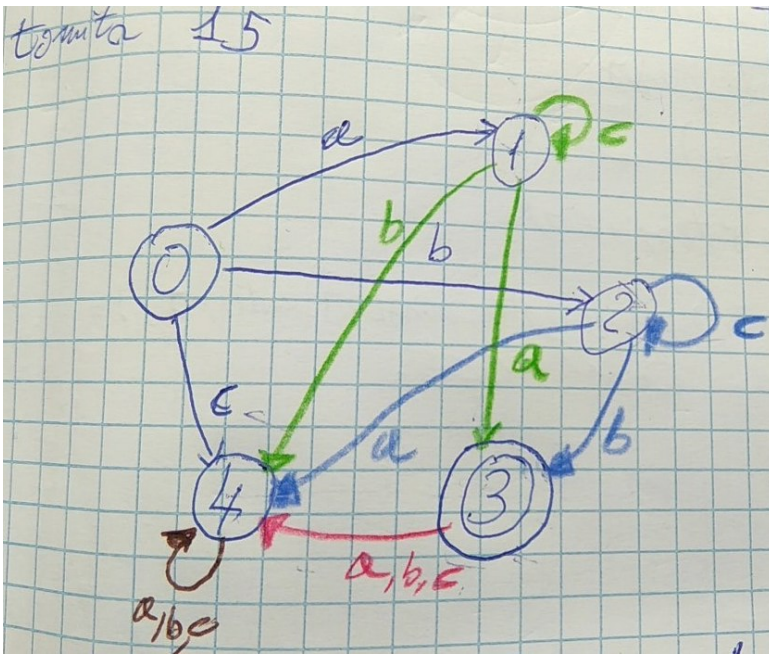


Esempio faccenda stati accettanti/rigettanti



Innanzitutto confrontando il tomita15 con sé stesso ritorna 1, bene.

Fissando il Tomita15 originale come reference dfa, (che ha solo lo stato 3 come accettante), faccio variare lo/gli stati accettanti del Tomita15 e l'automa che ottengo lo uso come subject. Riporto accanto il valore di similarità strutturale.

*****		*****
Stati accettanti		Similarità
Tomita15 modified		strutturale
0		0.000144239
1		0.290008
2		0.290008
3 (same as tomita15)		1
4		0.257799
none		0.378851
0,1		0.000101101
0,2		0.000101101
0,3		0.000405275
0,4		3.56063e-05
1,2		0.145027
1,3		0.340992
1,4		0.17786
2,3		0.340992
2,4		0.17786
3,4		0.270101
0,1,2		4.13084e-05
0,1,3		0.000163523
0,1,4		1.15549e-05

1,2,3	0.150086
1,2,4	0.100003
2,3,4	0.170869
0,1,2,3	6.16027e-05
0,1,2,4	1.85185e-06
1,2,3,4	0.100004
0,1,3,4	6.20058e-06
0,2,3,4	6.20058e-06
0,1,2,3,4	0

Che le coppie (1,4);(2,4) riportino lo stesso punteggio di similarità è un buon segnale perché i nodi 1 e 2 sono strutturalmente identici (stessi nodi sorgente che portano ad essi, stessi nodi terminazione a cui arrivano i loro archi, stessa combinazione di archi: uno che torna in sé stesso, uno che va nello stato 3, uno che va nello stato 4).

Quello che succede con questa implementazione è che viene fatto un controllo nella funzione Iterate() e se nel grosso ciclo su tutte le coppie dei nodi dei grafi trova una coppia con "colore"/label diversa, quello che fa è semplicemente andare avanti nel for, infatti:

```
for(int i=0; i<graph_a_n; i++)
    for(int j=0; j<graph_b_n; j++)
    {
        if(graph_a->Label(i)!=graph_b->Label(j))
            continue;
        ...
        ...
    }
```

Non so se è quello che vogliamo, potremmo ripensare la faccenda prevedendo un certo coefficiente di penalità per il confronto tra acc/rej anziché il taglio netto.

Proviamo a ragionare con il nostro scenario:

mettiamo che Alice per andare a lavoro si dirige verso casa di Bob, e Bob per andare al lavoro si dirige verso casa di Alice e si mettono d'accordo per fare esattamente lo stesso percorso in maniera speculare, tra andata e ritorno.

Possiamo dire che Alice e Bob fanno la stessa strada?

O il differenziarsi dei tragitti tra andata e ritorno deve farci considerare diversi i percorsi? E se è così, poichè sono esattamente speculari, allora si somigliano al 50%?

Altri esempi di calcolo di punteggio di similarità SENZA CONSIDERARE IL COLORE

Dal paper:

The termination condition is $\max_{ij} |x_{ij}^k - x_{ij}^{k-1}| < \varepsilon$ for some chosen precision ε .

Confronto 0

Confronto il Tomita15 con un suo duplicato in cui dallo Stato 4, con la lettera b, andiamo allo stato 2.

Quello che vediamo è che incrementando il valore del fattore di terminazione epsilon, il valore di similarità diminuisce di pochissimo, segnalando di fatto la convergenza dell'algoritmo:

Epsilon	Similarità
0.01	0.443864
0.001	0.418553
0.0001	0.415896
...	...
0.000000001	0.415584
...	...

Aver cambiato una sola transizione ha più che dimezzato il punteggio di similarità. Ce lo aspettavamo? Forse sì visto che abbiamo modificato un nodo fortemente caratteristico, che è lo stato pozzo.

Confronto 1

Confronto il Tomita15 con un suo duplicato in cui dallo Stato 3, con la lettera a, andiamo allo stato 0.

Il punteggio di similarità va diventando significativamente sempre più piccolo al diminuire del fattore di terminazione epsilon:

Epsilon	Similarità
0.1	0.617105
0.01	0.0679213
0.001	0.0067411
0.0001	0.000669046
...	...
0.000000001	7.15622e-09
...	...

Ne deduciamo che la similarità tende a zero.

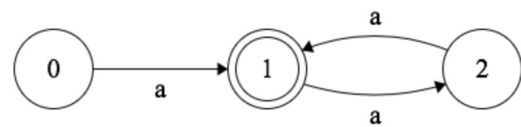
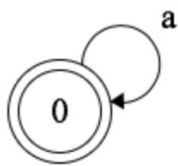
Cambiare una sola transizione da un automa può farlo diventare completamente diverso dalla sua versione originale?

Innanzitutto questo è un automa con pochi stati e poche transizioni.

Facendo arrivare una transizione all'interno dello stato 0, cambiamo l'insieme dei suoi vicini entranti, che nel Tomita15 era nullo, e adesso ha lo stato 3 come elemento. Lo stato 0 ha tre collegamenti con tre su quattro degli altri stati dell'automata, per cui quando si confronterà la similarità entrante dei suddetti stati, occorrerà cercare il matching ottimo per la similarità del loro vicinato entrante, e cioè proprio dello stato zero. E lo stato zero dei due automi confrontati tra

loro avrà sempre un punteggio di similarità entrante pari a zero, perché nel tomata originale non ci sono transizioni entranti nello stato 0, nella versione modificata ce n'è una.
 Poi l'unico stato non raggiunto dallo stato 0, è lo stato 3, al quale abbiamo variato l'insieme del vicinato uscente.
 Mi pare che sia per queste ragioni che il punteggio di similarità vada convergendo a zero.

