),' nodi e ',num2str(N/2),....
'  triangoli',10,' \lambda\fontsize{8}2= \fontsize{16}',num2str(lambda2),....
'; R\fontsize{8}c\fontsize{16}=',num2str(Rth),'; coeff. clust.
=',num2str(cc),10,' nodi hub= ',str])
axis('equal','off')
hold off

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%




      end
end




clc
close all
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%tipo:
%1)'NN': rete iniziale i tipo nearest- neighbor con k connessioni per lato ed N
%nodi,alla quale si aggiunge una connessione alla volta, fino ad arrivare
%alla rete di tipo globally connected;tabella finale di confronto tra le varie
%reti
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%2)'hub': rete di tipo hub, inizialmente simmetrica con due nodi hub; le
%connessioni di un nodo vengono di volta in volta spostate all'altro nodo;
%tabella finale di confronto tra le varie reti;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%3)'star':rete con connessione a stella
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%4)'triangoli1':rete  di triangoli in serie di un circuito aperto;N deve essere
un
%numero intero multiplo di 3 e diverso da tre. Numero di nodi= N, se N è un nu-
mero dispari,
%numero di nodi= N+1 se N è un numero pari
%Il numero di triangoli che si ottiene è'il più piccolo intero  vicino a N/2
%se N è dispari, a N/2-1 se N è pari
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
%5)'triangoli2': rete aperta di triangoli separati da una connessione, il numero
di nodi è
%uguale a N-1, il numero di triangoli è uguale al più piccolo intero vicino
%a N/3-1. N deve essere maggiore di 3 ed un multiplo  di 3

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% 'triangoli3': crea una rete chiusa di triangoli .
%N deve essere un numero pari ed N>=3. Il numero di triangoli che si ottiene
%è'uguale a N/2, il numero di nodi è uguale a N.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%il tipo 'quadrilateri' crea una rete di quadrilateri .
%N deve essere un numero positivo multiplo di 4. Il numero di quadrilateri che
si ottiene
%è'uguale a N/4, il numero di nodi è uguale a N+2

[A,lambda2,Rth]=TutteLeConnessioni(8,2,10e-9,100e-9,1780,6.31,'star');
```

```
function [Ac] = reti_irregolari2(N,p,p_star,tipo,tipo2,z)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Questa funzione restituisce in uscita una rete random,o una rete small-word
% o una rete scale-free;vediamo anche come si modifica lambda2 in una rete
%di tale tipo con il distacco di un numero z  di nodi e delle relative
%connessioni,sia partendo in ordine decrescente dal nodo più alto in grado,
%sia in modo casuale,e come si modifica con il pinning di un numero z di
%nodi.
%%I parametri di ingresso sono:
%N=numero di nodi della rete 'random' e numero di nodi finale per la rete 'sca-
le-free'
%z= numero di nodi da rimuovere o da pinnare; è richiesto solo se si
%seleziona il tipo 'distacco_nodi' o il tipo 'pinning' dalla funzione
%'input'
%p=probabilità di connessione di due nodi i e j(rete 'random')
%p-star=probabilità di distacco dei nodi o di pinning per ogni nodo
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%tipo=
```

```
%1)'random'
%2)'scale_free'
%3)'NW',rete small-world di tipo Newman Watts;inserire il numero
%di connessioni per nodo k,che deve essere un numero pari,minore di
%N,maggiore di logN.Controllo di errore.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%tipo2=
%1)'distacco_nodi'
%2)'pinning'
%3)'0'(termine)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
switch tipo
    case 'random'
A=zeros(N,N);
%inizializzazione matrice di incidenza
for i=1:N
    for j=1:N
        c=rand;
        if c<p && i~=j
            A(i,j)=1;
            A(j,i)=A(i,j);
        end
    end
end

D=diag(sum(A,2));
disp('matrice di connessione')
Ac=A-D
B=Ac;
U=Ac;
As=Ac;
autovaloridiA=eig(Ac)';
 V=diag(autovaloridiA);
 lambda2=TrovaLambda2(V);
 disp('Ci=gradi di ogni nodo;C=grado medio della rete ')
 Ci=-diag(Ac)';
  C=sum(Ci)/size(Ac,2);
 grado_max=max(Ci)
[gradi,nodi]=sort(Ci,'descend')
Z=diag(Ci);
[x,x]=find(Z==grado_max);
nodi_hub=x'  %nodi hub
if length(x)==N
    disp('rete regolare, tutti i nodi hanno lo stesso grado')
    str=('rete regolare');
elseif length(x)==1
    str=(['nodo hub=',num2str(nodi_hub)]);
else
    str=(['nodi hub=',num2str(nodi_hub)]);
end
t=0:2*pi/size(Ac,2):2*pi;
xy=[sin(t);cos(t)]';
gplot(A,xy,'r.')
hold on
gplot(A,xy,'-b.')
set(findobj('Type','Line'),'MarkerEdgeColor','k','MarkerSize',20,'LineWidth',0.0
8);
title(['\fontsize{18}rete random con ',int2str(N),' nodi e probabilità
p=',num2str(p),10,....
    ' \lambda\fontsize{9}2\fontsize{18} = ',num2str(lambda2),',  ',str])
```

```
axis('square','off')
hold off
pause
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
    case 'scale_free'
        Mo=input('inserire il numero di nodi iniziali, maggiore di 0 e minore
del 30% di N  \nMo = ')

        while Mo<=0 || Mo>ceil(0.3*N)
            errordlg('è necessario che Mo sia maggiore di 0 e minore del 30% di
N','file error')
           Mo=input('inserire il numero di nodi iniziali, \nMo = ')
            %return
        end

