

# Relazione Laboratorio di Calcolo: Oscillatore armonico smorzato

Alessandra Arcuri

Agosto 2019

## 1 Introduzione teorica

Si consideri un **oscillatore armonico smorzato** in una dimensione, costituito da una massa attaccata ad una molla e immersa in un mezzo resistente. Essa sarà soggetta ad una forza complessiva che comprende sia la forza elastica (data dalla *legge di Hooke*), sia la forza di attrito lineare (espressa dalla *legge di Stokes*):

$$F_{tot} = -kx - cv \quad (1)$$

Tenendo in considerazione la seconda legge della dinamica, si ottiene così l'equazione differenziale **non omogenea** corrispondente:

$$\frac{d^2x}{dt^2} + \frac{c}{m} \frac{dx}{dt} + \frac{k}{m} x = 0 \quad , \quad (2)$$

dove  $c$  e  $k$  sono costanti reali e positive. Il seguente programma si propone di risolvere l'equazione differenziale implementando l'algoritmo Euler-trapezoidale per la soluzione numerica di equazioni differenziali del secondo ordine.

Lo schema del programma è stato elaborato su modello dell'oscillatore armonico non smorzato dell'esercitazione n. 5.

### 1.1 L'algoritmo Euler-Trapezoidale

Considerando innanzitutto l'espressione della forza come  $f(t) = f(x(t), v(t), t)$ , l'algoritmo è quindi descritto dalle seguenti formule, dove  $dt$  è l'intervalllo infinitesimo di cui verrà incrementato l'algoritmo ad ogni iterazione :

---

#### Algoritmo Euler-trapezoidale

---

$$\begin{aligned} x(t + dt) &= x(t) + v(t) * dt + 0.5 * f(t)/m * dt * *2 \\ vp(t + dt) &= v(t) + f(t)/m * dt \\ fp(t + dt) &= f(x(t + dt), vp(t + dt), t + dt) \\ v(t + dt) &= v(t) + 0.5 * (f(t)/m + fp(t + dt)/m) * dt \\ f(t + dt) &= f(x(t + dt), v(t + dt), t + dt) \end{aligned}$$

---

## 1.2 Formule analitiche

Dopo aver implementato l'algoritmo, è stato verificato il suo funzionamento (ma soprattutto anche la sua accuratezza), attraverso la soluzione analitica dell'equazione differenziale per le posizioni:

$$x(t) = x_0 \cdot \exp\left[\frac{-c}{2m}t\right] \cos(wt) + \frac{1}{w} \cdot \left(v_o + \frac{c}{2m}x_0\right) \cdot \exp\left[\frac{-c}{2m}t\right] \sin(wt) \quad (3)$$

La formula analitica per le velocità è data dalla derivata rispetto al tempo della formula precedente:

$$\frac{dx(t)}{dt} = v(t) = v_0 \cdot \exp\left[\frac{-c}{2m}t\right] \cos(wt) - \frac{1}{w} \cdot \left(x_0 \frac{c^2}{(2m)^2} + \frac{c}{2m}v_0 + x_0w^2\right) \cdot \sin(wt) \quad (4)$$

dove  $w$  è la pulsazione e vale:

$$w = \sqrt{\frac{k}{m} - \left(\frac{c}{2m}\right)^2}$$

Nota: Nel programma si farà uso della variabile  $gam = c/(2m)$ .

Il risultato restituito dall'algoritmo differisce leggermente da quello dato dalle formule analitiche in quanto, nello svolgere il passaggio al discreto (quindi considerando l'evoluzione di **posizione** e **velocità** su piccoli intervalli discreti di tempo di ampiezza  $dt$ ), si generano degli *errori* ad ogni passaggio di iterazione dell'algoritmo, di cui si discuterà in seguito.

