

Conducibilità termica

Gruppo di lavoro A-1-3-2

Marco RICCI (m. 641084) m.ricci66@studenti.unipi.it
Martino BARBIERI (m. 640522) m.barbieri20@studenti.unipi.it

UniPi - Laboratorio di Fisica 1 - a.a. 2021-2022
Docente: Prof. Luca Baldini

1 Introduzione

Esistono diversi meccanismi di trasmissione del calore: conduzione, convezione, irraggiamento. In questa esperienza ci interessiamo al primo dei tre.

Gli atomi che compongono la materia non sono fermi, bensì vibrano e la temperatura rappresenta una misura dello stato di agitazione delle particelle. Maggiore è l'ampiezza delle vibrazioni di un atomo e maggiore è l'energia termica che esso possiede. Il trasferimento di calore da una estremità all'altra della sbarra di metallo (che rappresenta il fenomeno osservato in questa esperienza) può essere visualizzato microscopicamente come urti tra atomi dotati di maggior energia contro altri che ne hanno meno, determinando uno scambio di quantità di moto e energia e portando ad una tendenza di omologazione energetica. Quindi data questa semplice descrizione, risulta intuitivo pensare che il rapporto tra il calore scambiato tra due corpi (a temperature T_H e T_C) per unità di tempo sia proporzionale alla differenza di temperatura e all'area di contatto (maggiore è l'area e maggiore è il numero di urti, così come se è maggiore il dislivello energetico sarà maggiore il calore trasferito nell'unità di tempo) e inversamente proporzionale allo spessore. Quindi con lo strumento dell'analisi dimensionale possiamo giungere ad una ipotesi di quello che potrebbe essere il *flusso di calore*.

$$W = \lambda S \frac{\Delta T}{\Delta x} \quad (1)$$

λ costituisce un parametro che ci descrive la tendenza del materiale a condurre il calore: questa è la Legge di Fourier.

Se prendiamo in considerazione una sbarra conduttrice che presenti ai due estremi due bagni termici a temperature differenti, invertendo la formula appena considerata possiamo trovare la teorica relazione tra temperatura e distanza dall'estremo "caldo".

$$T_i = T_0 - \frac{W}{\lambda S} x_i \quad (2)$$

1.1 Obiettivo

Il fine di questa esperienza è comprendere se effettivamente la temperatura decresce linearmente con l'allontanarsi dall'estremità calda di un mezzo di conduzione termica. Quindi vogliamo misurare il coefficiente caratteristico del materiale di conduzione termica (λ) e confrontarlo con valori tipici di materiali noti. Dunque vogliamo comprendere in quali condizioni la coincidenza tra modello e realtà è maggiore ed infine ipotizzare le ragioni per cui vi sono delle discrepanze.

1.2 Materiale utilizzato

1. Due barre di materiali diversi;
2. due termistori per la misura della temperatura;
3. calcolatore con programma di acquisizione;

4. un alimentatore chiuso su due resistenza;
5. un circuito di acqua corrente;
6. un metro a nastro (± 1 mm).

2 Raccolta dati

In questa sezione descriveremo brevemente gli accorgimenti che abbiamo utilizzato per ridurre potenziali errori sistematici e raccogliere misure in una situazione il più possibile prossima a quella ideale, dove abbiamo ricavato analiticamente l'andamento della temperatura.

Sul sito <https://unilab.mbarbieri.it> rendiamo disponibile tutto il materiale raccolto ed elaborato. Lasciamo nell'Appendice A il codice Python utilizzato per l'analisi dei dati.

Sulle sbarre dei due diversi metalli misuriamo le posizioni dei due estremi di ogni foro. Per ogni posizione sulle sbarra dobbiamo misurare la temperatura del metallo. Evidenziamo che le misure delle posizioni sono state fatte dall'estremo delle sbarre a temperatura inferiore e di conseguenza il coefficiente angolare della retta che tentiamo di verificare sarà di segno opposto rispetto a quello descritto nella scheda di laboratorio.