Confronto 2



Confrontando questi due automi, il punteggio di similarità va diventando significativamente sempre più piccolo al diminuire del fattore di terminazione epsilon:

Epsilon	Similarità
0.1	0.316406
0.01	0.0316764
0.001	0.00317121
0.0001	0.000317479

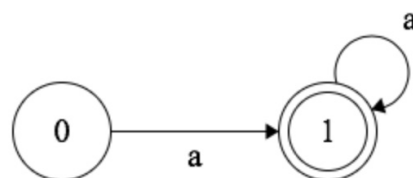
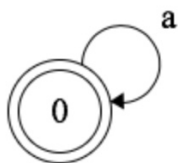
...

...

Ne deduciamo che la similarità tende a zero.

Sembra essere un risultato ragionevole dato che nessuno stato del secondo automa ha una transizione che torna in sé stesso. Eppure mi aspettavo un punteggio di similarità che tendesse più velocemente a zero al variare di epsilon.

Confronto 3



Epsilon	Similarità
0.1	0.237305
0.01	0.0237573
...	...
0.000000001	2.39457e-09

Consideriamo $\epsilon = 0.1$ può starci che la similarità sia vicina al 25%, dato che il matching è tra lo stato 0 e lo stato 1 dei due automi, dove però lo stato 1 ha un vicino entrante che non sia sé stesso, dimezzando per questa ragione la similarità entrante. Quello che mi ha sorpreso inizialmente è che a parità di ϵ , il confronto 2 scritto più sopra ha un punteggio di similarità più alto. Ma mi sono fatto ingannare dalla vista di una transizione che fa andare nello stesso stato, presente nel caso di questo ultimo confronto numero tre.

Guardiamo le matrici di similarità per i due confronti, con $\epsilon = 0.1$

Matrice similarità confronto 2
(0.158203 0.237305 0.316406)

Matrice similarità confronto 3
(0.158203 0.237305)

Lo stato 1 nei due automi subject dei confronti, è strutturalmente identico a quello dell'altro: hanno due transizioni entranti e una uscente (nel primo la transizione va in un altro stato, nel secondo la transizione torna nello stesso stato). Infatti il punteggio di similarità inter-nodo è identico.

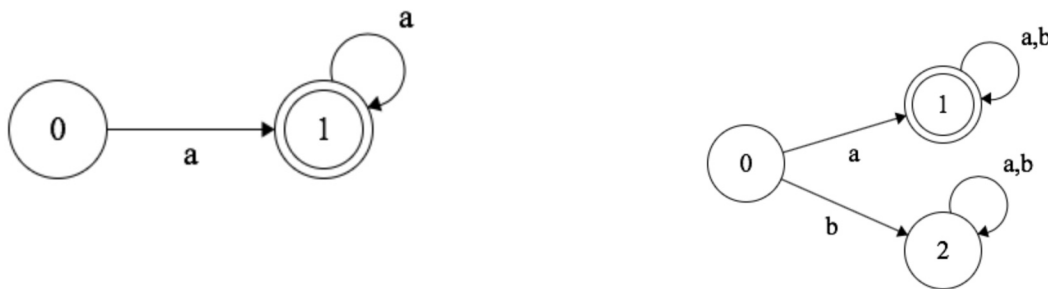
Lo stato 0 dell'automa reference è però più simile allo stato 2 del confronto 2, il quale ha una sola transizione entrante anziché due.

La differenza nel numero di stati non è penalizzante in questa implementazione perché ricordiamo che il valore di similarità globale è così ricavato:

$$s(G_A, G_B) = \frac{1}{n} \sum_{l=1}^n x_{f(l)g(l)}$$

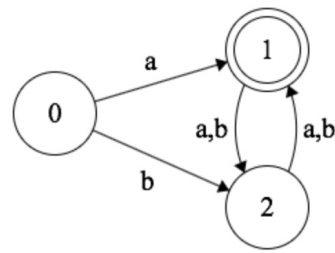
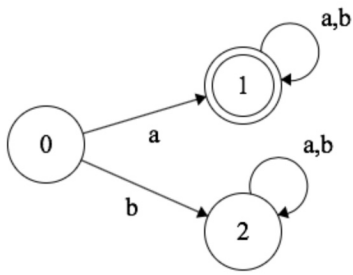
dove $n = \min(|V_A|, |V_B|)$.

Confronto 4



Epsilon	Similarità
0.1	0.84375
0.01	0.692688
0.001	0.66928
0.0001	0.666929

Confronto 5



Epsilon

0.0001

||| Similarità

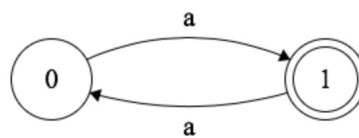
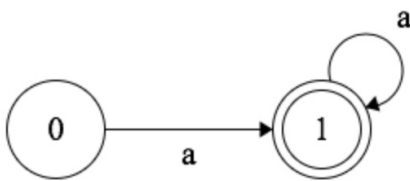
1

Esempio che mi sembra significativo di linguaggio diverso ma struttura identica.

Anche se mi chiedo: una transizione che fa tornare lo stato in sé stesso, è una transizione entrante ordinaria? Non dobbiamo considerarla in maniera diversa in questi casi? Forse no.

Sembra però mancare il concetto speculare al “nodo sorgente” ossia nodo senza alcuna transizione in ingresso ma solo in uscita. Un nodo pozzo non deve essere considerato come nodo terminazione? Senza quindi transizioni in uscita (anche se tecnicamente ha delle transizioni che portano su sé stesso)?

Confronto 6



Epsilon

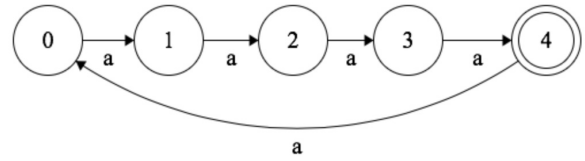
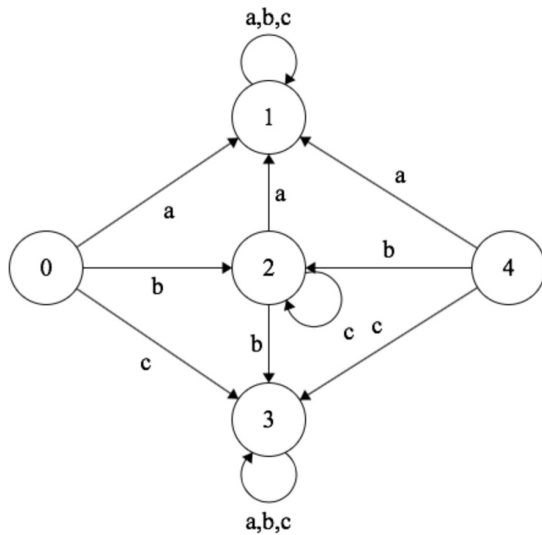
0.0001

||| Similarità

1

Come confronto 5.