        A=zeros(N,N);
for i=Mo+1:N
  for j=1:Mo
      p=0.75;
      c=rand;
        if i~=j && c<=p
           A(i,j)=1;
           A(j,i)=1;
           D=diag(sum(A,2));
            Ac=A-D; %matrice di connessione
            B=Ac;
            Ci=-diag(Ac)'; %gradi di ogni nodo
            [gradi,nodi]=sort(Ci,'descend');
       end
  end
end
for i=Mo+1:N
    for j=Mo+1:N
         p(j)=Ci(j)/(sum(Ci)-Ci(j));
         c=rand;
        %if  j~=1 && j~=2 && j~=3  && c<=p(j)
        if c<=p(j)&& i~=j
        Ac(i,j)=1;
        Ac(j,i)=1;
        end
    end
 end
D=diag(sum(Ac,2));
disp('MATRICE DI CONNESSIONE');
Ac=Ac-D
autovaloriAc=eig(Ac)';
V=diag(autovaloriAc);
lambda2=TrovaLambda2(V);
disp('GRADI DI OGNI NODO');
Ci=-diag(Ac)' %gradi di ogni nodo
disp('grado medio della rete')
C=sum(Ci)/N
disp('coefficiente di clustering')
cc=C/(N-1)  %coefficiente di clustering
disp('GRADI DELLA RETE IN ORDINE DECRESCENTE E NODI CORRISPETTIVI');
[gradi,nodi]=sort(Ci,'descend')
grado_max=max(Ci)
Z=diag(Ci);
[x,x]=find(Z==grado_max);
```

```
nodi_hub=x'  %nodi hub

if length(x)==N
    disp('rete regolare, tutti i nodi hanno lo stesso grado')
    str=('rete regolare');
elseif length(x)==1

    str=(['nodo hub=',num2str(nodi_hub)]);
else

    str=(['nodi hub=',num2str(nodi_hub)]);
end
B=Ac;
U=Ac;
As=Ac;
%plotting della rete
t=0:2*pi/N:2*pi;
xy=[sin(t);cos(t)]';
gplot(A,xy,'r.')
hold on
gplot(Ac,xy,'-m.')
set(findobj('Type','Line'),'MarkerEdgeColor','r','MarkerSize',25,'LineWidth',0.0
8);
title(['\fontsize{24}rete di tipo scale-free con ',int2str(Mo),' nodi iniziali e
',int2str(N),....
    ' nodi finali;',10, '
\lambda\fontsize{12}2\fontsize{24}=',num2str(lambda2),', CC=',num2str(cc),',
',str])

axis('square','off')
hold off
pause

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%

case 'NW'  %rete small world con il metodo di Newman Watts
    beep
    k=input('Inserisci k,il numero delle connessioni per nodo,e premi invio; k
deve essere un numero pari, maggiore di logN, minore di N,inserisci 0 per termi-
nare; \nk = ')

    while k>=N||rem(k,2)~=0||k<log(N)
        beep
        errordlg('errore: k deve essere un numero pari minore di N e maggiore di
log(N)','file error');
        k=input('Inserisci k,il numero delle connessioni per nodo,e premi invio;
k deve essere un numero pari, maggiore di logN, minore di N\nk = ')

    end


    A=zeros (N,N);
  for i=1:k/2
    diagonale=ones(1,N-i);
    lato=ones(1,i);
    A=A+diag(diagonale,-i)+diag(diagonale,i)+diag(lato,N-i)+diag(lato,-N+i);
%matrice di incidenza
  end
  D=diag(sum(A,2))
  Ac=A-D %matrice di connessione
```

```
  autovaloridiAc=eig(Ac)'
  V=autovaloridiAc;

  for i=1:length(V)
      if V(i)>=-0.05&&V(i)<=0.05
          V(i)=0;
      end
  end
   Nz=nonzeros(V);
   lambda2=max(Nz)

Ci=-diag(Ac)'
C=sum(Ci)/N
[gradi,nodi]=sort(Ci,'descend')
grado_max=max(Ci)

Z=diag(Ci);
[x,x]=find(Z==grado_max);
nodi_hub=x';
if length(x)==N
    disp('rete regolare, tutti i nodi hanno lo stesso grado')
    str=('rete regolare');
elseif length(x)==1

    str=(['nodo hub=',num2str(nodi_hub)]);
else

    str=(['nodi hub=',num2str(nodi_hub)]);
end

%plotting
t=0:2*pi/N:2*pi;
xy=[sin(t);cos(t)]';
gplot(Ac,xy,'-g.')
axis('equal','off')
set(findobj('Type','Line'),'MarkerEdgeColor','b','MarkerSize',20,'LineWidth',1);
title(['\fontsize{18}rete iniziale con ',num2str(N),' nodi e ',num2str(k),' con-
nessioni per nodo',10,'p=',num2str(p),',
\lambda\fontsize{9}2\fontsize{18}=',num2str(lambda2),'  ',str])
hold off
pause
for i=1:N
    for j=1:N
          c=rand;
          if i~=j && A(i,j)==0 && c<=p
              A(i,j)=1;
              A(j,i)=1;
          end
    end
end
D=diag(sum(A,2));
Ac=A-D
B=Ac;
U=Ac;
As=Ac;
autovaloridiAc=eig(Ac)'
  V=autovaloridiAc;

  for i=1:length(V)
      if V(i)>=-0.05&&V(i)<=0.05
          V(i)=0;
```

```
        end
    end
     Nz=nonzeros(V);
     lambda2=max(Nz)

