

Nanoelectrónica: Ejercicios 3.9 y 3.10

Adrián D. Escañuela Copado

16 de abril de 2021

1 Ejercicio 3.9

Se pide la obtención de las curvas características intensidad-tensión y de conductancia-tensión numéricamente para un dispositivo molecular a dos terminales con las características siguientes:

$$\left\{ \begin{array}{l} \eta = 0.5 \\ E_F = -5 \text{ eV} \\ HOMO = -5.5 \text{ eV} \\ \Gamma_S = \Gamma_D = 0.1 \text{ eV} \\ T = 298 \text{ K} \\ g(E - U) = 2\delta(E - U - HOMO) \\ (i) C_{ES} \rightarrow \infty \\ (ii) q^2/C_{ES} = 1 \text{ eV} \end{array} \right. \quad (1)$$

Observamos que $\eta = 0.5$, que se identifica con unos contactos simétricos. Debemos observar esto en el gap característico del resultado.

En las fórmulas a aplicar se usa el tiempo de permanencia del electrón en la molécula, que se obtiene a partir de $\Gamma_{D/S}$ a partir de la relación siguiente.

$$\tau_{S/D} = \frac{\hbar}{\Gamma_{S/D}} \quad (2)$$

Como veremos más adelante, haremos uso de constantes expresadas en electronvoltios para no tener que preocuparnos en las conversiones de unidades de energía, por lo tanto, en la expresión anterior se usará $\hbar = 6.582119569 \times 10^{-16} \text{ eV s}$

Se utiliza extensamente la distribución de Fermi-Dirac en los problemas, así que definimos una función en nuestro problema que llamaremos cada vez que sea necesario.

```
def fermi (E:float, mu:float, T:float) ->float:
    return 1/(1+np.exp((E-mu)/(k*T)))
```

En forma de ecuación esto es:

$$f(E, \mu, T) = \frac{1}{1 + \exp\left(\frac{E - \mu}{kT}\right)} \quad (3)$$

Y definimos la constante de Boltzmann k como $k = 8.617333262 \times 10^{-5} \text{ eV/K}$

Para calcular la intensidad es necesario hacer uso de un algoritmo autoconsistente entre U y N . Dicho algoritmo aparece en la figura 1.

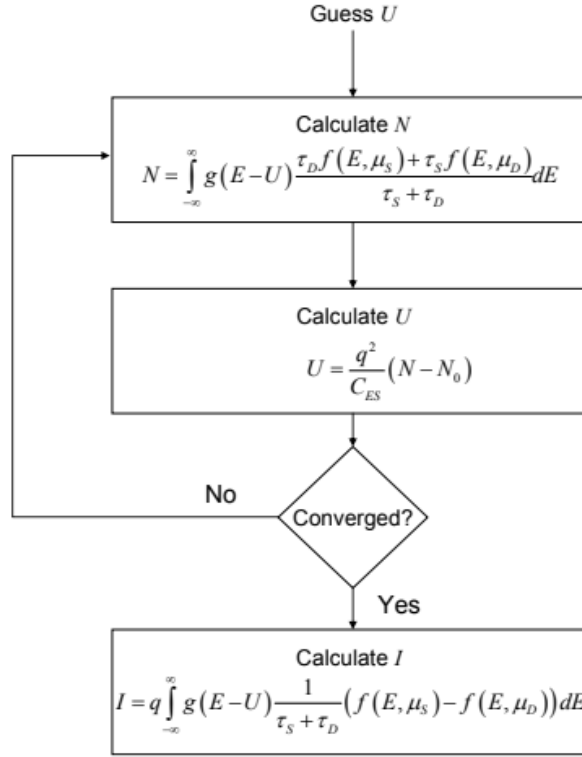


Figure 1: Algoritmo de autoconsistencia en U y N para el cálculo de I.

1.1 Caso (i): $C_{ES} \rightarrow \infty$

En este caso la energía de carga, que cuantifica el cambio en el potencial molecular por electrón transferido, es nula porque

$$U_c = \frac{q^2}{C_{ES}}(N - N_0) \quad (4)$$

y el $C_{ES} \rightarrow \infty$ del denominador hace que $U_c \rightarrow 0$.

Así que no tendremos que preocuparnos por el algoritmo de autoconsistencia en este caso, puesto que automáticamente conocemos el valor que ha de tomar U.