Confronto 7



Epsilon

||| Similarità

0.1

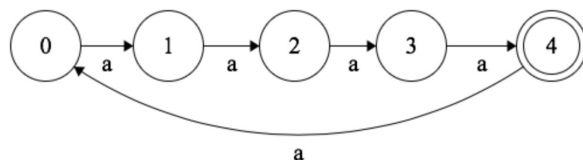
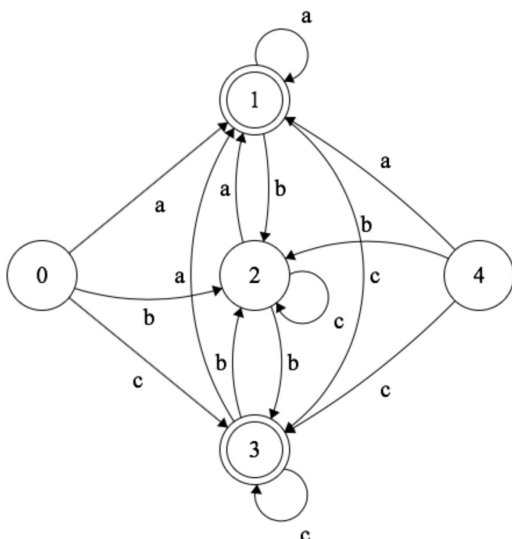
0.0876194

0.01

0.00835603

Già per valori alti di epsilon il punteggio di similarità è molto basso: questo accade perché nel reference dfa abbiamo due stati “sorgente” senza alcuna transizione entrante, mentre il subject dfa ha per tutti gli stati almeno una transizione entrante.

Confronto 8



Epsilon

||| Similarità

0.1

0.0145185

0.01

0.000813899

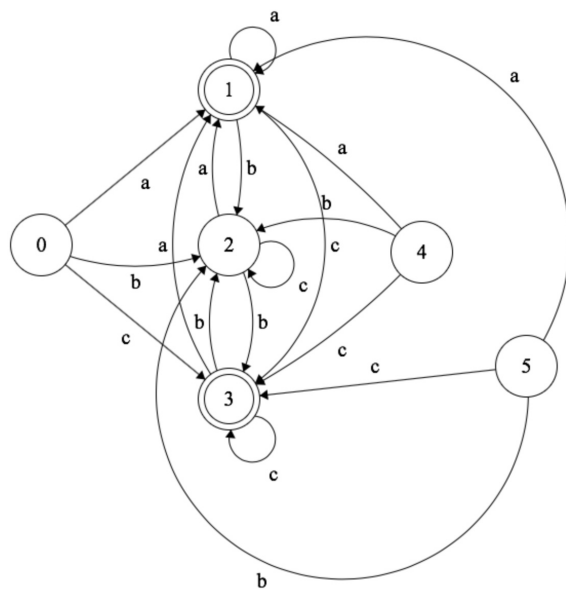
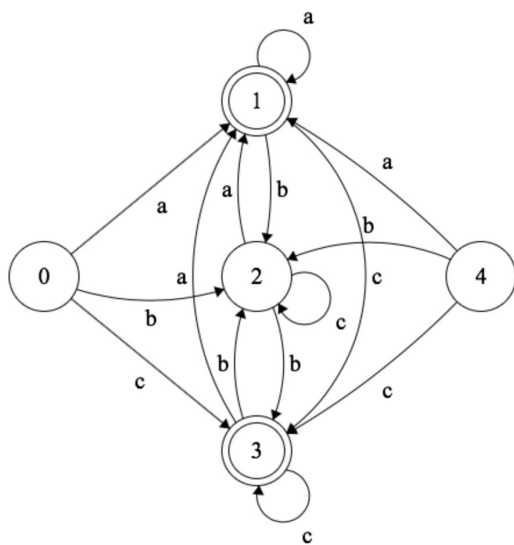
Caso più estremo del confronto 7: per epsilon pari a 0.1 gli automi si somigliano strutturalmente dell' 1%.

Non solo come prima non c'è nessuno stato nel secondo automa che non abbia alcuna transizione in ingresso. In più ogni stato del primo automa ha 3 transizioni distinte in uscita, nel secondo automa ciascuno stato ha una sola transizione.

Inoltre qui il minimo tra il numero degli stati dei due automi, il coefficiente n che smorza il punteggio di similarità globale, è pari al numero degli stati di entrambi gli automi.

Quindi secondo questa misura di similarità, due automi sono totalmente diversi quando uno possiede un nodo sorgente e l'altro no, e il numero di transizioni entranti e uscenti è uniformemente distante confrontando tutti gli stati di un automa con tutti gli stati dell'altro.

Confronto 9



Epsilon

0.0001

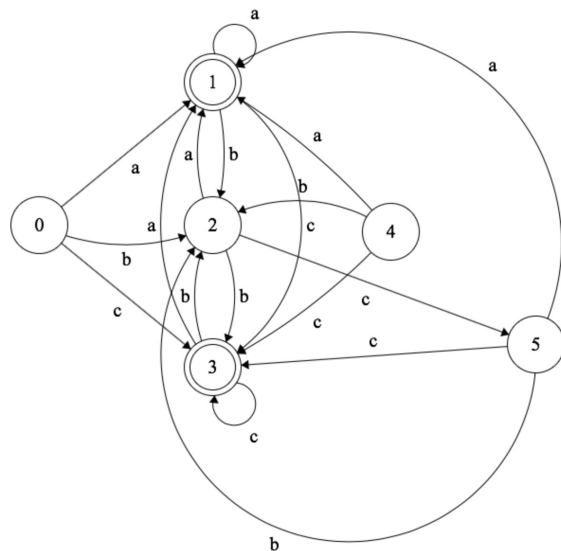
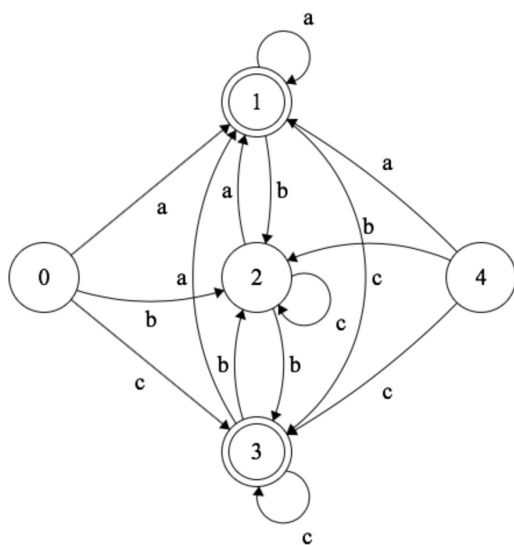
0.00001

