

Sincronizzazione in ensemble musicali

Dipartimento Fisica e Astronomia
Università Degli studi di Catania

Torrisi Giuseppe

27 novembre 2014

Sommario

Con questo lavoro si intende affrontare il problema della sincronizzazione in ensemble musicali in forma rettangolare tramite l'ambiente di simulazione Netlogo; tale configurazione è caratteristica di tutti i tipi di marching band. Da alcune assunzioni sul comportamento del singolo elemento e della sua interazione con gli altri membri del gruppo, emerge una continua competizione tra ordine e disordine il cui studio rappresenta, al variare di un solo importante parametro di controllo, l'obiettivo di tale lavoro.

Indice

I	Introduzione	2
1	Convenzioni musicali	2
II	Funzionamento	3
2	Setup	3
3	Procedura SUONIAMO	4
3.1	Procedura Sincronizziamo	5
3.2	Fuori tempo	6
III	Analisi dei risultati	7
4	Dispersione dei tempi	7

5	Comportamento a cluster	9
6	Conclusioni	10
IV	Pensieri successivi	12
7	Confrontiamo i Fit	13

Parte I

Introduzione

1 Convenzioni musicali

Indicherò alcune convenzioni di teoria musicale che userò in tale trattazione, necessarie per confrontare il modello costruito con l'esperienza musicale reale.

Scelgo come ipotesi di lavoro (come avviene spesso nelle marching-band)l'utilizzo di tempi quaternari, con una indicazione del tipo in figura.



Sto quindi imponendo che l'unità fondamentale di tempo in uso è la semiminima, con una frequenza di $120 \frac{\text{semiminima}}{\text{minuto}}$: a questa ci si riferirà come "*velocità di esecuzione*". Supposto un tempo di esecuzione pari a 8 minuti vi saranno $\frac{120 \cdot 8}{2} = 500$ battute. Suppongo, supportato dal fatto che l'accento musicale cade all'inizio di ogni battuta, che tutti i fenomeni che coinvolgono la scelta del tempo da parte del singolo musicista avvenga all'inizio della stessa; questa schematizzazione è approssimativa dato che il cervello fissa il tempo ottimale di esecuzione in maniera continua ed istintiva, ma per poter affrontare tale problema è necessario discretizzare; da tale assunzione segue che il numero di iterazioni rappresenta il numero di battute eseguito.

Parte II

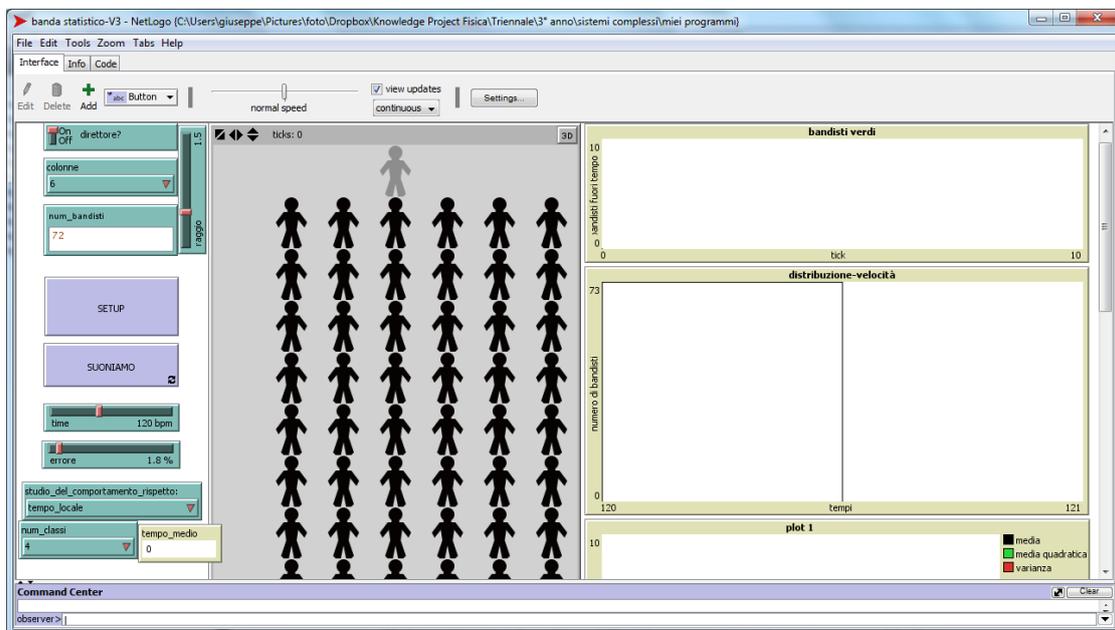
Funzionamento

2 Setup

L'ambiente di simulazione lavora in due differenti modalità tramite il chooser:

