

Il problema del pendolo

Gianluca Covini, # Matricola 481021, Università di Pavia
Relazione del corso di Modellistica Numerica

9 gennaio 2026

Il problema che vogliamo risolvere è un problema al bordo con condizioni di Dirichlet che modellizzi il comportamento di un pendolo ottenuto fissando una massa m a un'asta di lunghezza L . In particolare, la funzione incognita sarà $\theta(t)$, che indica l'angolo che l'asta forma con la verticale.

Tramite la legge di Newton, otteniamo $\theta''(t) = -\sin \theta(t)$, dove abbiamo assunto per semplicità $\frac{g}{L} = 1$.

Assumiamo, poi, che il pendolo parta al tempo $t = 0$, dal punto $\frac{\pi}{3}$ e torni nella stessa posizione dopo un tempo 2π

Il problema al bordo che andiamo a risolvere, quindi sarà:

$$\begin{cases} \theta'' = -\sin \theta \\ \theta(0) = \frac{\pi}{3}, \quad \theta(2\pi) = \frac{\pi}{3} \end{cases}$$

Affrontiamo il problema numericamente con due diversi metodi: il metodo di shooting con bisezione e il metodo di shooting con Newton.

1 Metodo di shooting con bisezione

Come primo approccio al problema, usiamo il metodo di shooting combinato con il metodo di bisezione. L'idea dei metodi di shooting è di ridurre il problema al bordo a un problema ai valori iniziali tramite un parametro s che indica il valore della derivata della funzione al tempo iniziale. Sono procedimenti iterativi in cui il parametro s viene affinato man mano, tramite un metodo di ricerca degli zeri. In questo primo esempio il metodo di ricerca degli zeri usato è il metodo di bisezione.

Scriviamo di seguito la funzione che inizializza il metodo:

```
1 function [t, y] = shooting_bisezione(OdeF, s0, s1, a, b, alpha,
   beta, eps, maxit)
2 F = @(t, u) [u(2); OdeF(t, u(1), u(2))];
3 [~, y0] = ode45(F, [a, b], [alpha; s0]);
4 phi_s0 = y0(end, 1) - beta;
5 [~, y1] = ode45(F, [a, b], [alpha; s1]);
6 phi_s1 = y1(end, 1) - beta;
7 it = 0;
8
9 while phi_s0*phi_s1 < 0 && abs(phi_s0) > eps && it < maxit
10     Sm = (s0+s1)/2;
11     [tm, ym] = ode45(F, [a, b], [alpha; Sm]);
12     phi_sm = ym(end, 1) - beta;
13     plot(tm, ym(:, 1))
14     hold on
15
```

```

16     if phi_s0*phi_sm < 0
17         s1 = Sm;
18         phi_s1 = phi_sm;
19     else
20         s0 = Sm;
21         phi_s0 = phi_sm;
22     end
23
24     it = it+1;
25 end
26
27 xlabel('t')
28 ylabel('teta(t)')
29 title('Shooting + Bisezione')
30
31 t = tm;
32 y = ym(:, 1);
33
34 end

```

Lo applichiamo, poi, al nostro problema tramite il seguente comando, prendendo come scelte iniziali per il metodo di bisezione $s_0 = -1$ e $s_1 = 1$

```

1  [~, ~] = shooting_bisezione(@(t, y, dydt) -sin(y(1)), -1, 1, 0,
    2*pi, pi/3, pi/3, 1e-7, 1e3);

```

Il plot ottenuto è quello in figura 1.

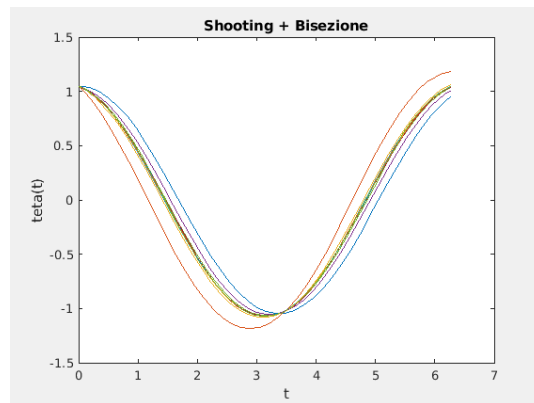


Figura 1: Metodo di shooting con bisezione

2 Metodo di shooting con Newton

Come secondo metodo, combiniamo il metodo di shooting con quello di Newton per la ricerca degli zeri. Come scelta iniziale per il metodo di Newton prendiamo $s = 0$