||| Similarità

0.600409

0.600041

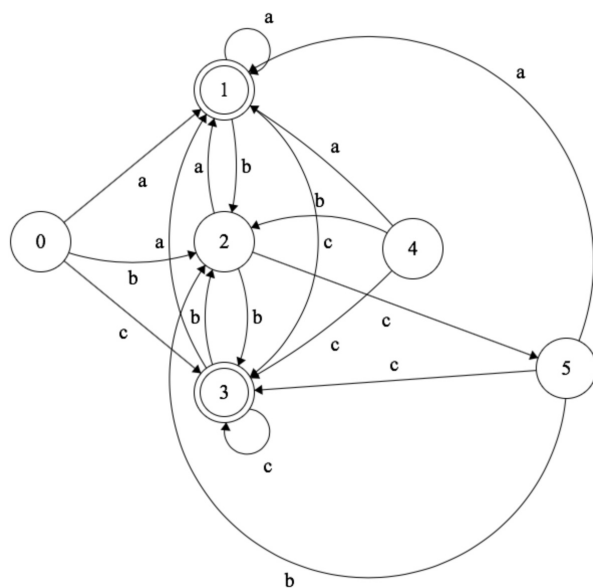
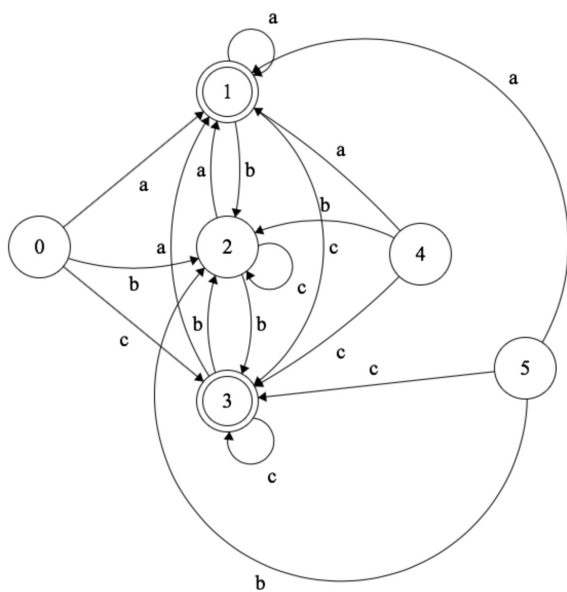
Confronto 10



Epsilon
0.0000000001

Similarità
0.595699

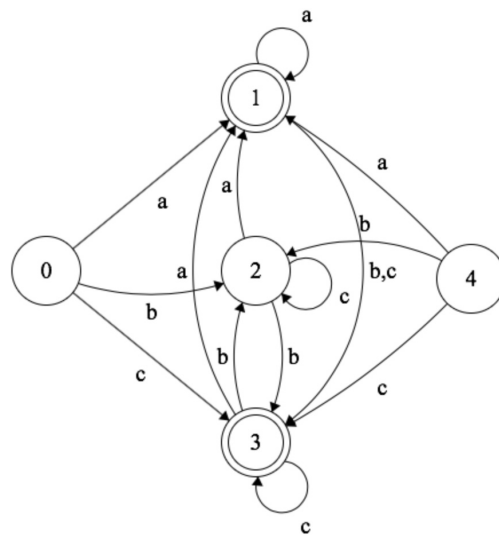
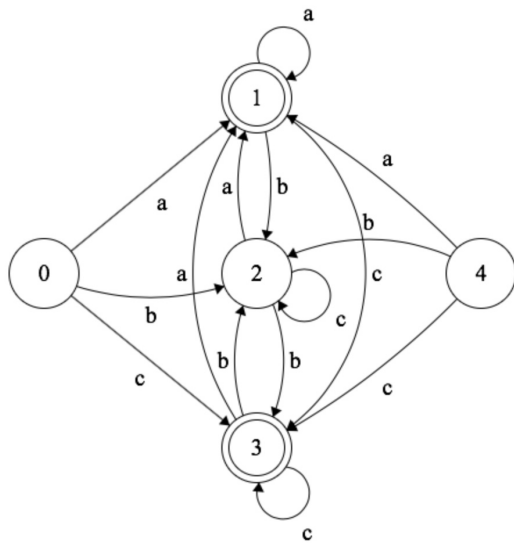
Confronto 11



Epsilon
0.1
0.01
0.001
0.0001

Similarità
0.811825
0.584368
0.545025
0.540999

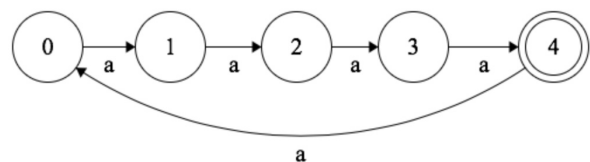
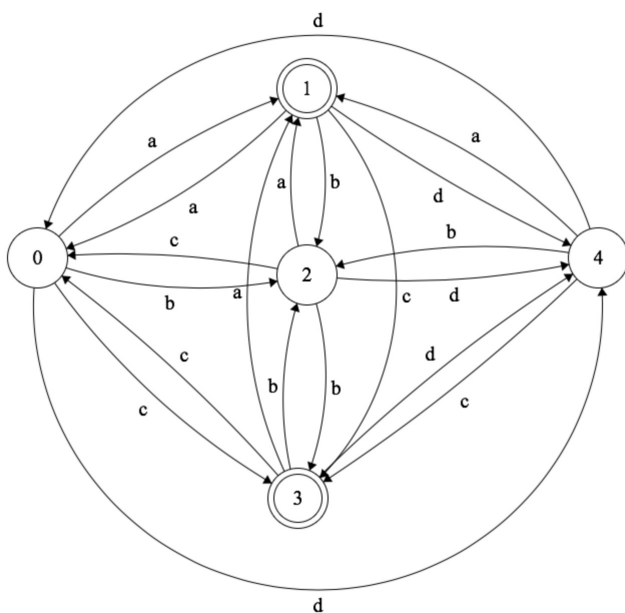
Confronto 12



Lo stato 1 non è più connesso allo stato 2 tramite b, nella versione modificata di destra la transizione da 1 con b va in 3.

