
Trabajo Práctico de Laboratorio
Universidad Tecnológica Nacional
Medidas Electrónicas I
R4052 Año:2023

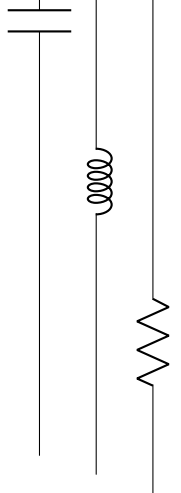
**TPL6: Tensión Bien Medida y
Corriente Bien Medida**

Profesores:

- Ing. Marinsek Emiliano
- Ing. Perdomo Juan Manuel

Integrantes:

- Borello Federico
- Dimaio Agustin
- Manoukian Francisco Tomas
- Mella Camila
- Ronchetti Juan Manuel



Objetivo

En esta practica, se buscó calibrar mediante los métodos de Tension Bien Medida y Corriente Bien Medida tres resistencias de valores de 10Ω , $10K\Omega$ y $1M\Omega$.

Además, se busco tomar mediciones necesarias para justificar experimentalmente la desviación de los errores de método, la resistencia de decisión y las cotas de las gráficas.

Instrumental empleado

Voltimetro:

- UT33C
- $3\frac{1}{2}$, 0,5% + 2d
- $R_V = 10M\Omega$

Amperimetro:

- UT61E
- $4\frac{1}{2}$, 0,5% + 2d
- $R_A = 10.7\Omega$

Inicializacion de bibliotecas

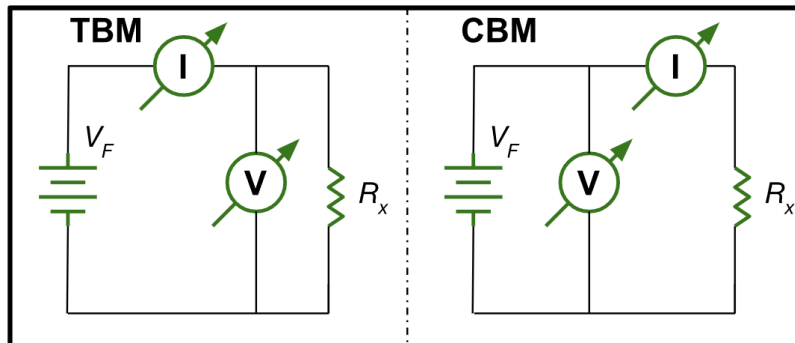
```
In [1]: # Biblioteca pandas para visualizar los datos
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import sympy as sp
import math as m

pd.set_option('display.max_rows', None)

n=3
k =2
erv = 0.5
eri = 0.5
n_er = 2
```

Introduccion teorica

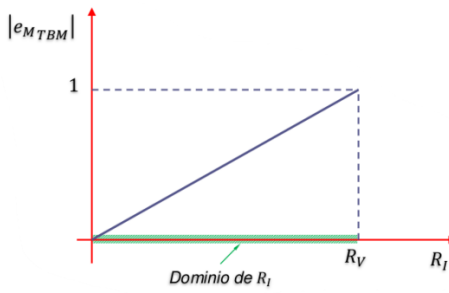
Los metodos de Tension Bien Medida (TBM) y Corriente Bien Medida (CBM) son utiles para medir el valor de una resistencia en un punto especifico, es decir, a una corriente y tension dadas. Las configuraciones circuitales de cada metodo son las siguientes:



Para ambos metodos la resistencia medida R_I resulta $\frac{V_I}{I_I}$, sin embargo debido a la topologia de cada circuito, se debe aplicar una correccion diferente para obtener la R_x real. Siendo R_V la resistencia interna del voltimetro y R_A la del amperimetro:

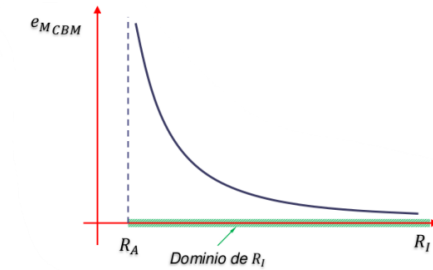
TBM

- $R_I = R_x // R_V$ por ende con este metodo se mide resistencia con defecto.
- $e_{MRT} = \frac{-R_I}{R_x}$