## 2 Verifica tramite grafico dell'equazione del moto e della velocità

Il programma è stato realizzato in modo tale da ricevere in input i seguenti valori:

1. Passo temporale **dt**, ossia l'ampiezza dell'intervallino di tempo ad ogni iterazione;
2. Il numero di punti o di step, **nstep**, il quale restituisce il tempo totale della simulazione quando moltiplicato per  $dt$ ;
3. La **massa** da utilizzare;
4. La costante elastica **k** della molla;
5. La costante di attrito lineare **c** (il cui valore influisce sul tipo di smorzamento, del quale verrà discusso più dettagliatamente in seguito);
6. **pos e vel**, ossia i valori di posizione e velocità iniziali.

Nel programma si è fatto uso di due variabili in input, **pos\_an** e **vel\_an**, le quali corrispondono ai valori iniziali di posizione e velocità da inserire all'interno delle due formule analitiche. Il loro valore deve quindi essere uguale a quello di **pos** e **vel** se si vuole confrontare il risultato ottenuto con l'algoritmo con quello analitico.

### 2.0.1 Valutazione degli errori

Al fine di ottenere risultati maggiormente in accordo con la teoria, è necessario valutare il valore della potenza di **dt** che permette di minimizzare l'errore sulla posizione e sulla velocità. Ciò può essere fatto procedendo per tentativi e tenendo in considerazione i diversi tipi di errore:

- Errori di troncamento: dovuti al fatto che il computer non può immagazzinare una quantità infinita di cifre significative; per questo motivo si effettua appunto un "troncamento" di esse. Questo tipo di errori può essere valutato raddoppiando il numero di intervallini, mantenendo costante l'ampiezza dei singoli *dt*;
- Errori di discretizzazione: si presentano in quanto gli algoritmi numerici computano approssimazioni *discrete* per quantità continue (ad esempio l'integrale viene approssimato con una sommatoria). Nel caso del programma, questo tipo di errori può essere valutato raddoppiando l'ampiezza del passo *dt*, senza modificare il numero di punti;
- Errori di arrotondamento: essi sono dovuti ai calcoli effettuati dal computer (o meglio, dalla rappresentazione dei numeri che il computer deve adottare). Possono essere valutati facendo attenzione al numero di cifre significative di ogni variabile, il quale dipende dal **kind** utilizzato per esse. Nel programma si è deciso di utilizzare i seguenti kind per le variabili di tipo REAL e per gli INTEGER, i quali sono stati inseriti nel modulo *prec*:

```
module prec
implicit none
  integer, parameter :: rk=selected_real_kind(16)
  integer, parameter :: ik=selected_int_kind(12)
end module prec
```

Figure 1: Kind utilizzati

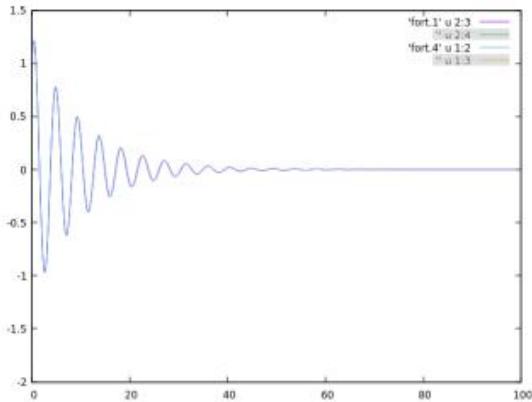
Il comando per plottare con *gnuplot* la legge oraria e la velocità è il seguente:

```
p 'fort.1' u 2:3 w 1, " u 2:4 w 1, ' fort.4 ' u 1:2 w 1, " u 1:3 w 1
```

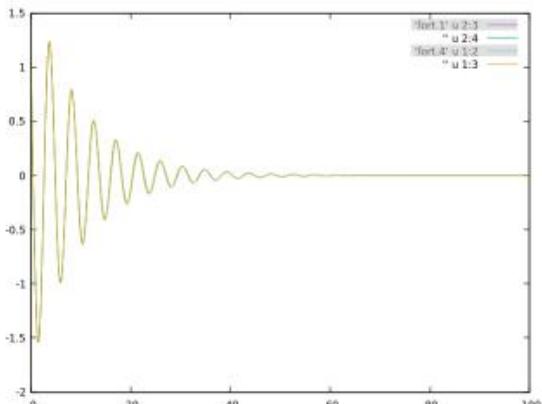
dove **fort.1** è il file contenente la posizione e la velocità date dall'algoritmo, mentre **fort.4** contiene la posizione e la velocità calcolate con le formule analitiche. D'ora in poi sarà omesso il fatto che **pos\_an** e **vel\_an** hanno sempre lo stesso valore di **pos** e **vel**.