Ci=-diag(Ac)'
C=sum(Ci)/N
[gradi,nodi]=sort(Ci,'descend')
grado_max=max(Ci)

Z=diag(Ci);
[x,x]=find(Z==grado_max);
nodi_hub=x';
if length(x)==N
    disp('rete regolare, tutti i nodi hanno lo stesso grado')
    str=('rete regolare');
elseif length(x)==1

    str=(['nodo hub=',num2str(nodi_hub)]);
else

    str=(['nodi hub=',num2str(nodi_hub)]);
end

%plotting
t=0:2*pi/N:2*pi;
xy=[sin(t);cos(t)]';
gplot(Ac,xy,'-g.')
axis('equal','off')
set(findobj('Type','Line'),'MarkerEdgeColor','b','MarkerSize',18,'LineWidth',1);
title(['\fontsize{18}rete small-world con ',num2str(N),'
nodi',10,'p=',num2str(p),',
\lambda\fontsize{9}2\fontsize{18}=',num2str(lambda2),'  ',str])
hold off
pause
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
switch tipo2
    case 'distacco_nodi'

        if z>N
            beep
            errordlg('z deve essere un numero intero minore di N','File Error')
            return
        end


 nodostaccato=0;
disp('distacco dei nodi a partire da quello avente il più alto grado')
for k=1:z
    for i=1:size(Ac,2)
        for j=1:size(Ac,2)
            if i==j && i==nodi(1) && nodostaccato<z
                       Ac(i,:)=0;
                       Ac(:,j)=0;
                       nodostaccato=nodostaccato+1
                       D=diag(sum(Ac,2));
                       As=Ac-D
                       autovaloridiA=eig(As)'
                        V=diag(autovaloridiA);
```

```
                        lambda2=TrovaLambda2(V);
                        Ci=-diag(As)' %gradi di ogni nodo
                        %disp('grado medio della rete');
                        C=sum(Ci)/N
                        disp('coefficiente di clustering')
                        cc=C/(N-1)  %coefficiente di clustering
        disp('GRADI DELLA RETE IN ORDINE DECRESCENTE E NODI CORRISPETTIVI');
                       [gradi,nodi]=sort(Ci,'descend')
                        grado_max=max(Ci)
                        Z=diag(Ci);
                        [x,x]=find(Z==grado_max);
                        nodi_hub=x'  %nodi hub
                   if length(x)==N
                       disp('rete regolare, tutti i nodi hanno lo stesso grado')
                       str=('rete regolare');
                   elseif length(x)==1
                      str=(['nodo hub=',num2str(nodi_hub)]);
                   else
                       str=(['nodi hub=',num2str(nodi_hub)]);
                   end
                        gplot(B,xy,'r.')
                        hold on
                        if grado_max==0
                            title('\fontsize{20}matrice nulla')
                           break
                    else
                        gplot(As,xy,'-g.')

set(findobj('Type','Line'),'MarkerEdgeColor','r','MarkerSize',25,'LineWidth',0.0
8);
title(['\fontsize{24}distacco di # ',num2str(nodostaccato),' nodi in ordine de-
crescente di grado;',10,....
    '\lambda\fontsize{12}2\fontsize{24}= ',num2str(lambda2),',
CC=',num2str(cc),'; ',str])
                        hold off
                        axis('square','off')
                        pause
                    end
                end
            end
        end
end
nodostaccato=0;
disp('DISTACCO CASUALE DI NODI')
for k=1:z
    for i=1:size(B,2)
        for j=1:size(B,2)
            c=rand;
            if i==j && c <= p_star && nodostaccato<z
                        B(i,:)=0;
                        B(:,j)=0;
                        nodostaccato=nodostaccato+1
                        D=diag(sum(B,2));
                        Ar=B-D
                        autovaloridiA=eig(Ar)';
                        V=diag(autovaloridiA);
                        lambda2=TrovaLambda2(V);
                        Ci=-diag(Ar)' %gradi di ogni nodo
                        %disp('grado medio della rete');
                        C=sum(Ci)/N
                        disp('coefficiente di clustering')
                        cc=C/(N-1)  %coefficiente di clustering
        disp('GRADI DELLA RETE IN ORDINE DECRESCENTE E NODI CORRISPETTIVI');
```

```
                    [gradi,nodi]=sort(Ci,'descend')
                    grado_max=max(Ci)
                    Z=diag(Ci);
                    [x,x]=find(Z==grado_max);
                     nodi_hub=x'   %nodi hub
                if length(x)==N
                    disp('rete regolare, tutti i nodi hanno lo stesso grado')
                    str=('rete regolare');
                elseif length(x)==1
                   str=(['nodo hub=',num2str(nodi_hub)]);
                else
                    str=(['nodi hub=',num2str(nodi_hub)]);
                end


                     gplot(U,xy,'r.')
                     hold on
                     gplot(Ar,xy,'-b.')