Misure sbarra di alluminio		
n foro	estremo 1° ± 0.5 mm	estremo 2° ± 0.5 mm
1	68.5	73.0
2	93.5	98.0
3	118.5	123.0
4	143.5	148.0
5	168.5	173.0
6	193.0	197.5
7	218.5	223.0
8	243.0	247.5
9	267.8	272.2
10	292.5	297.5
11	317.5	322.0
12	342.2	346.8
13	367.0	372.0
14	393.0	397.0
15	417.5	422.0

Misure sbarra di rame		
n foro	estremo 1° ± 0.5 mm	estremo 2° ± 0.5 mm
1	67.0	71.5
2	88.0	92.5
3	109.5	113.5
4	131.0	135.2
5	152.0	157.0
6	173.5	178.0
7	194.5	199.5
8	216.2	220.8
9	237.5	242.0
10	259.0	263.2
11	280.0	285.0
12	301.5	306.0
13	323.0	327.2
14	344.0	349.0
15	365.2	370.0
16	386.5	391.0
17	408.0	412.5
18	429.0	433.5
19	450.5	455.0
20	472.0	476.5

Per compiere una buona misura è necessario che la temperatura misurata dal termistore sia stabile e che quindi il sistema termalizzi. D'altra parte, è altrettanto necessario sottolineare il fatto che il compiere l'esperienza lentamente significa andare ad effettuare le misure in situazioni ambientali mutevoli e quindi non ottimali; infatti tali condizioni potrebbero dare origine ad errori non sistematici e di conseguenza difficili da quantificare e correggere. Per tale ragione tentiamo di capire quali sia il tempo minimo affinché avvenga la termalizzazione, compiendo una serie di misure di perlustrazione preliminari. Identifichiamo 25/30 secondi come il tempo minimo per ottenere la stabilizzazione e per tale ragione lasciamo la sonda all'interno delle cavità per un periodo di 30 secondi. Quindi prendiamo 10 campionamenti della temperatura per ogni foro, i quali sono stati effettuati dal termistore negli ultimi 5 secondi del periodo di 30, e tramite un programma in python facciamo la media della temperatura misurata. Data la stabilità della misura della temperatura (che comunque è soggetta a leggere variazioni nonostante la termalizzazione) poniamo l'incertezza ad un ragionevole doppio della risoluzione del termistore.

n foro	sbarra di alluminio	sbarra di rame
	temperature ± 0.2 °C	
1	21.2	20.0
2	21.7	21.4
3	22.3	22.8
4	22.4	24.1
5	22.8	25.4
6	23.3	26.5
7	24.2	27.9
8	25.1	29.0
9	25.4	30.7
10	26.4	32.2
11	27.3	33.5
12	27.9	35.0
13	28.6	36.4
14	29.3	38.1
15	29.9	39.6
16	30.7	
17	31.6	
18	32.1	
19	33.0	
20	34.0	

Inoltre il valore della corrente erogata dall'alimentatore è di (1.65 ± 0.01) A mentre il voltaggio è (10.3 ± 0.1) V.

Il diametro della sbarra di alluminio è (25.05 ± 0.05) mm e quello di rame (24.95 ± 0.05) mm.

3 Analisi dati

Come prima cosa ci calcoliamo la posizione dei fori sulle sbarre, facendo la media delle posizioni degli estremi. L'incertezza su queste posizioni la otteniamo propagando adeguatamente le incertezze. (0.35 mm)

n foro	sbarra di alluminio	sbarra di rame
	posizione ± 0.4 mm	
1	69.3	70.8
2	90.3	95.8
3	111.5	120.8
4	133.1	145.8
5	154.5	170.8
6	175.8	195.3
7	197.0	220.8
8	218.5	245.3
9	239.8	270.0
10	261.1	295.0
11	282.5	319.8
12	303.8	344.5
13	325.1	369.5
14	346.5	395.0
15	367.6	419.8
16	388.8	
17	410.3	
18	431.3	
19	452.8	
20	474.3	

Disegniamo un grafico che abbia in ascissa le posizioni dei fori ed in ordinata le temperature corrispondenti, al fine di verificare che effettivamente esista una dipendenza lineare tra le due grandezze.