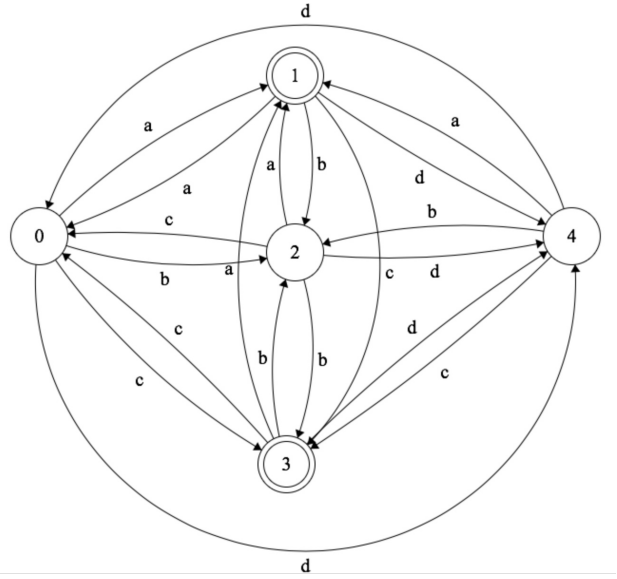
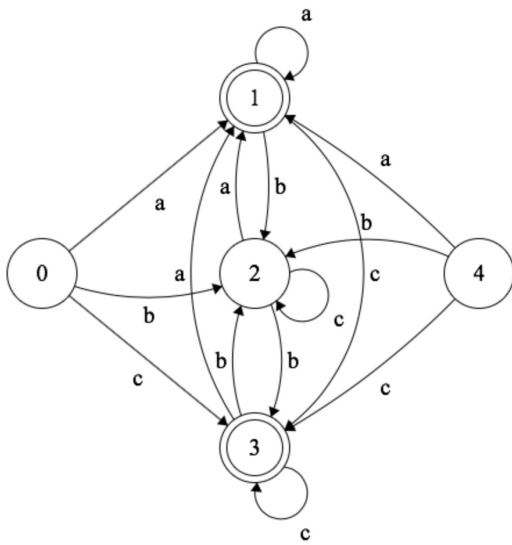
Epsilon	Similarità
0.1	0.892889
0.01	0.66627
0.001	0.63378
0.0001	0.630688

Confronto 13



Epsilon	Similarità
0.1	0.015625
0.01	0.000976562

Confronto 14



Epsilon

||| Similarità

0.1

0.17349

0.01

0.0185517

0.001

0.00198377

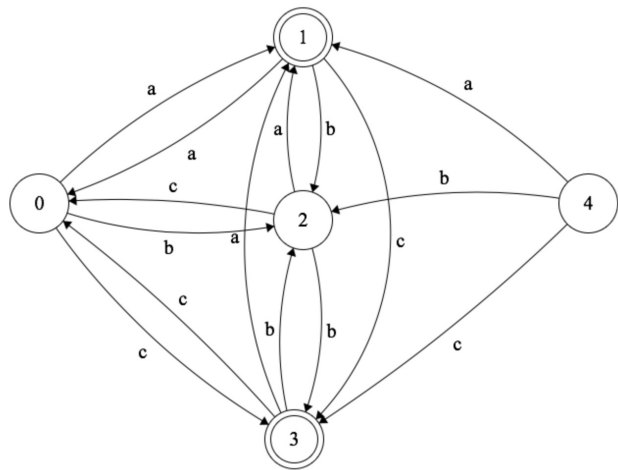
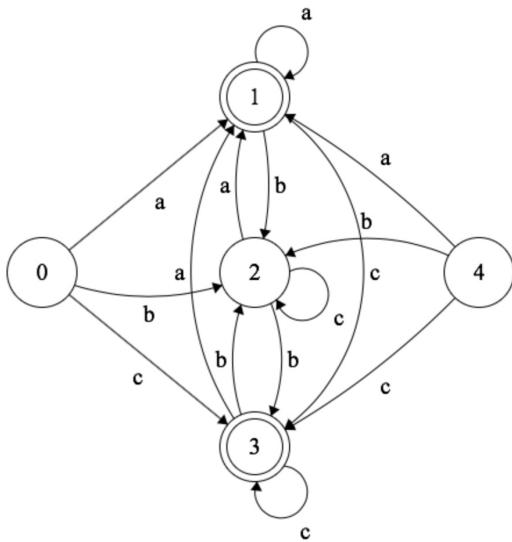
0.0001

0.000212128

0.00001

1.64819e-05

Confronto 15



Epsilon

||| Similarità

0.1

0.684733

0.01

0.340158

0.001

0.30083

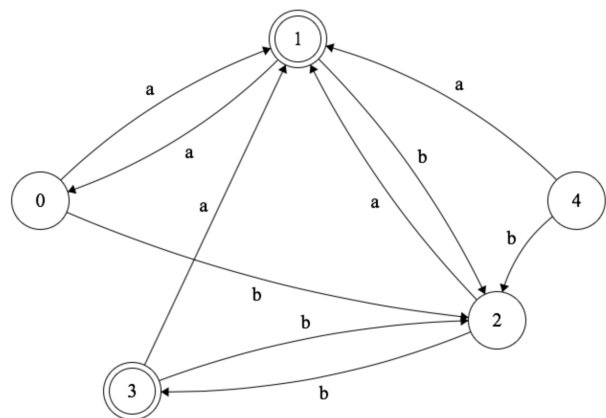
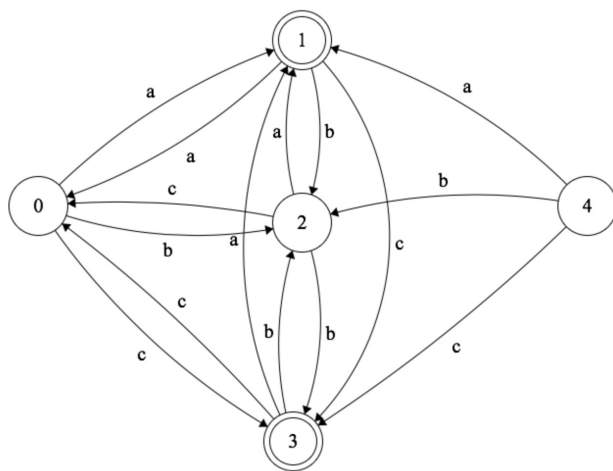
0.0001

0.29711

0.00001

0.296765

Confronto 16

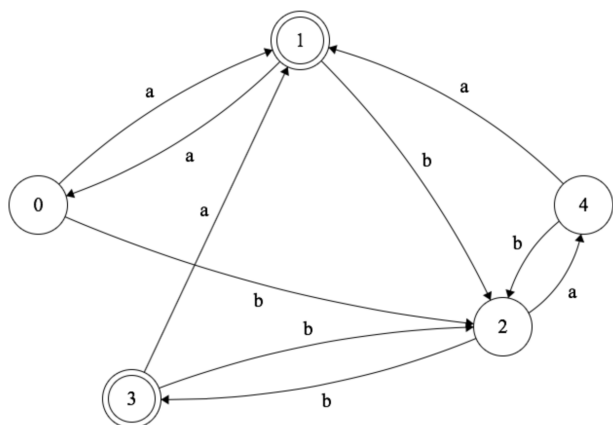
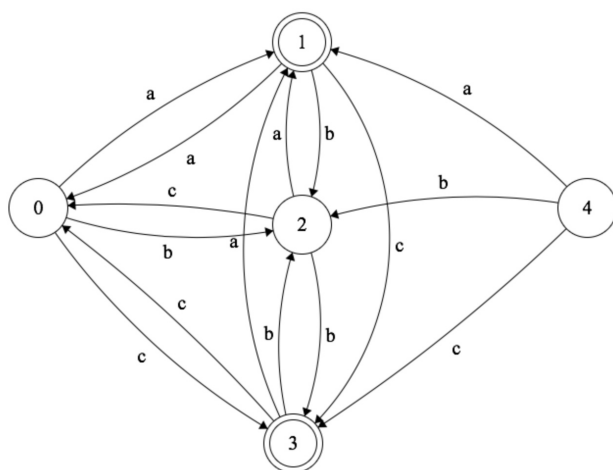


Epsilon

||| Similarità

0.1	0.285001
0.01	0.201314
0.001	0.189817
0.0001	0.188898
...	...
0.000000001	0.188764

Confronto 17



Il secondo automa è simile al secondo automa del confronto 16 ma, nello stato 2, la transizione con la lettera a anziché finire nello stato 1 va nello stato 4. Come si vede dai risultati, è bastato togliere dal secondo automa il nodo "sorgente" senza cioè transizioni entranti, per far convergere la similarità verso 0.