set(findobj('Type','Line'),'MarkerEdgeColor','r','MarkerSize',25,'LineWidth',0.0
8);
                    axis('square','off')

title(['\fontsize{24}distacco di # ',num2str(nodostaccato),' nodi in ordine ca-
suale;',10,....
 '\lambda\fontsize{12}2\fontsize{24}= ',num2str(lambda2),', CC=',num2str(cc),',
',str])
                    hold off
                    pause

            end
        end
    end
end
    case 0
        return
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    case 'pinning'
       if z>N
            beep
            errordlg('z deve essere un numero intero minore di N','File Error')
            return
        end
 disp('NODI PINNATI AVENTI I GRADI PIU ALTI')
X=zeros(1,N);
Y=zeros(1,N+1)';
A1=[X;As];
A2=[Y A1];
D=diag(sum(A2,2));
A=A2-D;
B=A;
T=A;
Ci=-diag(A)';  %Ci sono i gradi di ogni nodo;C è il grado medio della rete
Ci2=Ci;
disp('gradi in ordine decrescente e relativi nodi');
[gradi,nodi]=sort(Ci,'descend');
numerodinodipinnati=0;
for k=1:z
    for i=1:N+1
            if i==nodi(k)&& numerodinodipinnati<z
                     A(i,1)=1;
                     numerodinodipinnati=numerodinodipinnati+1
```

```
                        D=diag(sum(A,2));
                        Ap=A-D
                        autovaloridiA=eig(Ap);
                        V=diag(autovaloridiA);
                        lambda2=TrovaLambda2(V);
                        %Ci sono i gradi di ogni nodo;C è il grado medio della
rete
                        Ci=-diag(Ap)';
                        C=sum(Ci)/size(Ap,2);
                        disp('gradi in ordine decrescente');
                        S=sort(Ci,'descend')
                        grado_max=max(Ci)
                        Z=diag(Ci);
                        [x,x]=find(Z==grado_max);
                        nodi_hub=x'  %nodi hub
                    if length(x)==N
                       disp('rete regolare, tutti i nodi hanno lo stesso grado')
                       str=('rete regolare');
                    elseif length(x)==1
                      str=(['nodo hub=',num2str(nodi_hub)]);
                    else
                       str=(['nodi hub=',num2str(nodi_hub)]);
                    end
                        grado_max=max(Ci);
                        t=0:2*pi/(N+1):2*pi;
                        xy=[sin(t);cos(t)]';
                        gplot(Ap,xy,'-r.')

set(findobj('Type','Line'),'MarkerEdgeColor','y','MarkerSize',15,'LineWidth',0.0
8);

 title(['\fontsize{14}pinning di # ',num2str(numerodinodipinnati),....
 ' nodi in ordine decrescente di grado;',10,'\lambda\fontsize{7}2\fontsize{14}=
',....
 num2str(lambda2),'; ',str])

                        axis('square','off')
                        hold off
                        pause
            end
    end
end
disp('NODI PINNATI SCELTI CASUALMENTE')
numerodinodipinnati=0;

 for i=2:N
     for j=1:N
             c=rand;
            if i==j && c <=p_star && numerodinodipinnati<z && gradi(i)~=0
                        T(i,1)=1;
                        numerodinodipinnati=numerodinodipinnati+1
                        D=diag(sum(T,2));
                        Ap=T-D
                        autovaloridiA=eig(Ap);
                        V=diag(autovaloridiA);
                        lambda2=TrovaLambda2(V);
                %Ci sono i gradi di ogni nodo;C è il grado medio della rete
                       Ci=-diag(Ap)';
                       C=sum(Ci)/size(Ac,2);
                       disp('gradi in ordine decrescente e relativi nodi:')
                       [gradi,nodi]=sort(Ci,'descend')
                       grado_max=max(Ci)
```

```
                              Z=diag(Ci);
                              [x,x]=find(Z==grado_max);
                               nodi_hub=x'  %nodi hub
                           if length(x)==N
                              disp('rete regolare, tutti i nodi hanno lo stesso grado')
                              str=('rete regolare');
                              elseif length(x)==1
                                     str=(['nodo hub=',num2str(nodi_hub)]);
                           else
                                 str=(['nodi hub=',num2str(nodi_hub)]);
                          end
                               t=0:2*pi/(N+1):2*pi;
                               xy=[sin(t);cos(t)]';
                               gplot(Ap,xy,'-m.')

set(findobj('Type','Line'),'MarkerEdgeColor','k','MarkerSize',15,'LineWidth',0.0
8);
title(['\fontsize{14}rete  con numero ',num2str(numerodinodipinnati),' di nodi
pinnati in ordine casuale ',10,....
    '\lambda\fontsize{7}2\fontsize{14}= ',num2str(lambda2),';  ',str])
                               axis('square','off')
                               hold off
                               pause
            end
     end
 end
end
end
```

```
 clc
close all
%reti_irregolari2(N,p,p_star,tipo,tipo2,z)
%I parametri di ingresso sono:
%N=numero di nodi della rete 'random' e numero di nodi finale per la rete 'sca-
le-free'
%z= numero di nodi da rimuovere o da pinnare è richiesto solo se si
%seleziona il tipo 'distacco_nodi' o il tipo 'pinning' dalla funzione
%altrimenti si può mettere 0
%p=probabilità di connessione di due nodi i e j(rete 'random')
%p-star=probabilità di distacco dei nodi o di pinning per ogni nodo
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%tipo=
%1)'random'
%2)'scale_free'
%3)'NW',rete small-world di tipo Newman Watts;inserire il numero
%di connessioni per nodo k,che deve essere un numero pari,minore di
%N,maggiore di logN.Controllo di errore.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%tipo2=
%1)'distacco_nodi'
%2)'pinning'
%3)'0'(termine)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
help  reti_irregolari2
reti_irregolari2(14,0.5,0.3,'scale_free','pinning',3);
```