Il codice applicato al nostro problema è il seguente:

```

1  tol = 1e-7;
2  maxit = 1e3;
3
4  % y = [U, U', Us, Us']

```

```

5 % F = y' = [U', U'', Us', Us'']
6
7 F = @(t, y) [y(2); -sin(y(1)); y(4); -cos(y(1))*y(3)];
8
9 a = 0;
10 b = 2*pi;
11
12 alpha = pi/3;
13 beta = pi/3;
14
15 s0 = 1.7;
16
17 s = s0;
18 k = 0;
19 [t, y] = ode45(F, [a, b], [alpha; s; 0; 1]);
20 phi = y(end, 1) - beta;
21 phi_d = y(end, 3) - beta;
22
23 while abs(phi) > tol && k < maxit
24     s = s-phi/phi_d;
25     [t, y] = ode45(F, [a, b], [alpha; s; 0; 1]);
26     phi = y(end, 1);
27     phi_d = y(end, 3);
28     k = k+1;
29     plot(t, y(:, 1))
30     hold on
31 end
32
33 xlabel('t')
34 ylabel('teta(t)')
35 title('Shooting + Newton')
36 legend(num2str((0:k)'), 'Location', 'Best')

```

Il plot generato è contenuto in figura 2.

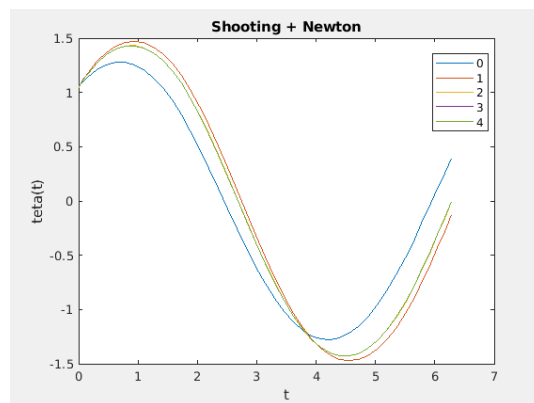


Figura 2: Metodo di shooting con Newton con $s_0 = 1$

Notiamo subito che il metodo di Newton converge in poche iterazioni (4 nel caso specifico), come ci si poteva aspettare. In particolare, converge più velocemente del metodo di bisezione che, nel nostro caso, impiega 19 iterazioni per raggiungere l'accuratezza desiderata.

Un'altra osservazione interessante è notare cosa succede se cambiamo la scelta del valore iniziale di s . Infatti, modificando tale parametro, i metodi convergono a una soluzione differente da quella ottenuta con i valori presi precedentemente.

Ad esempio, proviamo a considerare il metodo di Newton con scelta iniziale $s_0 = 1.7$. Il risultato è riportato in figura 3.

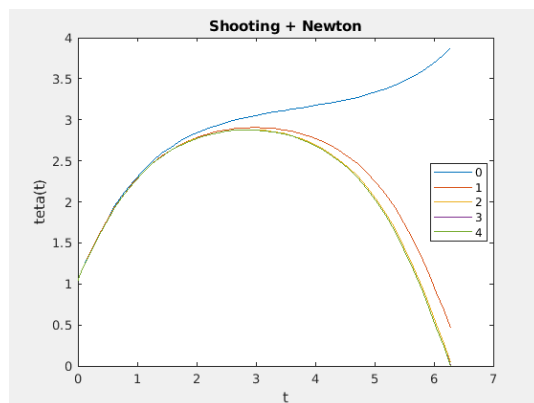


Figura 3: Metodo di shooting con Newton con scelta iniziale $s_0 = 1.7$

Di fatto, quelle ottenute sono altre soluzioni del problema al bordo. Dal punto di vista fisico, ciò si spiega con il fatto che le diverse soluzioni corrispondono a diversi giri del pendolo attorno al chiodo. Il pendolo, cioè, può ritornare al punto iniziale subito, o dopo aver compiuto attorno al chiodo un giro completo, due, tre...

Ci aspettiamo, quindi, che il problema al bordo abbia infinite soluzioni corrispondenti ciascuna a un numero diverso di giri completi compiuti dal pendolo attorno al chiodo.