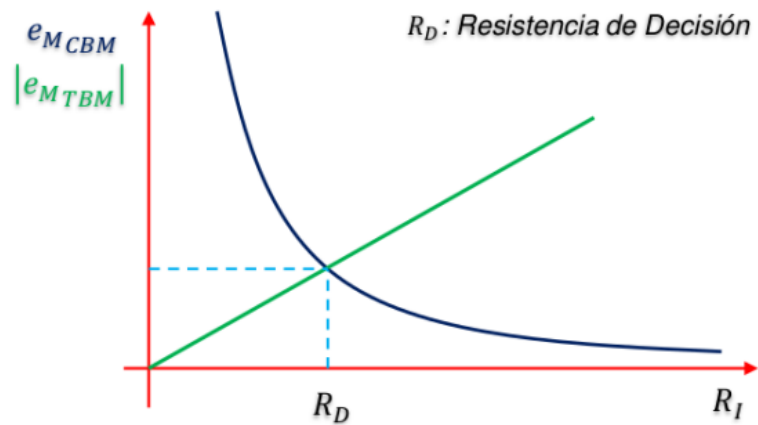


CBM

- $R_I = R_x + R_A$ por ende con este metodo se mide resistencia con exceso.
- $e_{MRC} = \frac{R_A}{R_I - R_A}$



Si se grafican los modulos de los errores de metodo en el mismo grafico, se obtiene lo siguiente:



Donde la resistencia de decision corresponde a: $R_D = \sqrt{R_A R_V}$

Desarrollo

Mediciones tomadas

En la siguiente tabla se presentan los datos obtenidos en el laboratorio, para las mediciones en TBM y CBM:

```
In [5]: datos = pd.read_excel("./Mediciones/TP6.xlsx")
```

```
datos
```

```
Out[5]:
```

	Valor Rx	Fuente	Medicion	TBM	Corriente en TBM [mA]	Valor corregido TBM	Rx TBM	Fuente.1	CBM [mA]	Tension en CBM [V]	Valor corregido CBM	Rx CBM
0	10	1,8V	1	1.000	100.990	9.901980e+00	9.901970	0,9V	48.580	0.99	9.678757e+00	2.037876e+01
1	10	1,8V	2	1.000	100.970	9.903942e+00	9.903932	0,9V	48.550	0.99	1.039135e+01	2.039135e+01
2	10	1,8V	3	1.000	100.950	9.905904e+00	9.905894	0,9V	48.560	0.99	1.038715e+01	2.038715e+01
3	10k	15V	1	15.040	1.518	9.917600e+03	9907.773386	15V	1.510	14.99	9.917152e+03	9.927152e+03
4	10k	15V	2	15.030	1.519	9.904468e+03	9894.667544	15V	1.511	15.00	9.917201e+03	9.927201e+03
5	10k	15V	3	15.031	1.519	9.905127e+03	9895.325872	15V	1.511	14.99	9.910582e+03	9.920582e+03
6	1M	19V	1	18.970	0.020	1.047893e+06	948500.000000	19V	0.018	19.02	1.056657e+06	1.056667e+06
7	1M	19V	2	18.990	0.020	1.049113e+06	949500.000000	19V	0.018	19.05	1.058323e+06	1.058333e+06
8	1M	19V	3	18.980	0.019	1.109812e+06	998947.368421	19V	0.018	19.03	1.057212e+06	1.057222e+06