```
function[Ac,lambda2,Rc] = MetodoCartaPenna1(V,C1,C2,R,Klim,tipo,tipo2,z,p_star)

%Questa funzione restituisce una rete di qualsiasi tipo e per qualsiasi
%numero N di nodi,se è data la matrice di connessione V,oppure il vettore V,dove
sono
%indicate tutte le connessioni dei nodi,in modo unilaterale.Nel parametro y,
%presente nella funzione,le connessioni vengono prese in senso inverso; c'è
%un avviso di errore se le connessioni sono ripetute nello sesso senso o in
%senso inverso.
%PARAMETRI di uscita:
%Ac=matrice di connessione
%%lambda2=il più grande autovalore delle matrici di connessione,escluso lo
%zero
%Rc=resistenza di soglia
%PARAMETRI DI INGRESSO:
%V= vettore che riporta tutte le connessioni dei nodi da uno a N,in modo unila-
terale, SI UTILIZZA
%SOLO PER IL TIPO 'vettore';un segnale di errore avverte se una connessione
%è ripetuta, in senso diretto o in senso inverso.
%V=matrice di incidenza o di connessione di una rete, SI UTILIZZA SOLO PER
%IL TIPO 'matrice'; un segnale di errore avverte se la matrice inserita non
%è quadrata.
%C1,C2 sono le capacità del circuito di Chua
%R è la resistenza del cicuito
%Klim è il parametro dal quale ricaviamo la resistenza di collegamento Rc
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%tipo: 1)'vettore' , si parte dal vettore x; 2)'matrice' si parte dalla
%matrice A;
%tipo2: 1)'pinning'esegue il pinning di z nodi, in ordine decrescente a
%partire da quello di grado più alto; 2)'0': termina la funzione
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%z  è il numero di nodi da pinnare o da distaccare;se il tipo2 è uguale a
%'0'non ha rilevanza,può essere un numero qualsiasi;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%p_star è la probabilità di distacco dei nodi; se il tipo2 è uguale a '0'
%non ha rilevanza,può essere un numero qualsiasi.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%OUTPUT
%Ci=gradi di ogni nodo;
%C=grado medio della rete;
%Gradi=gradi in ordine decrescente
%Nodi=rispettivi nodi
%grado_max=grado massimo
%nodi_hub=nodi aventi il grado massimo
%
switch tipo

case 'vettore'
    x=V;
   N=max(x);
 t=0:2*pi/N:2*pi;
 xy=[sin(t);cos(t)]';
 n=length(x);
 if rem(n,2)~=0
     errordlg(['errore:la lunghezza del vettore deve essere un numero pari
','File Error'])
 end
```

```
 for i=1:length(x)
  for k=2:2:length(x)-i
    if rem(i,2)~=0 && x(i)==x(i+1)
      errordlg(['nodo ', num2str(i), ' uguale a se stesso  ','File Error'])
      return
      else if rem(i,2)~=0 && ((x(i)== x(i+k)&& x(i+1)== x(i+1+k))|| (x(i)==
x(i+1+k)&& x(i+1)==x(i+k)))
      errordlg(['connessione tra i nodi ', num2str(x(i)),' e ',num2str(x(i+1)) '
ripetuta in senso diretto o inverso ','File Error'])
      return
          end
    end

   end

end
 y=zeros(1,length(x));
for i=1:n
        if rem(i,2)~=0
          y(i)=x(i+1);
          y(i+1)=x(i);
        end
end

y;
s=ones(n,1);
M=sparse(x,y,s,N,N);
%matrice di incidenza
B=full(M);
D=diag(sum(B,2));
Ac=B-D %matrice di connessione
Ac2=Ac;
B=Ac;
U=Ac;
autovaloridiAc=eig(Ac)'
V=diag(autovaloridiAc);
lambda2=TrovaLambda2(V);
Rc=abs((lambda2/Klim)*(C2*R/C1))
Ci=-diag(Ac)'%Ci sono i gradi di ogni nodo;C è il grado medio della rete
Ci2=Ci;
C=sum (Ci)/N
Ci2=Ci;
grado_max=max(Ci)
disp('gradi in ordine decrescente e relativi nodi')
[gradi,nodi]=sort(Ci,'descend')
 Z=diag(Ci);
[x,x]=find(Z==grado_max);
nodi_hub=x';%nodi hub
if length(x')==N
  disp('Rete regolare;tutti i nodi hanno lo stesso grado')
        %str=('tutti i nodi')
        str=('rete regolare')
            else
               disp('nodi_hub=x')
               str=(['nodihub = ', num2str(nodi_hub)]);
 end
Rc=abs((lambda2/Klim)*(C2*R/C1));
Rc=round(Rc)
gplot(Ac,xy,'-b.')
```