In seguito vengono riportati i diversi tentativi effettuati con relativo commento:

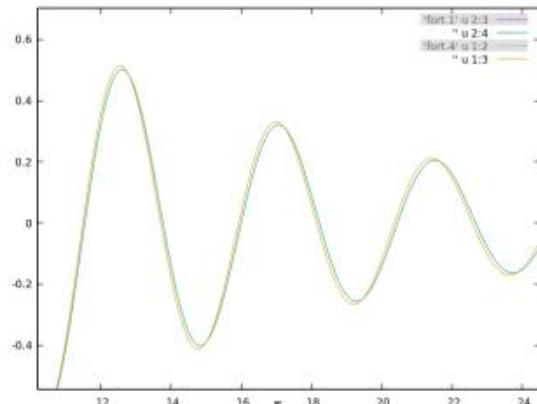
**1) massa: 5; kappa: 10; dt: 0.1; n.step: 1000; c: 1, pos: 1, vel: 1**



(a) posizione in funzione del tempo



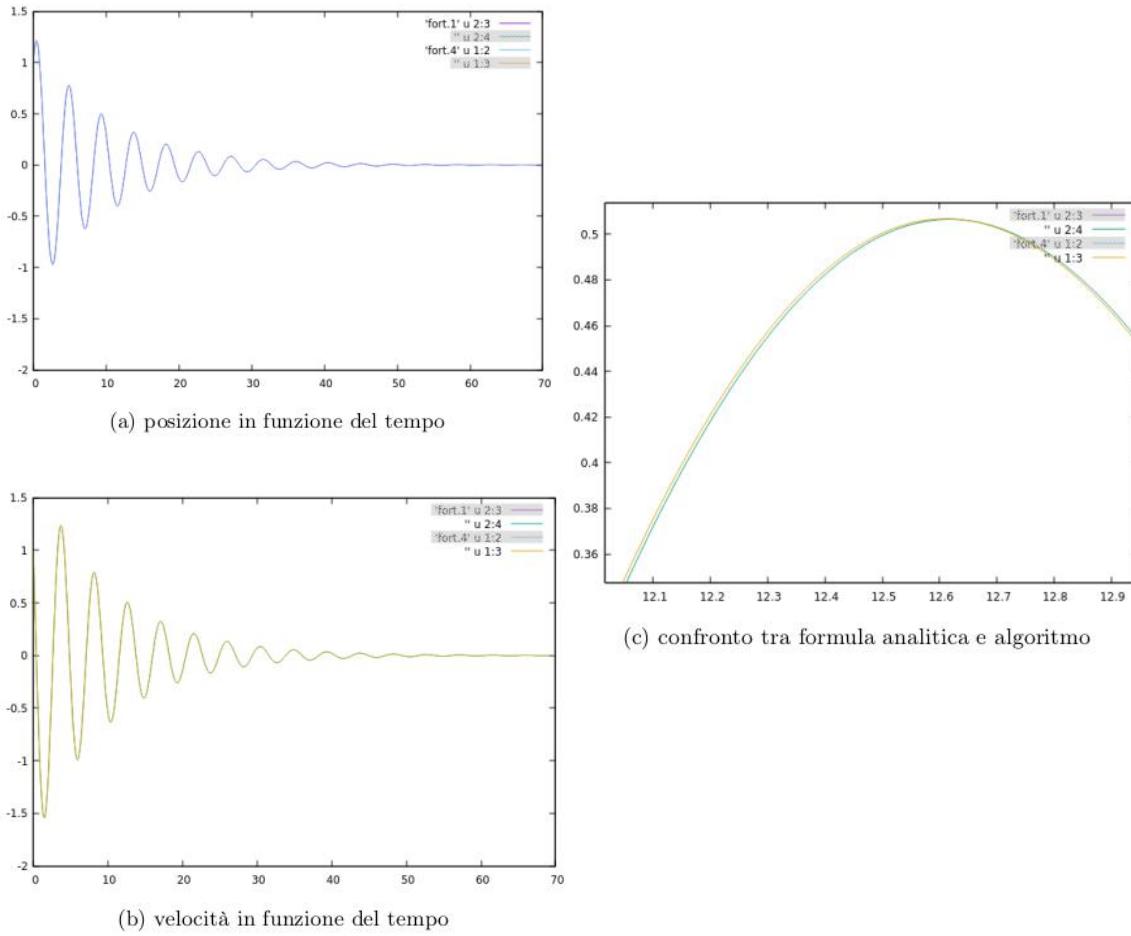
(b) velocità in funzione del tempo



(c) confronto tra formula analitica e algoritmo

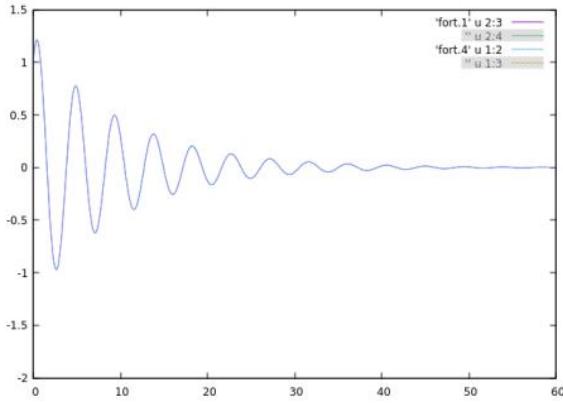
Nella figura (a) è stata graficata la posizione in funzione del tempo, sia utilizzando l'algoritmo, sia utilizzando la formula analitica. A prima occhiata non è visibile la differenza tra i due metodi, tant'è che le due curve sembrano perfettamente sovrapposte; la stessa cosa succede per la velocità (figura (b)). Nel momento in cui si ingrandisce il grafico, però, la differenza tra le due curve è visibile già quando la posizione è dell'ordine del  $10^{-1}$  (figura (c)). L'errore può essere migliorato diminuendo