- direttore? on
- direttore? off

Da un punto di vista operativo questo conduce a due differenti modalità di simulazioni che saranno oggetto delle sezioni successive, da un punto di vista grafico la presenza del direttore è data, dopo la pressione del pulsante *setup*, da:



Il world è popolato dalla breed "bandisti" il cui numero è impostato dall'utente tramite la variabile *num_bandisti*, a cui si aggiunge il direttore se l'opzione è stata scelta. Vengono schierati secondo il codice

```
create-direttori 1
create-bandisti num_bandisti
let j 0
let i 0
```

```

let k 1
ask direttore 0 [setxy 0 max-pycor]
loop
[
  set j 0;
  while [j < colonne]
  [
    ask bandista k [setxy (min-pxcor + j + 1) (max-pycor - i - 1) ]
    set j j + 1
    set k k + 1
    if k > num_bandisti [stop]
  ]
  set i i + 1
]

```

Gli agenti sono caratterizzati da due variabili proprietarie.

```
turtles-own[tempo sigma ]
```

Inizialmente si assume che tutti i musicisti siano sincronizzati alla velocità imposta dall'utente tramite uno slider.

In queste condizioni l'ensemble è pronto per suonare, quindi può partire la simulazione tramite il pulsante "SUONIAMO"; si fa osservare che i parametri visibili in figura 1 sono stati conservati nell'arco di tutta la simulazione, ad eccezione dell'errore percentuale ove espressamente indicato

3 Procedura SUONIAMO

E' cruciale riuscire a schematizzare nella maniera più fedele possibile il comportamento del singolo agente, poichè da tale operazione dipende la bontà dei risultati proposti. In questa simulazione si è ridotto il problema all'analisi di due fattori fondamentali:

- Un termine di sincronizzazione dell'individuo che deriva dall'interazione con gli altri musicisti
- un termine di asincronia di origine aleatoria, che dipende esclusivamente dalla velocità di esecuzione del singolo agente, quindi indipendente dal comportamento degli altri musicisti

La questione spinosa riguarda la forma funzionale da attribuire ai termini di sincronizzazione e di asincronia.

3.1 Procedura Sincronizziamo

Si suppone che l'ensemble sia diviso in diversi gruppi per tipologia di strumento , il numero di tali gruppi è definito tramite la variabile *num_classi*; per ciascuna classe è stata calcolata la media di classe come:

```
while[j < num_classi][  
  
    set media_classe mean [tempo] of bandisti with [ who < (j + 1)  
    * ceiling ( num_bandisti / num_classi)  
    and who >  j * ceiling( num_bandisti / num_classi) ]  
    ...  
    set j j + 1 ]
```

In tale modello si è descritto il termine di sincronizzazione attribuendo, a ciascun agente, la media dei tempi dei suoi primi vicini in un raggio impostato dall'utente, che poi viene mediata rispetto alle diverse *media_classe* ovvero¹

```
let j 0  
let media_classe 0  
  
ask bandisti  
[  
    let lista-tempi [tempo] of bandisti in-radius raggio  
    set lista-tempi lput ([tempo] of turtle 0) lista-tempi  
    while[j < num_classi][  
  
        set media_classe mean [tempo] of bandisti with  
        [ who < (j + 1) * ceiling ( num_bandisti / num_classi)  
        and who >  j * ceiling( num_bandisti / num_classi) ]  
  
        set lista-tempi lput media_classe lista-tempi  
        set j j + 1 ]  
    set tempo mean lista-tempi  
    ]  
end
```

¹Ho aggiunto anche il tempo della turtle 0 nella lista poichè rappresenta la velocità del direttore: essa viene considerata solo nel caso in cui il chooser sia in modalità on