```
set(findobj('Type','Line'),'MarkerEdgeColor','r','MarkerSize',15,'LineWidth',0.0
8);
title(['\fontsize{16}rete con ',num2str(N),' nodi; \lambda\fontsize{8}2=
\fontsize{16}',num2str(lambda2),10,....
    'R\fontsize{8}c\fontsize{16}=',num2str(Rc),('; '),str ])              %';
nodi hub=',num2str(str)])
axis('equal','off')
hold off
 pause
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%
    case 'matrice'
        A=V;
        [m,n] = size(A)
        if m~=n
            errordlg('errore: matrice non quadrata','File Error')
            return
        end
        N=length(A)
D=diag(sum(A));
Ac=A-D
Ac2=Ac;
B=Ac;
U=Ac;
autovaloridiAc=eig(Ac)'
V=autovaloridiAc;
for i=1:length(A)
    if V(i)>=-0.05 && V(i)<=0.05
        V(i)=0;
    end
end
lambda2=max(nonzeros(V))
Rc=abs((lambda2/Klim)*(C2*R/C1));
Rc=round(Rc)
disp('gradi di ogni nodo')
Ci=-diag(Ac)'
Ci2=Ci;
disp('grado medio della rete')
C=sum(Ci)/length(A)
[gradi,nodi]=sort(Ci,'descend')
Z=diag(Ci);
[x,x]=find(Z==max(Ci));
nodi_hub=x'
if length(x)==N
    disp('rete regolare, tutti i nodi hanno lo stesso grado')
    str=('rete regolare');
elseif length(x)==1

    str=(['nodo hub=',num2str(nodi_hub)]);
else

    str=(['nodi hub=',num2str(nodi_hub)]);
end

t=0:2*pi/N:2*pi;
xy=[sin(t);cos(t)]';
gplot(Ac,xy,'-r.')
%st=sprintf('CONFIGURAZIONE %1.0f \t %10.2f\t
%10.0f\n',CONFIGURAZIONE(i),lambda2(i),Rth(i));
set(findobj('Type','Line'),'MarkerEdgeColor','y','MarkerSize',15,'LineWidth',0.0
8)
```

```
title(['\fontsize{16}Rete con ',num2str(N),' connessioni;
\lambda\fontsize{8}2\fontsize{16}=',....
    num2str(lambda2),10,'R\fontsize{8}c\fontsize{16}=', num2str(Rc),';  nodi
hub=',str])
axis('square','off')
hold off
pause
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%for i=1:q+1

 switch tipo2
case 'pinning'

    if z>N
        errordlg('Errore: z deve essere minore o uguale ad N','File Error')
        return
    end


X=zeros(1,N);
Y=zeros(1,N+1)';
A1=[X;Ac];
A2=[Y A1];
D=diag(sum(A2,2));
A=A2-D;
B=A;
T=A;
Ci=-diag(A)'  %Ci sono i gradi di ogni nodo;C è il grado medio della rete
Ci2=Ci;
disp('gradi in ordine decrescente e relativi nodi')
[gradi,nodi]=sort(Ci,'descend')
numerodinodipinnati=0;
disp('Pinning dei nodi a partire da quelli aventi il grado più alto, o dai nodi
in ordine crescente se si tratta di una rete regolare')



for k=1:z
     for i=1:N+1
            if i==nodi(k)&& numerodinodipinnati<z && gradi(k)~=0
                       A(i,1)=1;
                       numerodinodipinnati=numerodinodipinnati+1
                       D=diag(sum(A,2));
                       Ap=A-D
                       autovaloridiA=eig(Ap);
                       V=diag(autovaloridiA);
                       lambda2=TrovaLambda2(V);
                       Rc=abs((lambda2/Klim)*(C2*R/C1));
                       Rc=round(Rc)
                       %Ci sono i gradi di ogni nodo;C è il grado medio della
rete
                       Ci=-diag(Ap)';
                       C=sum(Ci)/size(Ac,2);
                       disp('gradi in ordine decrescente e relativi nodi');
                       [gradi,nodi]=sort(Ci,'descend')
                       grado_max=max(Ci)
                       Z=diag(Ci);
                       [x,x]=find(Z==grado_max);
                       nodo_hub=x'  %nodi hub
```

```
                        if length(x)==N
                            disp('rete regolare, tutti i nodi hanno lo stesso
grado')
                            str=('rete regolare');
                       elseif length(x)==1
                             str=(['nodo hub=',num2str(nodi_hub)]);
                        else
                             str=(['nodi hub=',num2str(nodi_hub)]);
                        end

                        %plotting
                        t=0:2*pi/(N+1):2*pi;
                        xy=[sin(t);cos(t)]';
                        gplot(Ap,xy,'-r.')

set(findobj('Type','Line'),'MarkerEdgeColor','y','MarkerSize',15,'LineWidth',0.0
8);
                        title(['\fontsize{16}rete  con numero
',num2str(numerodinodipinnati),' di nodi pinnati',10,....
                              '\lambda\fontsize{8}2\fontsize{16}=
',num2str(lambda2),' R\fontsize{8}c\fontsize{16}= ',num2str(Rc)])
                        axis('square','off')
                        hold off
                        pause
            end
     end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    case 'distacco_nodi'

        while z>N
            errordlg('z deve essere un numero intero minore di N','File Error')
            z=input('inserire il numero di nodi da staccare e premere invio\nz=
')
        end

           while p_star<0 || p_star>1
            errordlg('la probabilità deve essere compresa tra 0 ed 1:','File Er-
ror')
             p_star=input('inserire la probabilità di distacco dei nodi\np_star=
')
        end

 nodostaccato=0;
disp('distacco dei nodi a partire da quello avente il più alto grado')
for k=1:z
    for i=1:size(Ac,2)
        for j=1:size(Ac,2)
            if i==j && i==nodi(1) && nodostaccato<z
                        Ac(i,:)=0;
                        Ac(:,j)=0;
                        nodostaccato=nodostaccato+1
                        D=diag(sum(Ac,2));
                        As=Ac-D
                        autovaloridiA=eig(As)'
                         V=diag(autovaloridiA);
                         lambda2=TrovaLambda2(V);
                         Ci=-diag(As)' %gradi di ogni nodo
                         %disp('grado medio della rete');
                         C=sum(Ci)/N
```