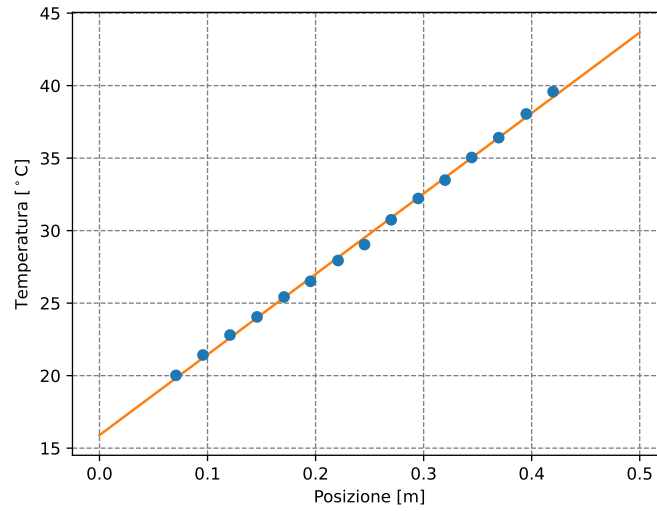


Figura 1: grafico alluminio

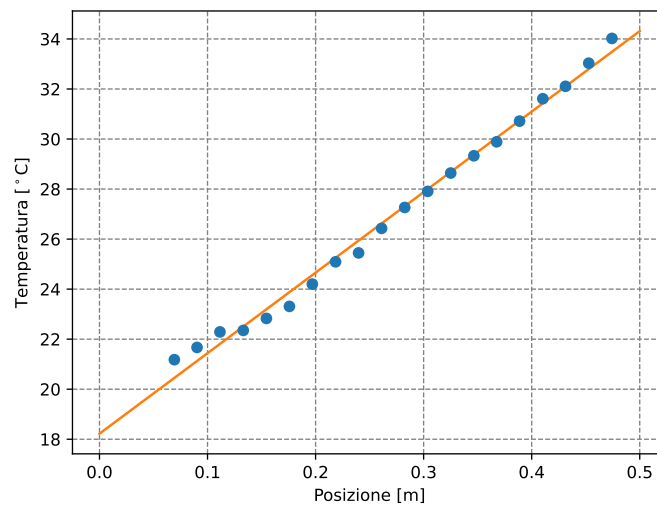


Figura 2: grafico rame

I valori che meglio approssimano il coefficiente angolare e dell'intercetta con l'asse dell'ordinata delle due rette sono:

rame: $m = 32.19 \pm 0.66 \text{ } ^\circ\text{C/m}$ $q = 18.22 \pm 0.20 \text{ } ^\circ\text{C}$ (Fig. 2)

alluminio: $m = 55.53 \pm 0.56 \text{ } ^\circ\text{C/m}$ $q = 15.90 \pm 0.15 \text{ } ^\circ\text{C}$ (Fig. 1)

Questi risultati sono stati ottenuti utilizzando un programma in python finalizzato a trovare i valori che meglio approssimino la retta di *bestfit*. Riportiamo inoltre i grafici dei resti. Cioè le differenze tra le temperature misurate sperimentalmente ed i valori in corrispondenze delle loro posizioni sulla sbarra, ma definite dalla retta di miglior approssimazione.