3.2 Fuori tempo

Riguardo il termine di asincronia è stato necessario fare alcune approssimazioni:

- Ho assunto che a ogni tick l'effettiva velocità di esecuzione non è esattamente quella "pensata" dal musicista come era stato calcolato nella sezione precedente, ma segue una distribuzione normale centrata sul valore desiderato. Indipendentemente da quale sia la causa di tale errore (se le difficoltà tecniche o quelle intrinseche nel mantenere il tempo) è ragionevole supporre che la sua funzione di distribuzione debba seguire un andamento a campana.

```
ask bandisti [ set tempo random-normal tempo sigma ]
ask direttori [ set tempo random-normal time sigma ]
```

2

- Resta da caratterizzare la distribuzione normale mediante una varianza, l'ipotesi che sta alla base di tale modello è la dipendenza lineare della varianza dalla velocità di esecuzione tramite un errore percentuale impostato come parametro dall'utente. Questa ipotesi rispecchia l'evidenza sperimentale sulla monotonia dell'errore commesso al crescere della velocità, ed almeno l'approssimazione al primo ordine è coerente con l'ipotesi fatta.

```
ask turtles [set sigma (errore * tempo / 100) ]
```

Infine gli agenti vengono colorati in gradazioni di rosso in base al loro tempo. Il risultato finale è riportato nelle figure 2a ed in 1

²Nel caso in cui ci fosse la modalità direttore attiva si immagina che anch'egli segua una distribuzione gaussiana ma centrata sempre sulla velocità preimpostata dall'utente



Figura 1: Schermata output

Parte III

Analisi dei risultati

4 Dispersione dei tempi

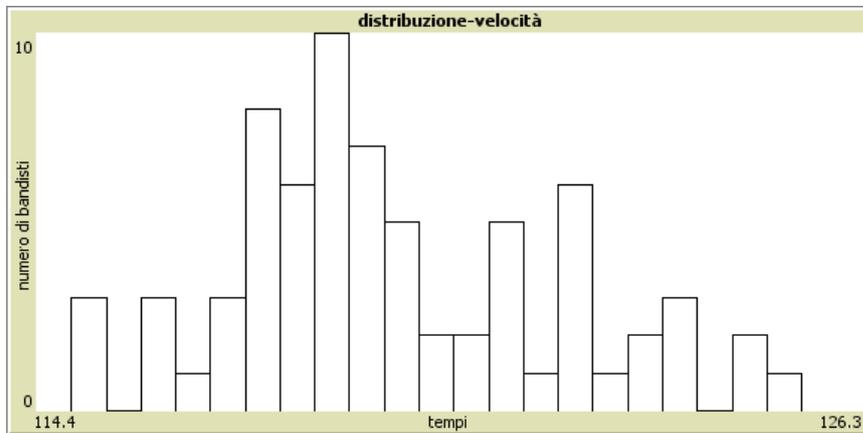
Per quanto detto nella parte I la simulazione è stata interrotta dopo 500 tick, che rappresenta approssimativamente l'intera esecuzione musicale; analizzo in questa fase la dispersione delle velocità.

Da calcoli statistici si ottiene che la varianza sulla popolazione delle velocità è $\sigma = 5.6$ ovvero il 4.9% della velocità media nella configurazione finale $\bar{v} = 119.8$.

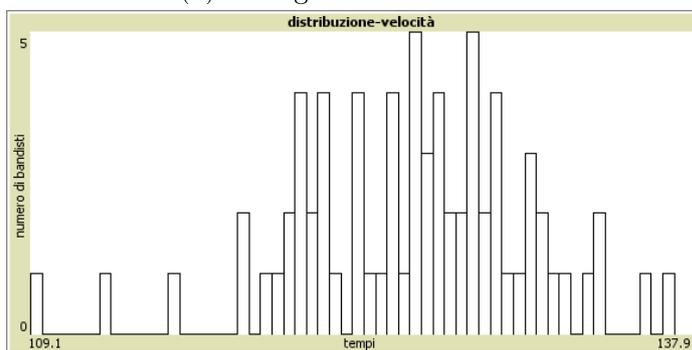
Bisogna verificare che un range così ampio nella distribuzione delle velocità di esecuzione come indica 2a sia realistico; a tal proposito si confrontano l'intervallo di tempo corrispondente allo scarto tra gli estremi, ovvero $\delta v = \frac{v_{max} - v_{min}}{2} = 5.3$, con il più piccolo intervallo di tempo che generalmente viene eseguito in un ensemble e corrisponde a $\frac{1}{8 \cdot \text{velocità esecuzione}}$ ³; ovvero:

$$\left| \frac{1}{\bar{v}} - \frac{1}{\bar{v} \pm \delta v} \right| < \frac{1}{8 \cdot \bar{v}} \quad (1)$$