innanzitutto l'ampiezza degli intervallini  $dt$ ; si è scelto anche di aumentare il numero di punti (infatti il grafico, se ingrandito ulteriormente, risulta essere poco "liscio"); ciò è necessario anche per essere in grado di vedere interamente il grafico dell'oscillazione smorzata (cioè che arriva a 0).

**2) massa: 5; kappa: 10; dt: 0.01; n.step: 7000; c: 1, pos: 1, vel: 1**

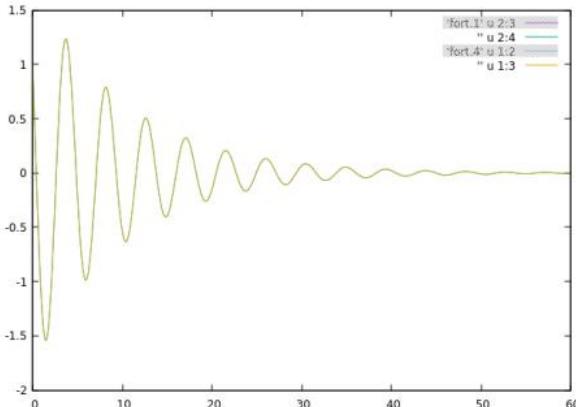


Come si può evincere dalla terza figura, in questo caso è stato necessario ingrandire il grafico della velocità rispetto al tempo fino all'ordine del  $10^{-2}$ . Ciò vuol dire che la differenza tra la formula analitica e quella dell'algoritmo è diventata minore. Per il terzo tentativo si procede analogamente come per il secondo, diminuendo ancora una volta di un fattore 10 l'ampiezza degli intervallini  $dt$  e aumentando il numero di punti per poter riportare la totale ampiezza del grafico:

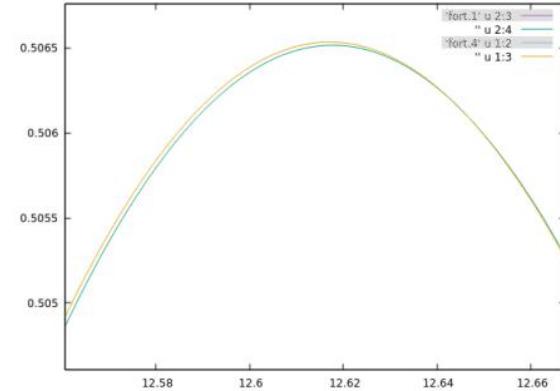
**3) massa: 5; kappa: 10; dt: 0.001; n.step: 60000; c: 1, pos: 1, vel: 1**



(a) posizione in funzione del tempo



(b) velocità in funzione del tempo

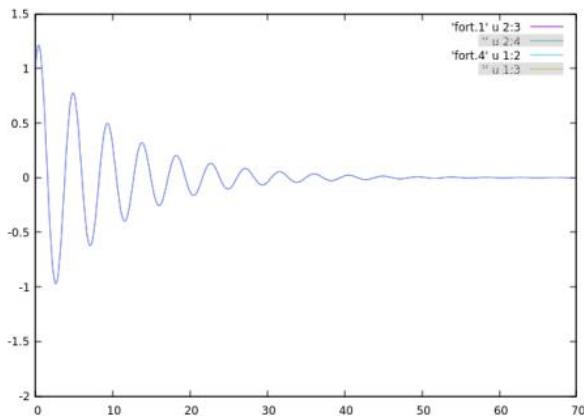


(c) confronto tra formula analitica e algoritmo

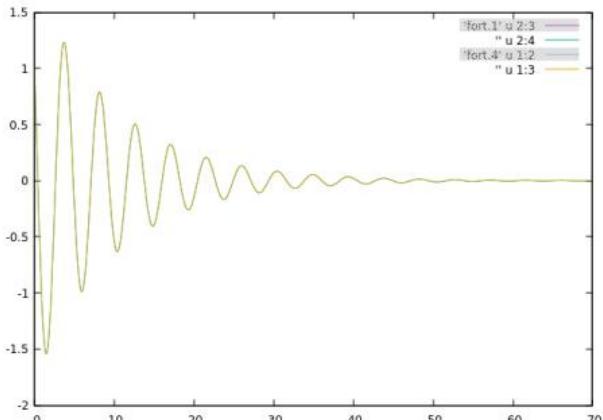
In questo caso la differenza è apprezzabile nel momento in cui si ingrandisce il grafico velocità-tempo fino all'ordine del  $10^{-3}$  (quasi  $10^{-4}$ ). A questo punto si prova ancora a diminuire l'ampiezza  $dt$  di un fattore 10, fino a portare il numero di step a 700000. Chiaramente il computer ora lavora più lentamente, infatti, usando il comando **time** dello shell, si scopre che il tempo di esecuzione del processo associato al programma è arrivato oltre i 30 s:

real	0m34,821s
user	0m14,130s
sys	0m0,812s

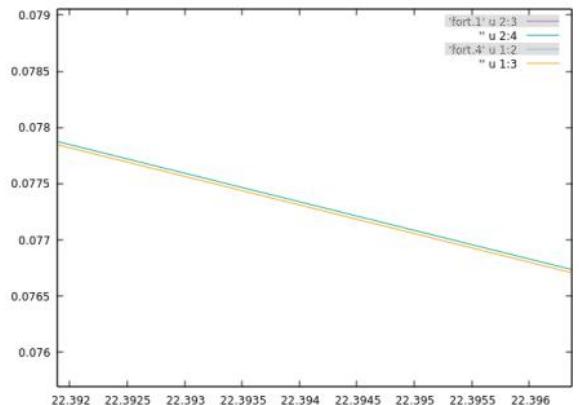
4) massa: 5; kappa: 10; dt: 0.0001; n.step: 700000; c: 1, pos: 1, vel:



(a) posizione in funzione del tempo



(b) velocità in funzione del tempo



(c) confronto tra formula analitica e algoritmo

Dalla terza figura si vede che di nuovo l'ordine di precisione è del  $10^{-3}$ ; i risultati sono simili al tentativo precedente. Diminuendo ancora l'ampiezza degli intervallini  $dt$ , diventa sempre più difficile riuscire a graficare anche solo un'oscillazione.

### 3 Verifica della perdita di energia meccanica

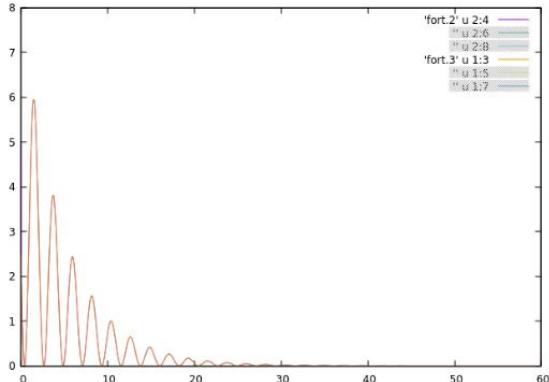
L'oscillatore smorzato è un sistema *non conservativo*, in quanto la presenza della forza di attrito lineare è una forza **dissipativa**: ciò implica quindi che l'**energia meccanica totale del sistema non si conserva**, anzi tende a 0, così come l'energia cinetica e l'energia potenziale elastica del sistema. Ciò può essere verificato graficamente; in questo caso i valori utilizzati sono i seguenti:

**massa: 5; kappa: 10; dt: 0.001; n.step: 60000; c: 1, pos: 1, vel: 1**

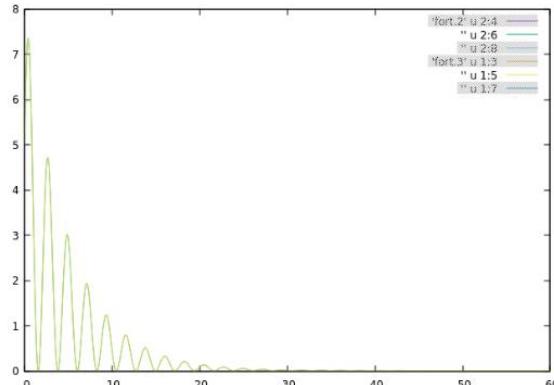
mentre il comando utilizzato per graficare l'energia cinetica, potenziale e meccanica è il seguente, dove nel file **fort.2** ci sono i valori ricavati con l'algoritmo, mentre nel file **fort.3** quelli ricavati con le formule analitiche:

```
p 'fort.2' u 2:4 w 1, " u 2:6 w 1, " u 2:8 w 1, 'fort.3' u 1:3 w 1, " u 1:5
```