Como el único nivel molecular que está disponible para la conducción de electrones en este caso es el HOMO, calculamos la intensidad y la conductancia como:

```

#Arrays de valores
n=100
rangoV=2
Vds=np.linspace(-rangoV,rangoV,n)
mu_s=eta*Vds
mu_d=mu_s-Vds

#ALGORITMO
q=1.602176565e-19
#Si C_ES -> oo
U=0
I1=2*q*(fermi(U+homo-Ef,mu_s,T)
-fermi(U+homo-Ef,mu_d,T))/(tau_s+tau_d)
conduct1=np.diff(I1)/np.diff(Vds)
  
```

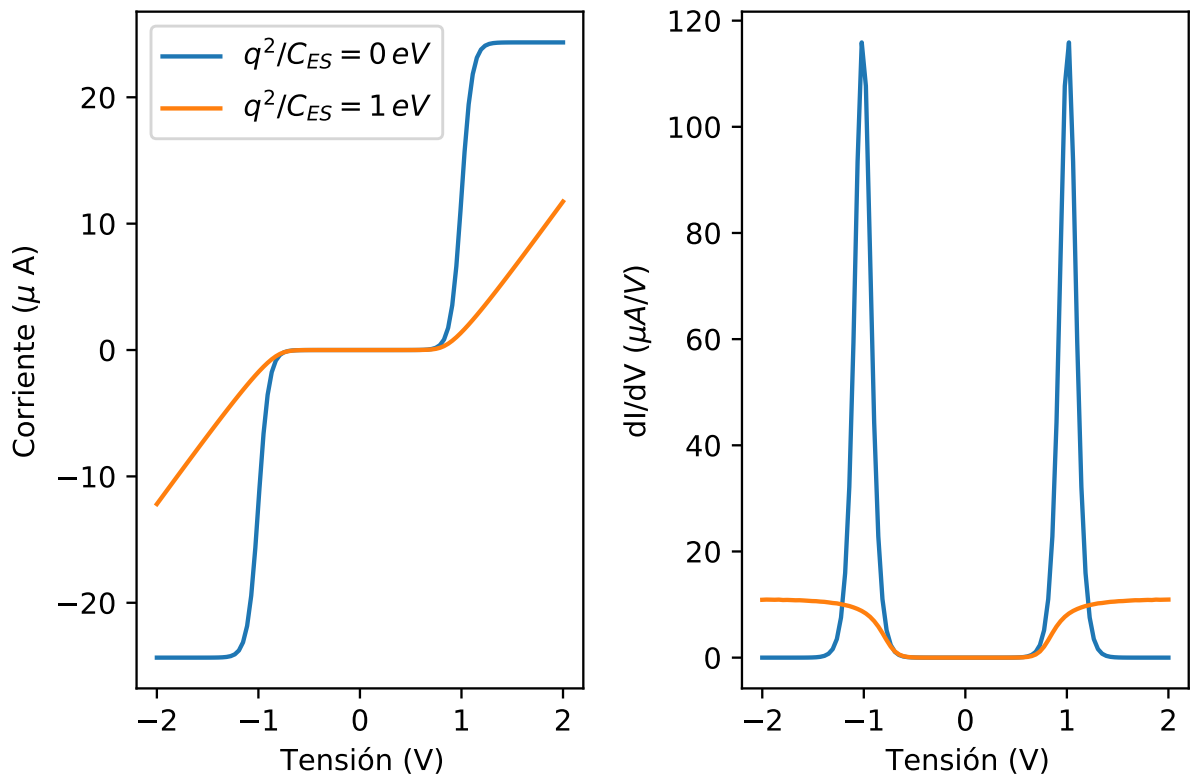


Figure 2: Corriente y conductancia en funcion de la tensión para el ejercicio 3.9.

No se evalúa la integral en el programa puesto que la densidad de estados que va como una delta de Dirac, $g(E - U) = 2\delta(E - U - \text{HOMO})$. Esto hace que el valor de E del integrando colapse en $E = U + \text{HOMO}$. Como podemos observar en el retazo de código anterior, en el lugar donde recibe la energía la función de fermi (E, μ, T) no escribimos $U + \text{HOMO}$, sino que escribimos $U + \text{HOMO} - E_f$. Esto es así porque tomamos como origen de energías el nivel de Fermi E_f , que se mantiene fijo.

Las funciones `np.diff()` son importadas de la librería `numpy` y hacen diferencia infinitesimal (depende del paso que tenga) de un array. Sería equivalente al cálculo de la derivada por diferencias finitas, que el lenguaje Python nos permite escribir de manera compacta.

Haciendo estos cálculos obtenemos lo que se puede ver en las líneas de color azul de figura 2, que es justo lo que esperábamos obtener tras comparar con lo dado en el libro de Marc Baldo. Además el gap es simétrico, como era de esperar por $\eta = 0.5$.