³Sto assumendo che il margine di errore nel tempo sia dell'ordine della biscroma: per la velocità di esecuzione in esame l'intervallo di tempo di accettazione è dell'ordine del ms



(a) Configurazione con errore 1.8%



(b) Configurazione con errore pari al 3 %

Figura 2: Distribuzioni velocità di esecuzione dopo 500 ticks, $time = 120$

da cui si ottiene approssimativamente

$$\delta v < \frac{\bar{v}}{8} \approx 14$$

Tale modello prevede che vi siano delle differenze di velocità di esecuzione, ma esse sono tali da non inficiare l'apparente sincronia dell'ensemble; ciò è vero per i parametri scelti, basta osservare che scegliendo l'errore percentuale pari a 3% per ottenere δv più di due volte maggiori come indica 2b per cui l'asincronia diventa percepibile.

Un ulteriore aspetto interessante è dato dal fatto che la distribuzione delle velocità non è simmetrica, media e mediana non coincidono a tal proposito si osserva che i momenti di ordine dispari rispetto alla media sono in generale diversi da 0; è da notare che, per l'ipotesi di dipendenza lineare della varianza del singolo agente nella velocità di esecuzione, le velocità di esecuzione più bassa sono più stabili rispetto al caso con velocità maggiori

5 Comportamento a cluster

Sono state colorati i vari agenti in gradazioni di rosso in base alla loro velocità di esecuzione, ciò permette da un'analisi qualitativa della formazione di gruppi di strumenti vicini che seguono tempi simili

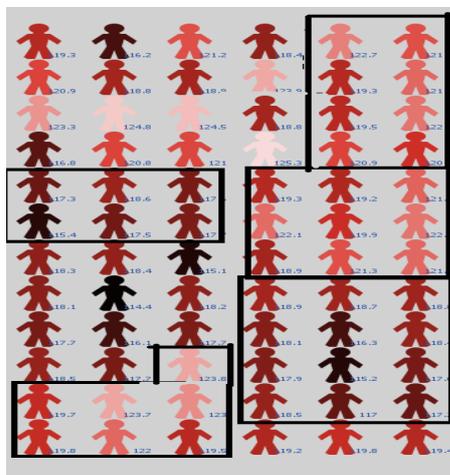


Figura 3: world di Netlogo
I riquadri racchiudono gli agenti con velocità simili

Le fluttuazioni fin qui osservate, ad esempio sulla velocità media di esecuzione, sono particolarmente ridotte poiché ci siamo concentrati su poche iterazioni; come dimostra la figura 5a su un numero arbitrariamente grande di iterazioni si osservano in genere fluttuazioni anche molto pronunciate rispetto alle condizioni iniziali, che sembrano totalmente incompatibili con la possibilità che un ensemble suoni insieme. Questo è dovuto al fatto che su un numero elevato di iterazioni anche gli eventi meno probabili possono dar luogo a effetti macroscopici apprezzabili.⁴

Nelle simulazioni si è osservato che la figura del direttore fissa il valore medio della velocità di esecuzione, anche in presenza di grandi errori percentuali, come mostra la figura 5b.