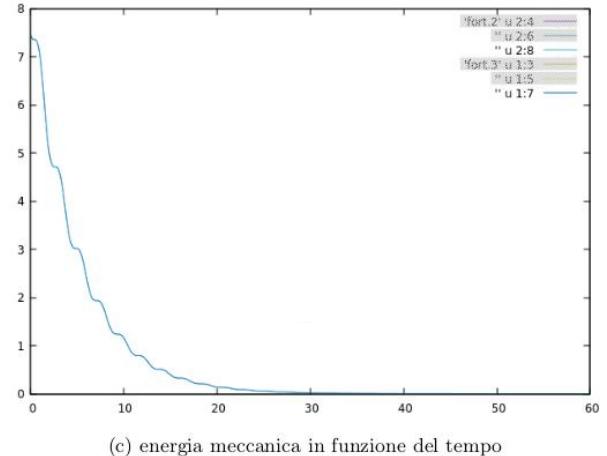
```
w 1, " u 1:7 w 1
```



(a) energia cinetica in funzione del tempo



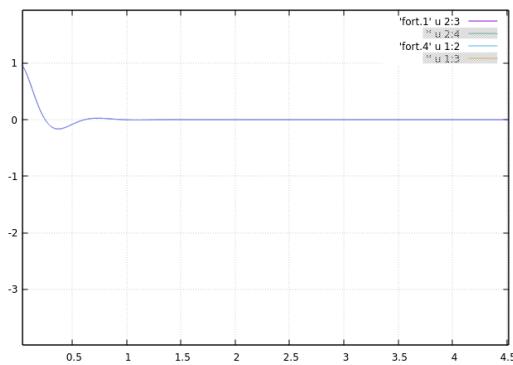
(b) energia potenziale in funzione del tempo



(c) energia meccanica in funzione del tempo

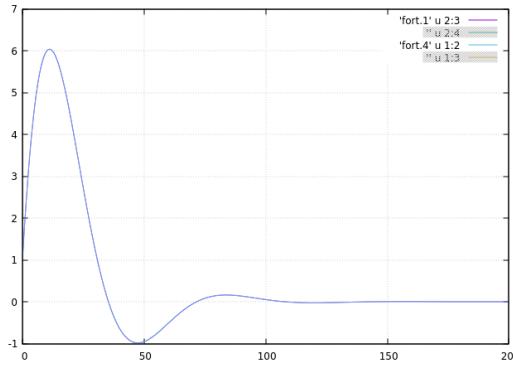
### 3.1 Comportamento del sistema se m o k sono troppo grandi

Analizziamo prima il caso in cui **massa: 0.1** e **kappa: 10**. Secondo la teoria, la massa avrà difficoltà a mettersi in moto, in quanto la costante elastica, la quale indica con che facilità essa si può allungare o comprimere, è troppo grande rispetto alla massa stessa. Ci si aspetta quindi che il sistema non oscilli secondo la norma. Graficando ancora una volta la posizione rispetto al tempo, si ottiene il seguente grafico:



dal quale si evince che la massa riesce a fare una singola oscillazione, che viene subito smorzata dalla forza di attrito.

Nel caso in cui **massa: 10** e **kappa: 0.1**, si ottiene invece il seguente risultato:



in questo caso, essendo la costante elastica troppo piccola, la prima oscillazione della massa ha ampiezza maggiore della norma (proprio perché la molla è più facile da comprimere o allungare). Tuttavia anche qui l'oscillazione viene smorzata quasi subito.

### 3.2 Diversi tipi di smorzamento: la costante c

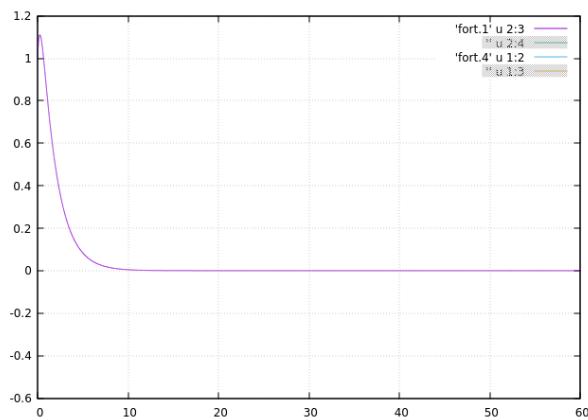
Dal procedimento di risoluzione analitico dell'equazione differenziale (2), si giunge a dover distinguere *tre diversi casi di smorzamento*, corrispondenti ai tre diversi casi del  $\Delta$  dell'equazione caratteristica ( $\Delta > 0, < 0, = 0$ ):

**1. Moto sottosmorzato:**  $c < 4mk$ :

In tutti i tentativi precedentemente analizzati è stato fatto uso della costante di attrito lineare  $c = 1$ ; questo valore soddisfa quindi la condizione del moto oscillatorio sottosmorzato.