Calculos

```
In [2]: V_10_t = [1, 1, 1]
I_10_t = [0.100990, 0.100970, 0.100950]

Va_10_t = np.average(V_10_t)
Ia_10_t = np.average(I_10_t)
R_10_t = Va_10_t/Ia_10_t
print("R_10_t = ", R_10_t)

ua_V_10_t = ( 1/m.sqrt(n) * np.std(V_10_t) )
ua_I_10_t = ( 1/m.sqrt(n) * np.std(I_10_t) )

ubr_V_10_t = ( 1/m.sqrt(3) * Va_10_t * (erv/100 + n_er/1000) )
ubr_I_10_t = ( 1/m.sqrt(3) * Ia_10_t * (eri/100 + n_er/10097) )

ur_V_10_t = m.sqrt(ua_V_10_t**2 + ubr_V_10_t**2) / Va_10_t
ur_I_10_t = m.sqrt(ua_I_10_t**2 + ubr_I_10_t**2) / Ia_10_t

U_R_10_t = m.sqrt(ur_V_10_t**2 + ur_I_10_t**2) * R_10_t * k
print("U_R_10_t = ", U_R_10_t)

R_10_t = 9.903931860948797
U_R_10_t = 0.09972767832810293
```

```
In [4]: V_10k_t = [15.040,15.030,15.031]
I_10k_t = [0.001518, 0.001519, 0.001519]

Va_10k_t = np.average(V_10k_t)
Ia_10k_t = np.average(I_10k_t)
R_10k_t = Va_10k_t/Ia_10k_t
print("R_10k_t = ", R_10k_t)

ua_V_10k_t = ( 1/m.sqrt(n) * np.std(V_10k_t) )
ua_I_10k_t = ( 1/m.sqrt(n) * np.std(I_10k_t) )

ubr_V_10k_t = ( 1/m.sqrt(3) * Va_10k_t * (erv/100 + n_er/15036) )
ubr_I_10k_t = ( 1/m.sqrt(3) * Ia_10k_t * (eri/100 + n_er/1519) )

ur_V_10k_t = m.sqrt(ua_V_10k_t**2 + ubr_V_10k_t**2) / Va_10k_t
ur_I_10k_t = m.sqrt(ua_I_10k_t**2 + ubr_I_10k_t**2) / Ia_10k_t

U_R_10k_t = m.sqrt(ur_V_10k_t**2 + ur_I_10k_t**2) * R_10k_t * k
print("U_R_10k_t = ", U_R_10k_t)

R_10k_t = 9899.253731343286
U_R_10k_t = 93.16791377851304
```

```
In [7]: V_1M_t = [18.970,18.980,18.990]
I_1M_t = [0.000020, 0.000020, 0.000019]

Va_1M_t = np.average(V_1M_t)
Ia_1M_t = np.average(I_1M_t)
R_1M_t = Va_1M_t/Ia_1M_t
print("R_1M_t = ", R_1M_t)

ua_V_1M_t = ( 1/m.sqrt(n) * np.std(V_1M_t) )
ua_I_1M_t = ( 1/m.sqrt(n) * np.std(I_1M_t) )

ubr_V_1M_t = ( 1/m.sqrt(3) * Va_1M_t * (erv/100 + n_er/18980) )
ubr_I_1M_t = ( 1/m.sqrt(3) * Ia_1M_t * (eri/100 + n_er/200) )

ur_V_1M_t = m.sqrt(ua_V_1M_t**2 + ubr_V_1M_t**2) / Va_1M_t
ur_I_1M_t = m.sqrt(ua_I_1M_t**2 + ubr_I_1M_t**2) / Ia_1M_t

U_R_1M_t = m.sqrt(ur_V_1M_t**2 + ur_I_1M_t**2) * R_1M_t * k
print("U_R_1M_t = ", U_R_1M_t)

R_1M_t = 965084.7457627117
U_R_1M_t = 32023.70586666085
```

```
In [14]: V_10_c = [0.99, 0.99, 0.99]
I_10_c = [0.048580, 0.048550, 0.048560]

Va_10_c = np.average(V_10_c)
Ia_10_c = np.average(I_10_c)

# En este caso es necesario corregir la resistencia teniendo en
# cuenta la Ra ya que el error de metodo es demasiado alto.
R_10_c = Va_10_c/Ia_10_c - 10.7
print("R_10_c = ", R_10_c)

ua_V_10_c = ( 1/m.sqrt(n) * np.std(V_10_c) )
ua_I_10_c = ( 1/m.sqrt(n) * np.std(I_10_c) )

ubr_V_10_c = ( 1/m.sqrt(3) * Va_10_c * (erv/100 + n_er/99) )
ubr_I_10_c = ( 1/m.sqrt(3) * Ia_10_c * (eri/100 + n_er/4857) )

ur_V_10_c = m.sqrt(ua_V_10_c**2 + ubr_V_10_c**2) / Va_10_c
ur_I_10_c = m.sqrt(ua_I_10_c**2 + ubr_I_10_c**2) / Ia_10_c

U_R_10_c = m.sqrt(ur_V_10_c**2 + ur_I_10_c**2) * R_10_c * k
print("U_R_10_c = ", U_R_10_c)

R_10_c = 9.685750566270851
U_R_10_c = 0.2883025894946216
```

```
In [16]: V_10k_c = [14.99, 14.99, 15]
I_10k_c = [0.001511, 0.001511, 0.001510]

Va_10k_c = np.average(V_10k_c)
Ia_10k_c = np.average(I_10k_c)
R_10k_c = Va_10k_c/Ia_10k_c
print("R_10k_c = ", R_10k_c)

ua_V_10k_c = ( 1/m.sqrt(n) * np.std(V_10k_c) )
ua_I_10k_c = ( 1/m.sqrt(n) * np.std(I_10k_c) )

ubr_V_10k_c = ( 1/m.sqrt(3) * Va_10k_c * (erv/100 + n_er/1499) )
ubr_I_10k_c = ( 1/m.sqrt(3) * Ia_10k_c * (eri/100 + n_er/1511) )

ur_V_10k_c = m.sqrt(ua_V_10k_c**2 + ubr_V_10k_c**2) / Va_10k_c
ur_I_10k_c = m.sqrt(ua_I_10k_c**2 + ubr_I_10k_c**2) / Ia_10k_c

U_R_10k_c = m.sqrt(ur_V_10k_c**2 + ur_I_10k_c**2) * R_10k_c * k
print("U_R_10k_c = ", U_R_10k_c)

R_10k_c = 9924.977934686674
U_R_10k_c = 102.70113615571081
```

```
In [17]: V_1M_c = [19.02,19.03,19.05]
I_1M_c = [0.000018, 0.000018, 0.000018]

Va_1M_c = np.average(V_1M_c)
Ia_1M_c = np.average(I_1M_c)
R_1M_c = Va_1M_c/Ia_1M_c
print("R_1M_c = ", R_1M_c)

ua_V_1M_c = ( 1/m.sqrt(n) * np.std(V_1M_c) )
ua_I_1M_c = ( 1/m.sqrt(n) * np.std(I_1M_c) )

ubr_V_1M_c = ( 1/m.sqrt(3) * Va_1M_c * (erv/100 + n_er/1905) )
ubr_I_1M_c = ( 1/m.sqrt(3) * Ia_1M_c * (eri/100 + n_er/180) )

ur_V_1M_c = m.sqrt(ua_V_1M_c**2 + ubr_V_1M_c**2) / Va_1M_c
ur_I_1M_c = m.sqrt(ua_I_1M_c**2 + ubr_I_1M_c**2) / Ia_1M_c

U_R_1M_c = m.sqrt(ur_V_1M_c**2 + ur_I_1M_c**2) * R_1M_c * k
print("U_R_1M_c = ", U_R_1M_c)

R_1M_c = 1057407.4074074072
U_R_1M_c = 21027.90360606089
```