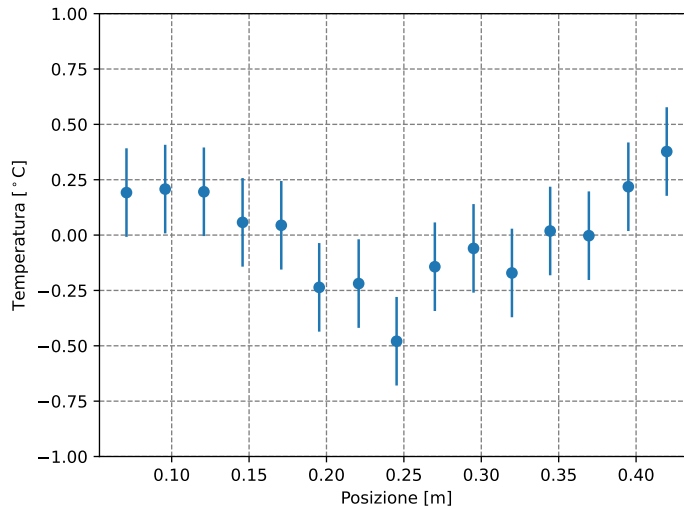


Figura 3: grafico dei resti relativi all'alluminio

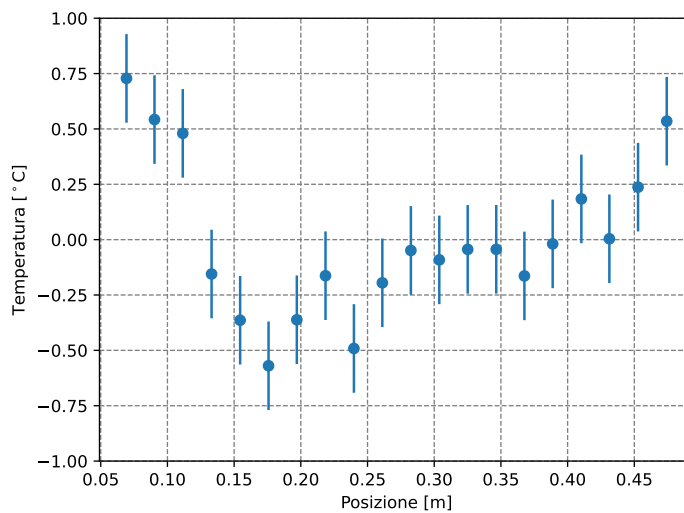


Figura 4: grafico dei resti relativi al rame

Inoltre calcoliamo anche gli intervalli di confidenza $1 - \beta$, i quali sono entrambi prossimi alla soglia del 60%.

materiali	$1 - \beta$
alluminio	60.0%
rame	55.0%

A questo punto abbiamo una stima della grandezza $W/\lambda S$ e conoscendo i valori di W e S possiamo valutare il valore del coefficiente di conducibilità termico con la relativa incertezza.

$$S = \left(\frac{d}{2}\right)^2 \pi \quad (3)$$

$$\sigma_S = \frac{d}{2} \pi \sigma_d \quad (4)$$

Dunque $S_a = (0.491 \pm 0.002)$ mm mentre $S_r = (0.489 \pm 0.002)$ mm

$$\lambda = \frac{VI}{2mS} \quad (5)$$

dove m è il coefficiente angolare

$$\sigma_\lambda^2 = \left(\frac{V}{2mS}\right)^2 \sigma_I^2 + \left(\frac{I}{2mS}\right)^2 \sigma_V^2 + \left(\frac{IV}{-2Sm^2}\right)^2 \sigma_m^2 + \left(\frac{IV}{-2mS^2}\right)^2 \sigma_S^2 \quad (6)$$

Applicando queste due formule nei due casi otteniamo:

rame: $\lambda_r = (540 \pm 13)$ W/(m°C)

alluminio: $\lambda_a = (312 \pm 5)$ W/(m°C)

4 Conclusioni

Abbiamo verificato che effettivamente sussiste una dipendenza lineare tra la temperatura e la distanza dall'estremo a contatto con la sorgente di calore a temperatura maggiore. Abbiamo trovato dei valori del coefficiente di conducibilità termica confrontabile con i valori tabulati di metalli noti:

Alluminio $\lambda = 235$ W m⁻¹ K⁻¹

Rame $\lambda = 401$ W m⁻¹ K⁻¹

Ottone $\lambda = 110$ W m⁻¹ K⁻¹

D'altra parte notiamo una certa discrepanza tra i valori tabulati dei metalli che inizialmente pensavamo componessero le sbarre (cioè alluminio e rame) ed i valori ottenuti sperimentalmente. Una possibile spiegazione per questa differenza è il fatto che le sbarre potrebbero essere realizzate in materiali misti, leghe, oppure che l'apparato di misura, composto principalmente da due termocoppie, non sia sufficientemente accurato.

A Appendice: Codici Python utilizzati

```

1 import numpy as np
2 from matplotlib import pyplot as plt
3 from scipy.optimize import curve_fit
4
5
6 file_path = 'pseudoacciaio.txt' # FILE DI TESTO CON I DATI SULLE TEMPERATURE PER
  ALLUMINIO
7 x, y, z, w= np.loadtxt(file_path, skiprows=4, unpack=True)
8
9
10 ta = np.array([])
11
12 for i in range (1,16):
13     s=0
14     for j in range (0,10):
15         s=s+y[60*i-10+j]
16
17     ta= np.append(ta, s/10)
18 print(ta)
19
20
21 file_path = 'pseudorame.txt' # FILE DI TESTO CON I DATI SULLE TEMPERATURE PER IL RAME
22 x1, y1, z1, w1= np.loadtxt(file_path, skiprows=4, unpack=True)
23
24
25 tr = np.array([])
26
27 for i in range (1,21):
28     s=0
29     for j in range (0,10):
30         s=s+y1[60*i-10+j]
31
32     tr = np.append(tr, s/10)
33
34 print(tr)
35
36 T=tr #VARIABILE DA CAMBIARE TRA TR E TA PER PASSARE DALL'ANALISI DELL'ACCIAIO E IL RAME
37
38 #POSIZIONI RELATIVE ALLA BARRA DI RAME
39 xr= np.array([6.93, 9.03, 11.15, 13.31, 15.45, 17.58, 19.70, 21.85, 23.98, 26.11,
40             28.25, 30.38, 32.51, 34.65, 36.76, 38.88, 41.03, 43.13, 45.28, 47.43])
41 xr= xr/100 #PORTIAMO LE MISURE IN METRI
42
43 #POSIZIONI RELATIVE ALLA BARRA DI ALLUMINIO
44 xa= np.array ([7.08, 9.58, 12.08, 14.58, 17.08, 19.53, 22.08, 24.53, 27.00, 29.50,
45             31.98, 34.45, 36.95, 39.50, 41.98])
46 xa=xa/100
47 x=xr #VARIABILE DA CAMBIARE TRA XA E XR PER PASSARE DELL'ANALISI DELL'ACCIAIO E IL RAME
48
49 sigma_x = np.full(x.shape, 0.0004)
50 sigma_T = np.full(T.shape, 0.2)
51
52 def line(x, m, q):
53     """Modello di fit lineare.
54     """
55     return m*x+q
56
57 plt.figure('Grafico posizione-temperatura')
58 # Grafico dei punti sperimentali.
59 plt.errorbar(x, T, sigma_T, sigma_x, fmt='o')
60 # Fit con una retta.
61
62 popt, pcov = curve_fit(line, x, T, sigma=sigma_T)
63 m_hat, q_hat = popt
64 sigma_m, sigma_q = np.sqrt(pcov.diagonal())
65 print(m_hat, sigma_m, q_hat, sigma_q)

```



```

66 # Grafico del modello di best fit.
67
68 x = np.linspace(0., 0.5, 100)
69 plt.plot(x, line(x, m_hat, q_hat))
70 # Formattazione del grafico.
71 plt.xlabel('Posizione [m]')
72 plt.ylabel('Temperatura [ $^{\circ}\text{C}$ ]')
73 plt.grid(which='both', ls='dashed', color='gray')
74
75 plt.savefig('posizione_temperatura-FINALE.pdf')
76 print(m_hat)
77 print(q_hat)
78 print(sigma_m)
79
80 plt.show()