Epsilon

0.1

0.01

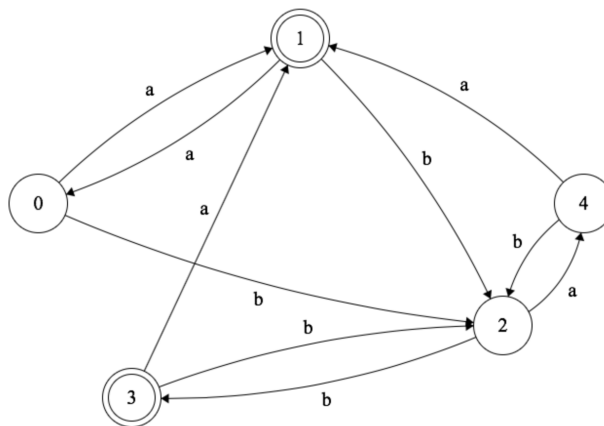
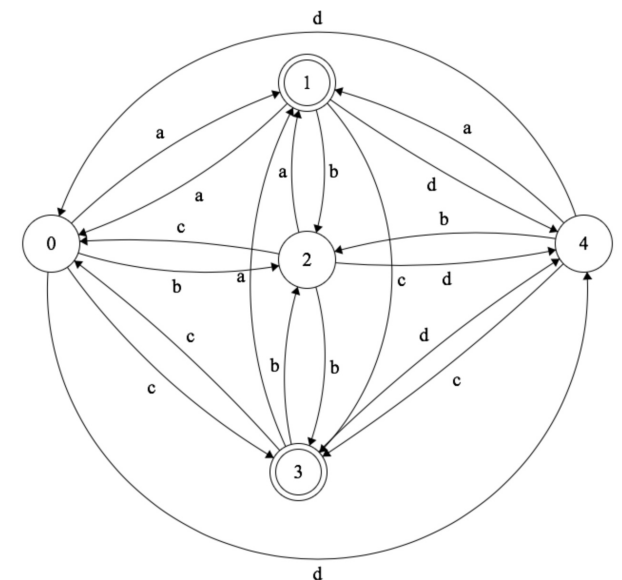
0.001

||| Similarità

0.0869032

0.00848263

0.000837798

Confronto 18**Epsilon**

0.1

0.01

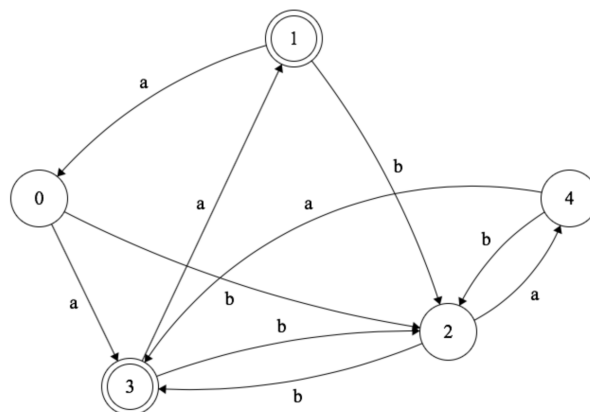
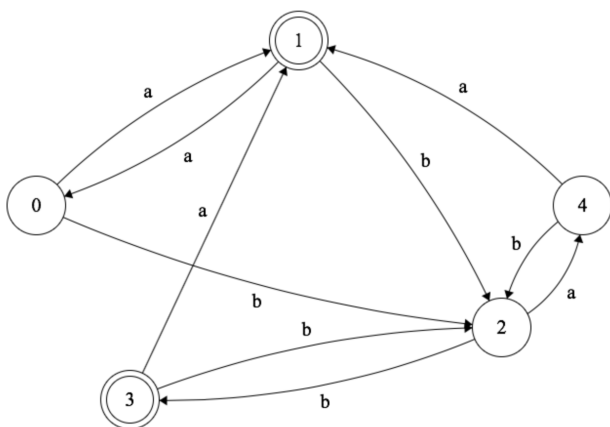
0.001

||| Similarità

0.0394043

0.00586572

0.000857949

Confronto 19

Il secondo automa è una copia del primo con due transizioni cambiate:

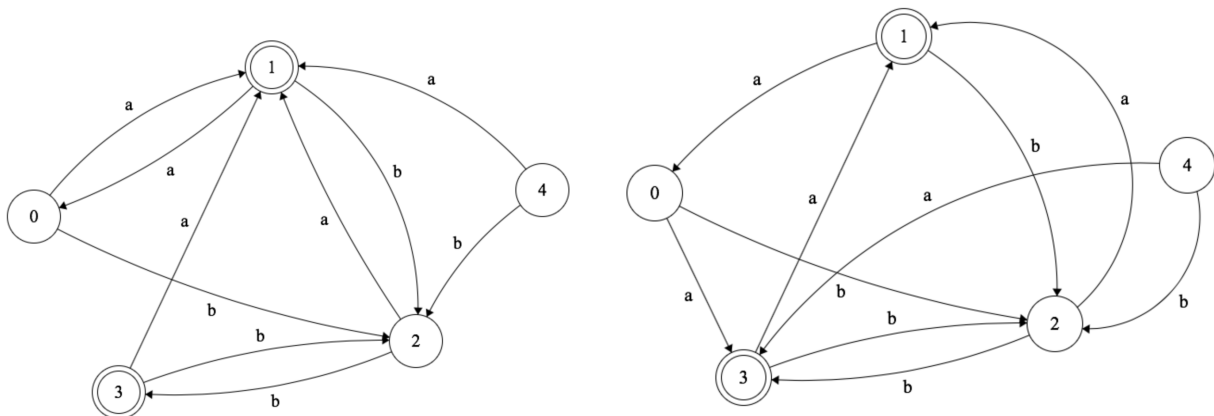
1) nello stato 0 la transizione con a adesso va nello stato 3

2) nello stato 4 la transizione con a adesso va nello stato 3

Quello che conta è che in nessuno dei due automi c'è un nodo sorgente, e il punteggio di similarità converge a 0, nonostante tra i due automi siano cambiate due sole transizioni.