Resultados

TBM

$$R_{10-TBM} = (9, 90 \pm 0, 10)\Omega \quad k_{95\%} = 2$$

$$R_{10k-TBM} = (9900 \pm 93)\Omega \quad k_{95\%} = 2$$

$$R_{1M-TBM} = (965 \pm 32)k\Omega \quad k_{95\%} = 2$$

CBM

$$R_{10-CBM} = (9, 69 \pm 0, 29)\Omega \quad k_{95\%} = 2$$

$$R_{10k-CBM} = (9900 \pm 100)\Omega \quad k_{95\%} = 2$$

$$R_{1M-CBM} = (1057 \pm 21)k\Omega \quad k_{95\%} = 2$$

Error de Método

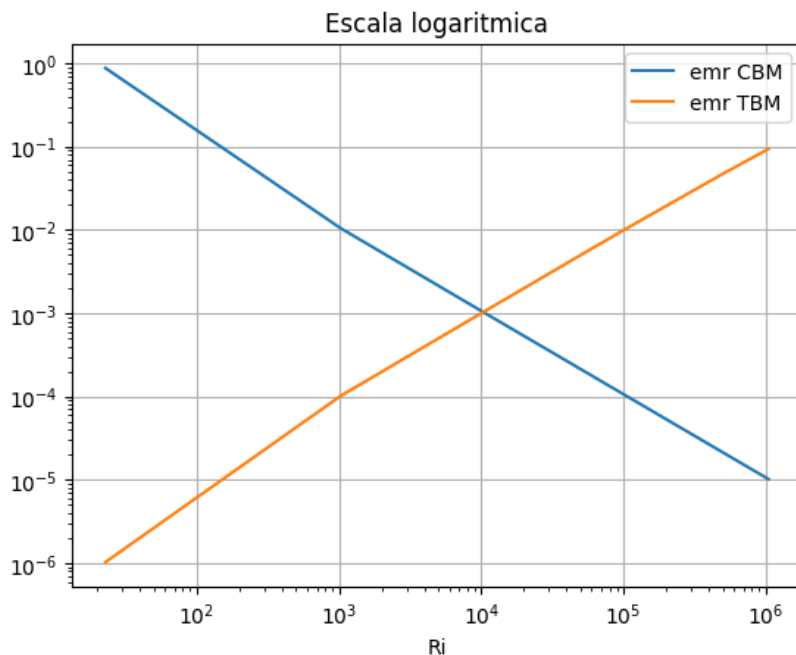
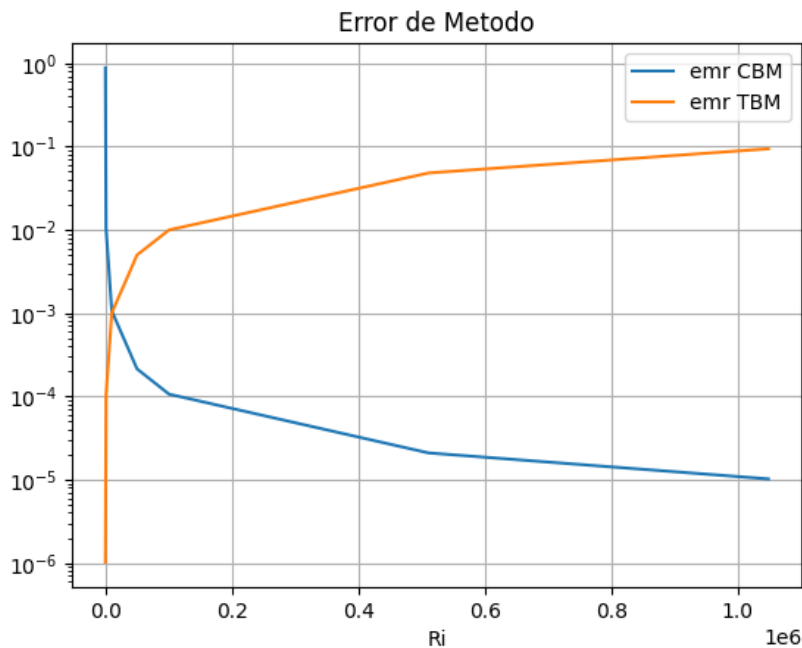
Con el objetivo de poder trazar el grafico de los errores de metodo presentado en la introduccion teorica, se tomaron las mediciones que se encuentran en la tabla a continuacion. Para ello, se empleo una resistencia por decadas, para obtener valores de error de metodo en el rango suficiente.