⁴Si potrebbe pensare che all'aumentare del tempo la distribuzione delle velocità si allarghi, ciò non è vero come dimostra la Figura 4

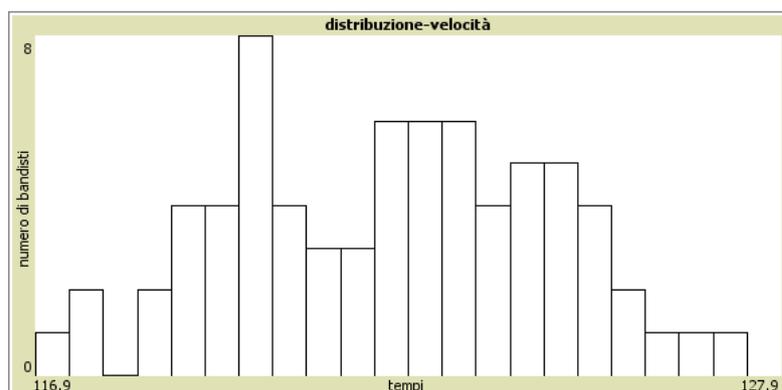


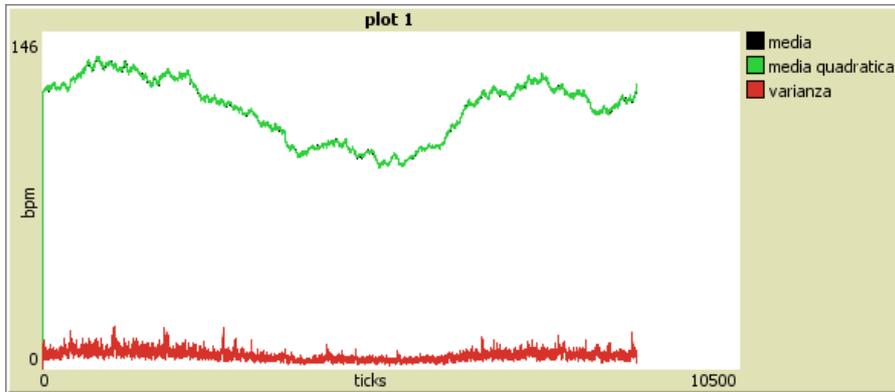
Figura 4: Distribuzione delle velocità di esecuzione su tempi lunghi nelle condizioni di riferimento

Osserva che lo scarto tra gli estremi è analogo a quello di 2a

6 Conclusioni

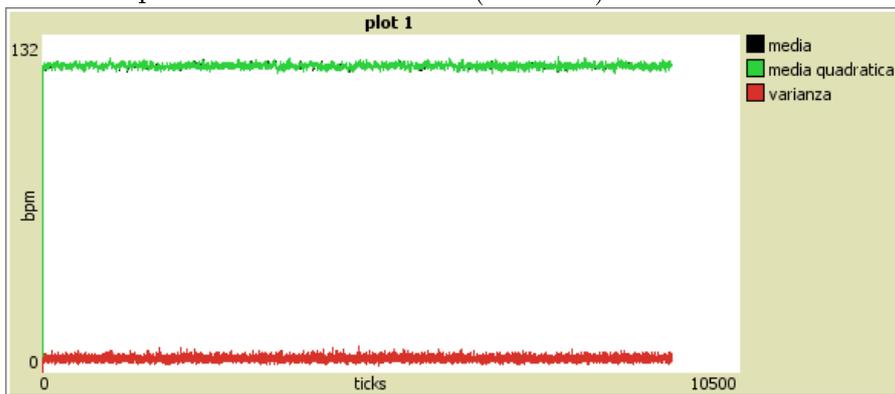
Uno dei limiti di tale modello è la non riproducibilità dei risultati da esso forniti; questo è un limite intrinseco che deriva dall'utilizzo di variabili di natura puramente casuale; nonostante le approssimazioni che sono stati via via sottolineate nel corso della presentazione, tale modello riesce a fornire dei risultati particolarmente interessanti sulle fluttuazioni nella distribuzioni delle velocità di esecuzione, ed in tale direzione si può pensare a verifiche sperimentali; ulteriore successo di tale modello sta nella capacità di prevedere il comportamento a cluster nella distribuzione dei tempi, fenomeno che si manifesta spontaneamente negli ensemble musicali.

Tale modello può essere generalizzato per descrivere, sotto opportuni parametri, sistemi di agenti interagenti tramite meccanismi di feedback in uno spazio metrico, soggetti a un generica forma di rumore descrivibile in termini di una distribuzione gaussiana.



(a) Caso senza direttore

Nota la dipendenza tra la varianza (in rosso) e la media della velocità di esecuzione.