Epsilon	Similarità
0.1	0.763744
0.01	0.0902915
0.001	0.00893256
...	...
0.0000001	9.40549e-07

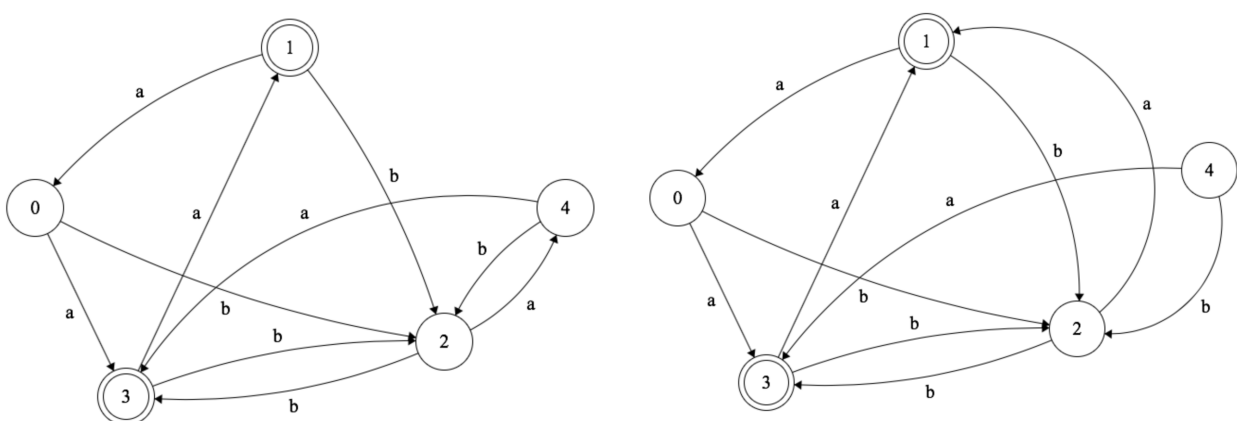
Confronto 20



Gli automi sono uguali a quelli del confronto 19, tranne che gli stati 4 adesso sono nodi senza transizioni entranti. Il punteggio di similarità non converge a zero.

Epsilon	Similarità
0.1	0.747526
0.01	0.334862
0.001	0.29568
0.0001	0.291955

Confronto 21

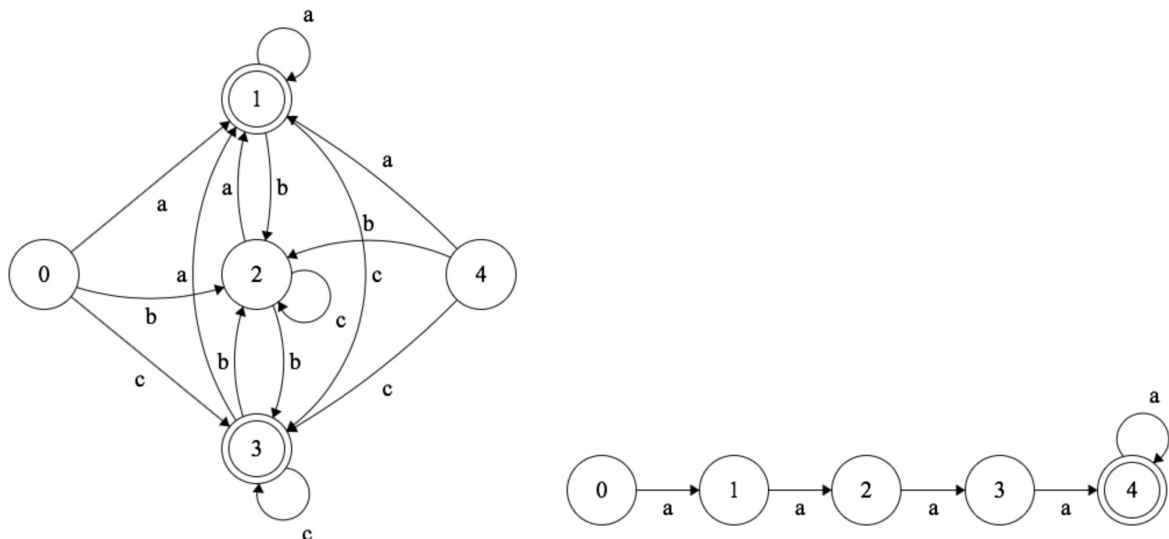


Gli automi sono identici, ma nello stato 4 adesso non c'è più la transizione che arrivava dallo stato 2, e quindi nel secondo automa non c'è un nodo sorgente. La similarità converge a zero.

Epsilon	Similarità
---------	------------

0.1	0.60026
0.01	0.0580037
0.001	0.00632808

Confronto 22



Sono due automi che non si somigliano nemmeno per sbaglio. Eppure la presenza di un nodo sorgente in entrambi fa sì che il punteggio di similarità non converga a zero al decrescere del fattore di terminazione epsilon.

Epsilon	Similarità
0.1	0.131296
0.01	0.11502
0.001	0.113298
0.0001	0.113242
...	...
0.00000001	0.113235

Conclusioni

Il punteggio di similarità non tende a zero al diminuire di epsilon se c'è almeno una coppia di nodi senza transizioni entranti, come vediamo nel confronto 22.

Nel confronto 21 vediamo come confrontare due automi tra loro identici tranne che per l'assenza in uno dei due di un nodo sorgente (assenza realizzata cambiando solo una transizione) faccia convergere a zero la similarità.

Guardando il confronto 19 vediamo come confrontando due automi in cui cambiano solo 2 transizioni su 15, ma in cui non ci sono nodi sorgente, il punteggio di similarità converge comunque a zero.

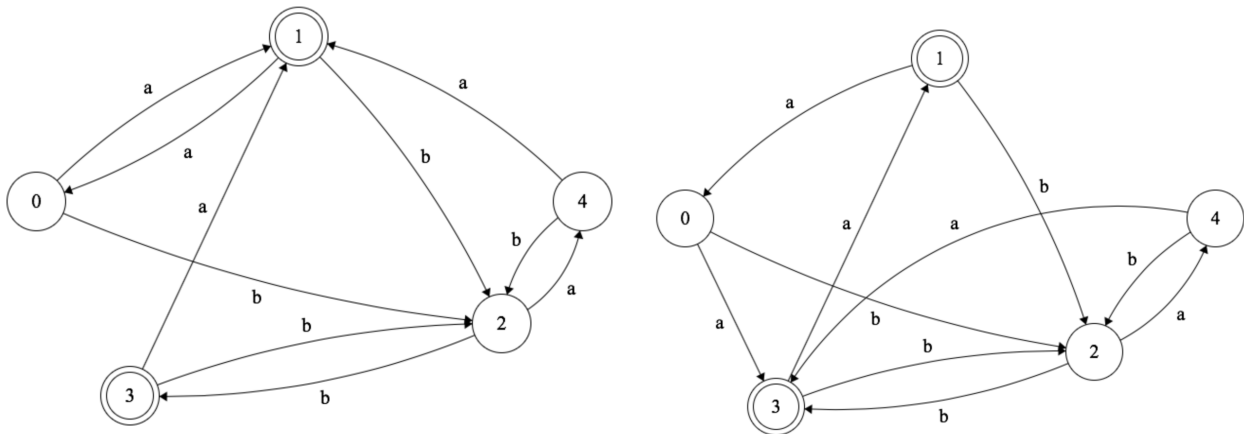
Quindi nel caso in cui si confrontano tra loro automi in cui almeno uno dei due non possiede uno stato sorgente, si deve usare una soglia grande come $\varepsilon = 0.1$ altrimenti il punteggio va rapidamente verso zero.