**2. Moto sovrasmorzato:**  $c > 4mk$ :

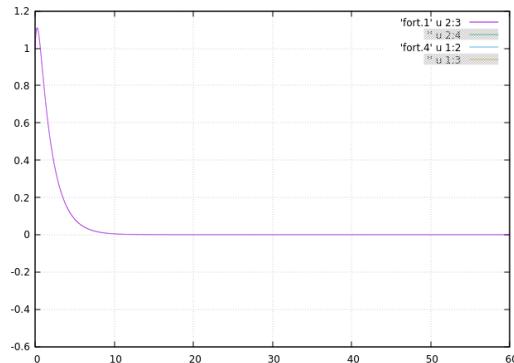
In questo caso è stato scelto un valore della costante di attrito lineare pari a **c: 20**, con **massa: 5** e **kappa: 10**. Il risultato è il seguente:



ossia il sistema **non oscilla** e tende direttamente verso la posizione di equilibrio.

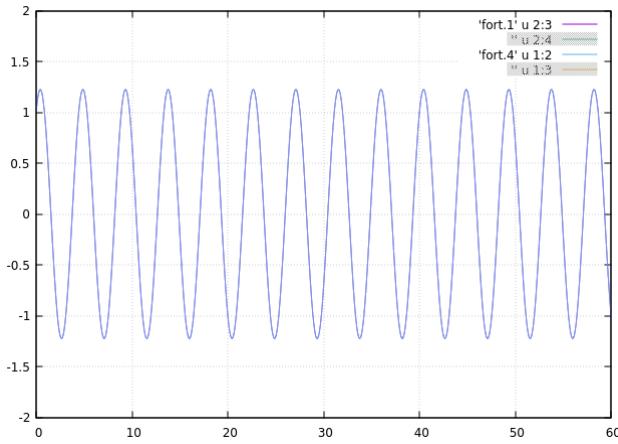
**3. Smorzamento critico:**  $c = 4mk$ :

Scegliendo un valore pari a **c: 14.4**  $\approx \sqrt{200}$ , viene soddisfatta la condizione per lo smorzamento critico, nel quale ancora una volta non si ha nessuna oscillazione, ma la curva tende a 0 più lentamente del caso precedente:

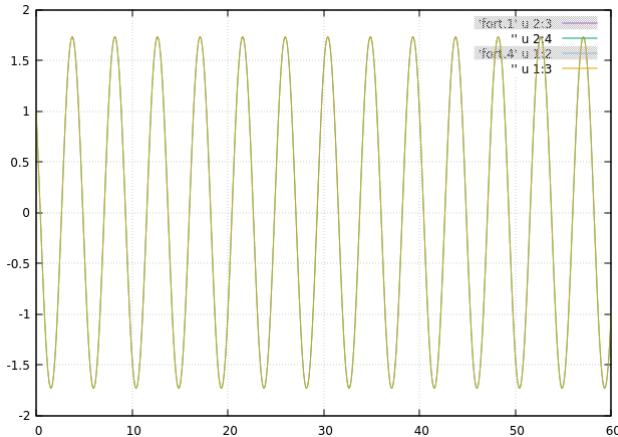


### 3.3 Oscillatore non smorzato

Un caso particolarmente interessante è quello con  $c=0$ . Ciò si verifica nel momento in cui la forza di attrito lineare agente sul sistema è nulla: esso si muoverà quindi di moto oscillatorio non smorzato, senza fermarsi:



(a) posizione in funzione del tempo



(b) velocità in funzione del tempo

Il sistema preso ora in considerazione sarà quindi un sistema **conservativo**. Si può quindi verificare la **conservazione dell'energia meccanica** (rosso: energia cinetica; verde: energia potenziale; blu: energia meccanica totale):