```

Codice 1: Trova i parametri di best fit

```

1 import numpy as np
2 from matplotlib import pyplot as plt
3 from scipy.optimize import curve_fit
4
5
6
7 file_path = 'pseudoacciaio.txt' # FILE DI TESTO CON I DATI SULLE TEMPERATURE PER
8 ALLUMINIO
9 x, y, z, w = np.loadtxt(file_path, skiprows=4, unpack=True)
10
11 ta = np.array([])
12
13 for i in range (1,16):
14     s=0
15     for j in range (0,10):
16         s=s+y[60*i-10+j]
17
18     ta= np.append(ta, s/10)
19 print(ta)
20
21
22 file_path = 'pseudorame.txt' # FILE DI TESTO CON I DATI SULLE TEMPERATURE PER IL RAME
23 x1, y1, z1, w1 = np.loadtxt(file_path, skiprows=4, unpack=True)
24
25
26 tr = np.array([])
27
28 for i in range (1,21):
29     s=0
30     for j in range (0,10):
31         s=s+y1[60*i-10+j]
32
33     tr = np.append(tr, s/10)
34
35 print(tr)
36
37 T=ta #VARIABILE DA CAMBIARE TRA TR E TA PER PASSARE DALL'ANALISI DELL'ACCIAIO E IL RAME
38
39 #POSIZIONI RELATIVE ALLA BARRA DI RAME
40 xr= np.array([6.93, 9.03, 11.15, 13.31, 15.45, 17.58, 19.70, 21.85, 23.98, 26.11,
41 28.25, 30.38, 32.51, 34.65, 36.76, 38.88, 41.03, 43.13, 45.28, 47.43])
42
43
44 #POSIZIONI RELATIVE ALLA BARRA DI ALLUMINIO
45 xa= np.array ([7.08, 9.58, 12.08, 14.58, 17.08, 19.53, 22.08, 24.53, 27.00, 29.50,
46 31.98, 34.45, 36.95, 39.50, 41.98])
47
48 xa=xa/100
49
50 x=xa #VARIABILE DA CAMBIARE TRA XA E XR PER PASSARE DELL'ANALISI DELL'ACCIAIO E IL RAME
51 d=x

```

```
50 sigma_x = np.full(x.shape, 0.0004)
51 sigma_T = np.full(T.shape, 0.2)
52
53 def line(x, m, q):
54     """Modello di fit lineare.
55     """
56     return m*x+q
57
58 plt.figure('Grafico resti')
59 # Grafico dei punti sperimentali.
60
61 popt, pcov = curve_fit(line, x, T, sigma=sigma_T)
62 m_hat, q_hat = popt
63 sigma_m, sigma_q = np.sqrt(pcov.diagonal())
64 print(m_hat, sigma_m, q_hat, sigma_q)
65 #RISULTATI BEST_FIT
66
67 x = np.linspace(0., 0.5, 100)
68 res= T - line(d, m_hat, q_hat)
69 plt.errorbar(d, res, sigma_T, fmt='o')
70 # Formattazione del grafico.
71 plt.xlabel('Posizione [m]')
72 plt.ylim(-1,1)
73 plt.ylabel('Temperatura [^\circ C]')
74 plt.grid(which='both', ls='dashed', color='gray')
75
76
77 #CALCOLO DEL CONFIDENCE LEVEL
78 c=0
79 for i in range(len(res)):
80     if abs(res[i])<0.2:
81         c=c+1
82
83 confidence= c/len(res)*100
84 print(confidence)
85
86
87 plt.savefig('resti-ALL.pdf')
88
89 #plt.show()
```

Codice 2: Calcolo di residui e *c.l.*