Come mai accade questo?

La condizione di terminazione è $\max_{ij} |x_{ij}^k - x_{ij}^{k-1}| < \varepsilon$, dove x_{ij}^k indica la similarità tra il nodo i del dfa X e il nodo j del dfa Y alla iterazione k .

Guardiamo le matrici di similarità inter nodo per i confronti 19 e 20:

Confronto 19



Epsilon	Similarità strutturale automi
0.1	0.763744
0.01	0.0902915
0.001	0.00893256
...	...
0.0000001	9.40549e-07

$\varepsilon = 0.1$

0.688151	0.762153	0.408529	0.474175	0.805990
0.484809	0.492911	0.702474	0.779333	0.420790
0.348687	0.396322	0.780490	0.606228	0.339084
0.647135	0.682726	0.388672	0.488715	0.849609
0.647135	0.682726	0.388672	0.488715	0.849609

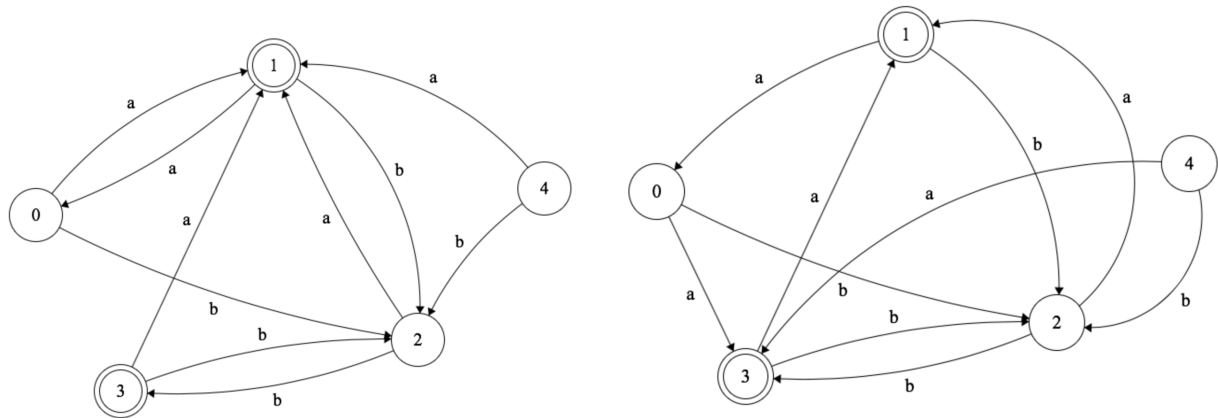
$\varepsilon = 0.01$

0.082118	0.091196	0.048469	0.05606	0.095146
0.05726	0.057986	0.081814	0.091227	0.049473
0.040781	0.045605	0.092631	0.070380	0.039494
0.075355	0.079415	0.045523	0.058028	0.101049
0.075355	0.079415	0.045523	0.058028	0.101049

$\varepsilon = 0.001$

0.008128	0.009024	0.004797	0.005548	0.009413
0.005664	0.005735	0.008093	0.009023	0.004894
0.004033	0.004511	0.009162	0.00696	0.003906
0.007455	0.007858	0.004505	0.005743	0.009998
0.007455	0.007858	0.004505	0.005743	0.009998

Confronto 20



Epsilon	Similarità
0.1	0.747526
0.01	0.334862
0.001	0.29568
0.0001	0.291955

$\varepsilon = 0.1$

0.772135	0.557292	0.481771	0.535156	0.417969
0.454427	0.608724	0.777344	0.712891	0.365885
0.454427	0.608724	0.777344	0.712891	0.365885
0.772135	0.557292	0.481771	0.535156	0.417969
0.417969	0.342448	0.380208	0.391927	0.917969

$\varepsilon = 0.01$

0.255225	0.192529	0.161311	0.181105	0.148804
0.150328	0.204874	0.297853	0.279902	0.123723
0.150328	0.204874	0.297853	0.279902	0.123723
0.255225	0.192529	0.161311	0.181105	0.148804
0.148804	0.115792	0.125277	0.129947	0.648804

$\varepsilon = 0.001$

0.206391	0.157315	0.131311	0.147289	0.123271
0.121202	0.165481	0.252885	0.238541	0.100422
0.121202	0.165481	0.252885	0.238541	0.100422
0.206391	0.157315	0.131311	0.147289	0.123271
0.123271	0.093878	0.101394	0.104997	0.623271

$\varepsilon = 0.0001$

0.201747	0.153967	0.128458	0.144073	0.120844
0.118432	0.161735	0.248609	0.234608	0.098207
0.118432	0.161735	0.248609	0.234608	0.098207
0.201747	0.153967	0.128458	0.144073	0.120844
0.120844	0.091794	0.099123	0.102625	0.620844

Nel confronto 20 vediamo cosa significa avere due nodi sorgenti nei due automi, gli stati 4 dei due rispettivi dfa:

il loro vicinato entrante è l'insieme vuoto, quindi il loro punteggio di similarità entrante risulta $\frac{0}{0} = 1$, qualunque sia poi il punteggio di similarità uscente, quello di similarità globale sarà dunque $> \frac{1}{2}$. Questo punteggio abbastanza elevato e fisso funge da una sorta di ancora che mantiene elevato il punteggio di similarità di tutti i nodi a cui confluiscono i nodi sorgente, visto che la formula prevede di prendere ogni volta il matching ottimo tra il vicinato entrante dei nodi per calcolarne la similarità alla iterazione.

Nel confronto 19 non abbiamo questo contributo fisso di $\frac{1}{2}$ che si propaga altrove. Questo fattore (ricordando che la condizione di terminazione è $\max_{ij} |x_{ij}^k - x_{ij}^{k-1}| < \varepsilon$) era sicuramente quel max tra i punteggi di similarità da attenzionare per far fermare le iterazioni dell'algoritmo, e poiché c'era un grosso contributo monopolizzante, la differenza tra una iterazione e l'altra era relativamente piccola.

Mancando tutto questo l'algoritmo continua a iterare fino a quando i punteggi di similarità internodo non aggiungono quello zero dopo la virgola che consenta di, sottraendoli ai punteggi della iterazione precedente, si mantengano al di sotto della soglia.

La mia conclusione è che la condizione di terminazione formulata così non va bene.