(b) Con direttore

Figura 5: Grandezze di riferimento su un numero grande di iterazioni

Parte IV

Pensieri successivi

Ho provato a togliere la media di classe, fissare la sigma costante (non indipendente dalla velocità di esecuzione) e mi sono concentrato sulla forma della distribuzione delle velocità; si è osservato che in generale anche errori del 3 mi distruggono ogni forma di simmetria della distribuzione, se metto errori bassi ad esempio 1, si hanno forme molto più regolari che potrebbero assomigliare a gaussiane. Ho usato questo codice per gnuplot, da cui si ottiene

```
a=0.7
set boxwidth a
classe(x, a) = a*int(x/a) + a/2.
set table "hist.dat"

plot 'C:\Users\giuseppe\Pictures\foto\Dropbox\Knowledge Project Fisica\
Triennale\
3°anno\ sistemi complessi\ miei programmi\ tempi.dat' u (classe($1,a)):(1./Nt)
smooth frequency t '' w boxes

unset table

g(x)=(N/(sqrt(2*pi)*s))*exp(-(x-m)**2/(2*s**2))

fit g(x) 'C:\Users\giuseppe\Pictures\foto\Dropbox\Knowledge Project Fisica\
Triennale\
3° anno\sistemi complessi\miei programmi\hist.dat' via m,s,N

plot 'C:\Users\giuseppe\Pictures\foto\Dropbox\Knowledge Project Fisica\
Triennale\3° anno\sistemi complessi\miei programmi\hist.dat',g(x)
```

Come si vede

Il codice per il fit sia con la q-gaussiana è:

```
qg(x)=A*sqrt(b)*e_q(-b*((x-m)**2),q)/C(q)
e_q(x,q)=(1+((1-q)*x))**(1/(1-q))
```

```
C(q)=sqrt(3.14)*gamma((3-q)/(2*(q-1)))/(sqrt(q-1)*gamma(1/(q-1)))
```

fit qg(x) 'C:\Users\giuseppe\Pictures\foto\Dropbox\Knowledge Project Fisica\Triennio
 3° anno\sistemi complessi\miei programmi\hist.dat' via q,b,A,m

7 Confrontiamo i Fit

Ho fittato sia con q-gaussiana che con la gaussiana (q=1)

- Il report del fit di figura 6a

After 46 iterations the fit converged.

final sum of squares of residuals : 8.36908

rel. change during last iteration : -4.24505e-016

degrees of freedom (FIT_NDF) : 5
 rms of residuals (FIT_STDFIT) = sqrt(WSSR/ndf) : 1.29376
 variance of residuals (reduced chisquare) = WSSR/ndf : 1.67382

Final set of parameters		Asymptotic Standard Error	
=====		=====	
q	= 1.89057	+/- 0.1677	(8.872%)
b	= 1.69664	+/- 0.3063	(18.05%)
A	= 94.844	+/- 7.605	(8.018%)
m	= 123.362	+/- 0.0266	(0.02156%)

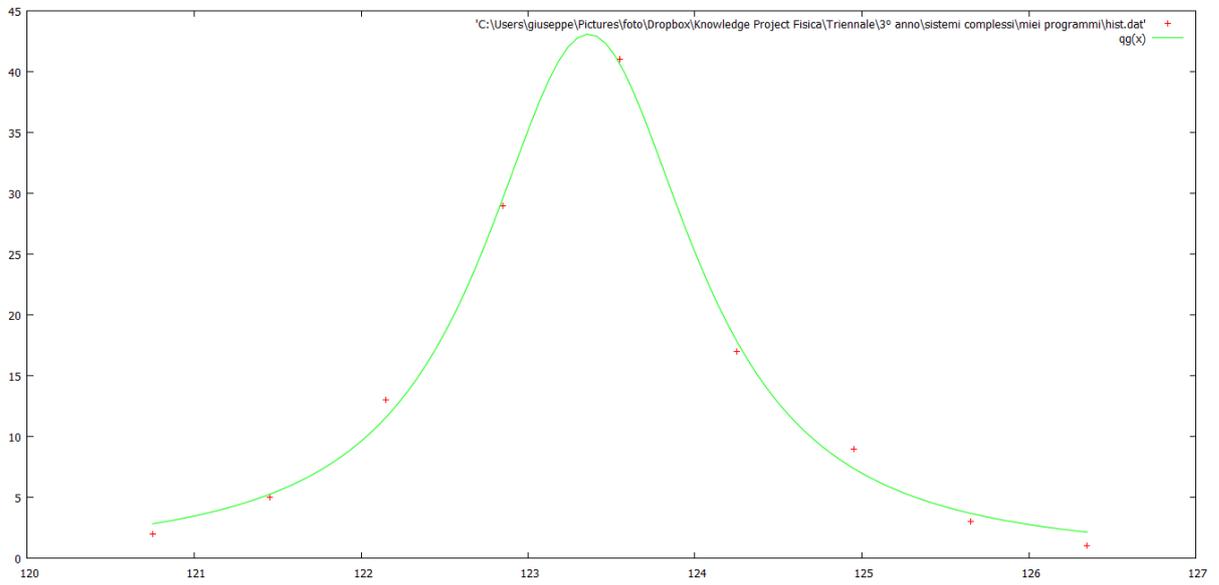
correlation matrix of the fit parameters:

	q	b	A	m
q	1.000			
b	0.856	1.000		
A	0.912	0.631	1.000	
m	-0.270	-0.290	-0.216	1.000

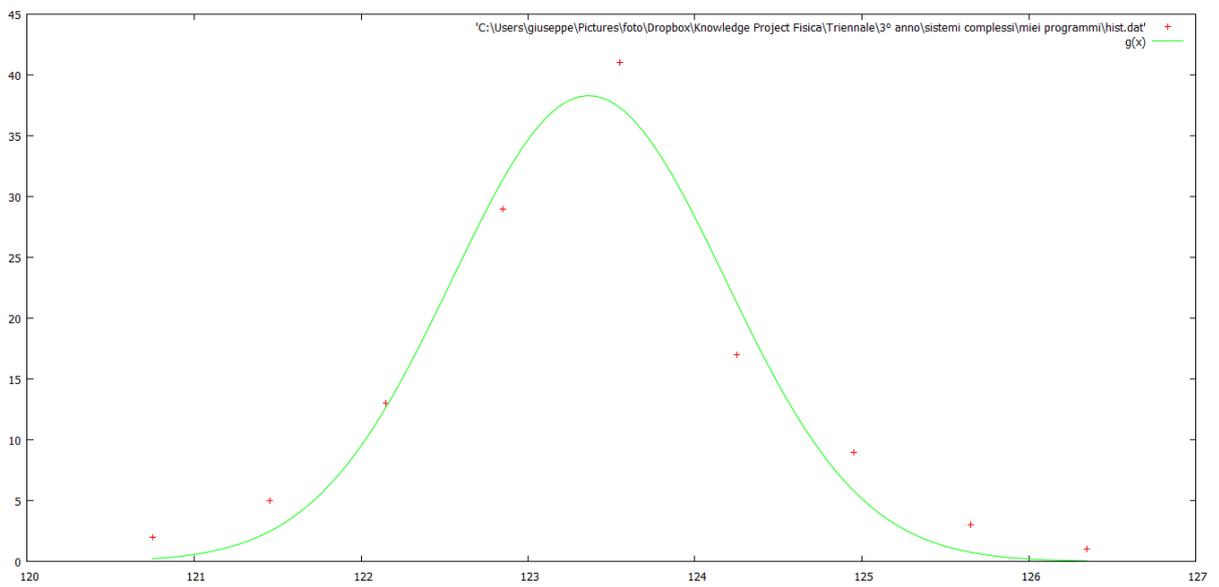
- Il report del Fit di figura 6b è

After 8 iterations the fit converged.

final sum of squares of residuals : 63.3537



(a) Fit con q-gaussiana



(b) Fit con gaussiana

rel. change during last iteration : -9.08056e-006

degrees of freedom (FIT_NDF) : 6
rms of residuals (FIT_STDFIT) = sqrt(WSSR/ndf) : 3.24945
variance of residuals (reduced chisquare) = WSSR/ndf : 10.5589

Final set of parameters		Asymptotic Standard Error	
=====		=====	
N	= 78.467	+/- 5.668	(7.224%)
s	= 0.817105	+/- 0.06841	(8.372%)
m	= 123.364	+/- 0.06853	(0.05555%)

correlation matrix of the fit parameters:

	N	s	m
N	1.000		
s	0.576	1.000	
m	0.000	-0.075	1.000