A continuacion se presentan las tablas y el grafico de error de metodo resultante:

```
In [22]: datos2 = pd.read_excel("./Mediciones/TP6_2.xlsx")
datos2.plot(kind='line', x='Ri',y=['emr CBM','emr TBM'], logy=True, grid=True, title="Error de Metodo")
datos2.plot(kind='line', x='Ri',y=['emr CBM','emr TBM'], logy=True, logx=True, grid=True, title="Escala loga
datos2
```

Out[22]:

	Unnamed: 0	Rdec	Tension [V]	Corriente [mA]	emr CBM	Ri	Rdec2	Tension [V].1	Corriente [mA].1	emr TBM	Ri.1
0	10R	10	0.45	19.691	0.880435	2.285308e+01	10	0.20	19.482	0.000001	10.265886
1	1k	1000	19.51	18.931	0.010491	1.030585e+03	1000	19.55	19.208	0.000102	1017.805081
2	10k	10000	19.89	1.977	0.001065	1.006070e+04	10000	19.76	1.968	0.001004	10040.650407
3	50k	50000	19.87	0.397	0.000214	5.005038e+04	50000	19.76	0.396	0.004990	49898.989899
4	100k	100000	19.94	0.198	0.000106	1.007071e+05	100000	19.76	0.198	0.009980	99797.979798
5	500k	500000	19.93	0.039	0.000021	5.110256e+05	500000	19.76	0.041	0.048195	481951.219512
6	1M	1000000	19.92	0.019	0.000010	1.048421e+06	1000000	19.76	0.021	0.094095	940952.380952



Observando el grafico del error de metodo trazado a partir de los datos experimentales, se comprueba que presenta la misma forma que el grafico presentado en la introduccion teorica.

En el segundo grafico se puede observar perfectamente como ambos graficos se intersectan en el valor de la resistencia de desicion calculado teoricamente, el cual corresponde a:

$$R_D = \sqrt{R_A R_V} = \sqrt{10.7\Omega \cdot 10M\Omega} \approx 10k\Omega$$

Conclusiones

Se concluye que se realizo exitosamente la practica planteada, calculando valores de resistencias empleando los metodos de Tension y Corriente Bien Medidas, y posteriormente se logro obtener graficamente la comparativa de errores de metodo de cada uno.

Al observar los resultados de las mediciones, se comprueba que el metodo TBM resulta mas efectivo para medir resistencias mas pequeñas, mientras que el metodo CBM es mas efectivo para resistencias mas grandes. Esto se debe a las influencias de las resistencias internas de los voltímetros y los amperímetros, y las maneras en las cuales interfieren en las medicion planteadas en la introduccion teorica. Se concluye que se obtuvo una demostracion practica del fenomeno estudiado.