```
                        disp('coefficiente di clustering')
                        cc=C/(N-1)  %coefficiente di clustering
        disp('GRADI DELLA RETE IN ORDINE DECRESCENTE E NODI CORRISPETTIVI');
                      [gradi,nodi]=sort(Ci,'descend')
                       grado_max=max(Ci)
                       Z=diag(Ci);
                       [x,x]=find(Z==grado_max);
                       nodi_hub=x'  %nodi hub
                  if length(x)==N
                      disp('rete regolare, tutti i nodi hanno lo stesso grado')
                      str=('rete regolare');
                  elseif length(x)==1
                     str=(['nodo hub=',num2str(nodi_hub)]);
                  else
                      str=(['nodi hub=',num2str(nodi_hub)]);
                  end
                       gplot(B,xy,'r.')
                       hold on
                       if grado_max==0
                           title('\fontsize{20}matrice nulla')
                          break
                   else
                       gplot(As,xy,'-g.')

set(findobj('Type','Line'),'MarkerEdgeColor','r','MarkerSize',15,'LineWidth',0.0
8);
title(['\fontsize{14}distacco di # ',num2str(nodostaccato),' nodi in ordine de-
crescente di grado;',10,....
    '\lambda\fontsize{7}2\fontsize{14}= ',num2str(lambda2),',
CC=',num2str(cc),'; ',str])
                        hold off
                        axis('square','off')
                        pause
                   end
             end
          end
      end
end
nodostaccato=0;
disp('DISTACCO CASUALE DI NODI')
for k=1:z
    for i=1:size(B,2)
        for j=1:size(B,2)
            c=rand;
            if i==j && c <= p_star && nodostaccato<z
                       B(i,:)=0;
                       B(:,j)=0;
                       nodostaccato=nodostaccato+1
                       D=diag(sum(B,2));
                       Ar=B-D
                       autovaloridiA=eig(Ar)';
                       V=diag(autovaloridiA);
                       lambda2=TrovaLambda2(V);
                       Ci=-diag(Ar)' %gradi di ogni nodo
                       %disp('grado medio della rete');
                       C=sum(Ci)/N
                       disp('coefficiente di clustering')
                       cc=C/(N-1)  %coefficiente di clustering
        disp('GRADI DELLA RETE IN ORDINE DECRESCENTE E NODI CORRISPETTIVI');
                      [gradi,nodi]=sort(Ci,'descend')
                      grado_max=max(Ci)
                      Z=diag(Ci);
                      [x,x]=find(Z==grado_max);
```

```
                    nodi_hub=x'  %nodi hub
                if length(x)==N
                    disp('rete regolare, tutti i nodi hanno lo stesso grado')
                    str=('rete regolare');
                elseif length(x)==1
                   str=(['nodo hub=',num2str(nodi_hub)]);
                else
                    str=(['nodi hub=',num2str(nodi_hub)]);
                end


                    gplot(U,xy,'r.')
                    hold on
                    gplot(Ar,xy,'-b.')

set(findobj('Type','Line'),'MarkerEdgeColor','r','MarkerSize',15,'LineWidth',0.0
8);
                    axis('square','off')

title(['\fontsize{14}distacco di # ',num2str(nodostaccato),' nodi in ordine ca-
suale;',10,....
 '\lambda\fontsize{7}2\fontsize{14}= ',num2str(lambda2),', CC=',num2str(cc),',
',str])
                    hold off
                    pause

            end
        end
    end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    case 0
        return
end

end



%function[Ac,lambda2,Rc] = MetodoCartaPenna1(V,C1,C2,R,Klim,tipo,tipo2,z)
%Questa funzione restituisce una rete di qualsiasi tipo e per qualsiasi
%numero N di nodi,se nel parametro x, oppure nella matrice A, sono
%indicate tutte le connessioni dei nodi,in modo unilaterale.Nel parametro y,
%presente nella funzione,le connessioni vengono prese in senso inverso; c'è
%un avviso di errore se le connessioni sono ripetute nello sesso senso o in
%senso inverso.

%PARAMETRI DI INGRESSO:
%V=x=vettore che riporta tutte le connessioni dei nodi da uno a N,in modo
%unilaterale, SI UTILIZZA SOLO PER IL TIPO 'vettore';un segnale di errore
%avverte se una connessione è ripetuta, in senso diretto o in senso inverso.
%V=A=matrice di incidenza o di connessione di una rete, SI UTILIZZA SOLO PER
%IL TIPO 'matrice'; un segnale di errore avverte se la matrice inserita non
%è quadrata.
%C1,C2 sono le capacità del circuito di Chua
%R è la resistenza del cicuito
%Klim è il parametro dal quale ricaviamo la resistenza di collegamento Rc
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%tipo: 1)'vettore' , si parte dal vettore x; 2)'matrice' si parte dalla
%matrice A;
%tipo2: 1)'pinning'esegue il pinning di z nodi, in ordine decrescente a
%partire da quello di grado più alto; 2)'0': termina la funzione
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%z è il numero di nodi da pinnare o da distaccare;se il tipo2 è uguale a '0'
%non ha rilevanza,può essere un numero qualsiasi.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%p_star è la probabilità di distacco dei nodi;se il tipo2 è uguale a '0'
%non ha rilevanza,può essere un numero qualsiasi.
%%%%%%%%%%%%%%%%%%%%


clc
close all
help  MetodoCartaPenna1
V=[1 2 2 3 3 4 4 5 5 6 6 7 7 8 8 1 1 3 3 5 5 7 7 1 2 4 4 6 6 8 8 2 1 5]

    %V=[       -6     1     1     1     0     1     1     1
    %          1    -5     1     1     0     0     1     1
    %          1     1    -4     1     1     0     0     0
    %          1     1     1    -6     1     1     1     0
    %          0     0     1     1    -4     1     1     0
    %          1     0     0     1     1    -5     1     1
    %          1     1     0     1     1     1    -6     1
    %          1     1     0     0     0     1     1    -4
    %                                                      ]


MetodoCartaPenna1(V,10e-9,100e-9,1780,6.31,'vettore','0',3,0.3);
```

Bibliografia

S. Boccaletti, V. Latora, Y. Moreno, e M. Chavez , D.-U. Hwanga , *Complex networks: Structure and dynamics,* Physics Reports 424 (2006) 175 – 308

M.P. Kennedy, *Robust op-amp realization for Chua's circuits*. Frequents vol. 46, N°3-4 March-April 1992, pag. 66-80

Posadas-Castillo C. , Cruz –Hernandez C. and Lopez-Guitierrez R.M. *Sincronization in a network of Chua 's circuits*, proceedings of the Fourth IASTED international conference circuits, signals and systems, November 20-22-2006 , San Francisco, CA, USA

Xiao Fan Wang, Huang, Ying-Cheng Lai, Louis M. Pecora, *Generic behavior of master-stability functions in coupled non linear dynamical system*, International Journal of Bifurcation and Chaos, Physical review E 80,036204(2009), pag.1 -11

Xiao Fan Wang, *Complex networks: topology, dynamics and synchronization*, international journal of bifurcation and Chaos, Vol. 12, N°5 (2002), pag.889

Xiao Fan Wang, Guanrong Chen, Pinning control of scale-free dynamical networks, Physica A 310 (2002) pag. 521-531.

Leon Chua, Makoto Itoh, Ljupco Kocarev e Kevin Eckert, *Chaos Synchronization in Chua's circuit*, Journal of circuits, Systems and Computers, Vol. 3 N° 1(1993)Pag. 93 -108

C. Posadas-Castillo, C. Cruz-Hernandez, R.M. Lopez-Gutierrez, *Experimental realization of synchronization in complex networks with Chua's circuits like nodes,* In ScienzeDirect ,settembre 2007, San Francisco, CA, USA.

M.E.J.Newman, *The Structure and Function of Complex Networks*, Siam Review, vol 45 N°2, pag. 167 - 256

Matsumoto, T., *A chaotic attractor from Chua's circuit* , IEEE Trans. Circuit & Systems, 1984

Pecora L., Carrol, T., *Synchronization in chaotic system*, Phys. Rev. Lett ,1990

Pecora L., Carrol, T., *Master Stability Function for Synchronized Coupled Systems*, Phys. Rev Letters, vol. 80, n.10, 1988

Leon O.Chua, Chai Wah Wu, Hanshan Huang and Guo-Qun Zhong, *Generating Chaos-Part I: Routes to Chaos*, IEEE Trans. Circuit & Systems, Fundamental Theory and Application, vol.40, n.10, oct.1993

Leon O.Chua, Chai Wah Wu, Hanshan Huang and Guo-Qun Zhong, *Generating Chaos-Part II: Strange Attractors*, IEEE Trans. Circuit & Systems, Fundamental Theory and Application, vol.40, n.10, oct.1993

Michael Peter Kennedy, *Three steps to chaos-Part II: A Chua's Circuit Primer*, IEEE Trans. Circuit & Systems, Fundamental Theory and Application, vol.40, n.10, oct.1993

Michael Peter Kennedy, *Three steps to chaos-Part II: Evolution*, IEEE Trans. Circuit & Systems, Fundamental Theory and Application, vol.40, n.10, oct.1993

Massimiliano de Magistris, *Comportamento asintotico dei circuiti dinamici*, Modulo di Teoria dei Circuiti,Dipartimento di ingegneria elettrica, Università di Napoli Federico II, novembre 2009

Massimiliano de Magistris, *Circuiti con dinamiche complesse, biforcazioni e caos deterministico*, Modulo di Teoria dei Circuiti,Dipartimento di ingegneria elettrica, Università di Napoli Federico II, novembre 2008

Massimiliano de Magistris, *Una implementazione del circuito di Chua con operazionali: simulazioni Spice ed un esperimento*, Modulo di Teoriadei Circuiti,Dipartimento di ingegneria elettrica, Università di Napoli Federico II, novembre 2009

Maurizio Porfiri, Mario di Bernardo *Criteria for global pinning-controllability of complex networks*, Automatica 44 (2008) , 3100-3106

Francesco Sorrentino, Mario di Bernardo, Franco Garofalo and Guarong Chen, *Controllability of complex networks via pinning*, Physical review E 75, 046103 (2007)

L.A.B. Tõrres and L.A. Aguirre, *Inductorless Chua's circuit,* Electonics Letters, 9th November, vol. 36, N°23.

tesi di laurea:

Dr.Paola della Corte, *Sincronizzazione di circuiti caotici di Chua*: *analisi di robustezza*

Dr. Maiorano Claudio, *Analisi sperimentale della sincronizzazione di circuiti di Chua: effetti di accoppiamenti dinamici e topologie con elevati numeri di nodi*

Dr. Cesarino Claudia, *Analisi sperimentale della sincronizzazione dei circuiti di Chua*: *effetti topologici di link direzionali e non direzionali*