1.2 Caso (ii): $q^2/C_{ES} = 1 \text{ eV}$

En este caso sí que hay que evaluar U , porque ya no es nulo, por lo tanto hay que recorrer el algoritmo.

```
#Si q^2/C_ES = 1 eV
q2_entre_C_ES=1
N0=2*fermi(Ef,0,T) #=2
I2=np.linspace(-rangoV,rangoV,n) #Defino I2 como un array
                                   #de n valores

contador=0
alpha=1e-2
for Vds in np.arange(-rangoV,rangoV,2*rangoV/n):
    mu_s=eta*Vds
```

```

mu_d=mu_s-Vds
Uaux,U=0,1 #Valores para el U auxiliar y U iniciales
eps=1e-5 #Epsilon: diferencia entre U y Uaux deseada
while(abs(U-Uaux)>=eps):
    Uaux=U
    N=2*((tau_d*fermi(U+homo-Ef,mu_s,T))
    +(tau_s*fermi(U+homo-Ef,mu_d,T)))/(tau_s+tau_d)
    U=q2_entre_C_ES*(N-N0)
    U=Uaux+alpha*(U-Uaux)
#end-while
I2[contador]=2*q*(fermi(U+homo-Ef,mu_s,T)
-fermi(U+homo-Ef,mu_d,T))/(tau_s+tau_d)
contador+=1
#end-for
Vds=np.linspace(-rangoV,rangoV,n)
#Reconvierto Vds de tipo a linspace para las siguientes
#cuentas
conduct2=np.diff(I2)/np.diff(Vds)

```

Ahora utilizamos una modificación de lo que veíamos en el algoritmo de la figura 1. Esto es así porque se encontraba una solución oscilatoria, y para salir de ella usamos el truco que se nos da en el enunciado:

$$U = U_{old} + \alpha(U_{calc} - U_{old}) \quad (5)$$

Representando gráficamente esto obtenemos las curvas de color anaranjado de la figura 2.

1.3 Preguntas

- Explica el origen del gap de conductancia. ¿Qué determina su magnitud?

Los electrones de la fuente/drenador siguen la distribución de Fermi-Dirac, estando la mitad de ellos por encima de $\mu_{S/D}$ y la otra mitad por debajo.

Cuando aumentamos la tensión hacemos que los niveles de Fermi de fuente/drenador varíen según

$$\begin{cases} \mu_S = \eta q V_{DS} \\ \mu_D = \mu_S - q V_{DS} \end{cases}$$

Esto evidencia que los electrones cuya energía se sitúa alrededor de dichos niveles de Fermi cambian en energía. Cuando la tensión es positiva, aumenta la energía de la fuente, y algunos de los electrones (según la distribución) en la fuente pasarán a tener una energía superior al HOMO, pudiendo conducir si a su vez se disminuye la energía de los electrones del drenador por debajo del HOMO. Antes no podían porque las energías de los electrones de la fuente eran inferiores a la del HOMO y las del drenador superiores, a esta falta de conducción (junto con la que ocurre con el caso contrario de tensión negativa) es lo que llamamos gap en conducción.

Es por esto que la magnitud del gap de conducción lo determina cuanto de lejos queda el HOMO respecto del caso inicial (equilibrio $\mu_S = \mu_D = E_F$).

- Explica por qué la conductancia es mucho menor cuando la energía de carga es no-nula.

Cuando $U_c \neq 0$ el algoritmo autoconsistente nos dice que existe una dependencia de esta U_c con el número de electrones que se pueden transferir de los contactos a la molécula, N .

En el caso ideal, cuando $U_c = 0$, no existe dicha dependencia. La existencia de una energía de carga no-nula va en detrimento del número de electrones candidatos, ya que el efecto de carga nos dice que la adición de electrones mueve al potencial molecular para repeler a los nuevos electrones candidatos a añadirse.

Además, sabemos que esta energía se puede modelar en un equivalente circuital como un condensador, que es una impedancia. Es por estos motivos que la conductancia se ve afectada cuando tenemos $U_c \neq 0$.

2 Ejercicio 3.10

En este ejercicio se añade el nivel energético molecular $LUMO = -1.5$ eV. Esto hace que la conducción pase de ser solo posible a través del HOMO a poder ocurrir por el HOMO o por el LUMO dependiendo de la tensión aplicada.

$$\left\{ \begin{array}{l} \eta = 0.5 \\ E_F \in \{-5, -3.5, -2.5\} \text{ eV} \\ HOMO = -5.5 \text{ eV} \\ LUMO = -1.5 \text{ eV} \\ \Gamma_S = \Gamma_D = 0.1 \text{ eV} \\ T = 298 \text{ K} \\ g(E - U) = 2\delta(E - U - HOMO) \\ q^2/C_{ES} = 1 \text{ eV} \end{array} \right. \quad (6)$$

Todos los cálculos serán equivalentes a los del anterior ejercicio pero teniendo en cuenta que hay un número de electrones en el LUMO y otro en el HOMO, y por lo tanto una corriente en el LUMO y otra en el HOMO. Estos valores han de sumarse para obtener la corriente total.

Como se van a usar varios valores de E_F definimos una función que haga el algoritmo usando como variable dicha magnitud.

```
def algoritmo(n:int, rangoV:int, q2_entre_C_ES:float,
             I1:float, Ef:float)->float:

    N0=2*fermi(Ef, 0, T)
    contador=0
    alpha=1e-2
    for Vds in np.arange(-rangoV, rangoV, 2*rangoV/n):
        mu_s=eta*Vds
        mu_d=mu_s-Vds
        Uaux, U=0, 1 #Valores para el U auxiliar y U iniciales
        eps=1e-5 #Epsilon: diferencia entre U y Uaux deseada
        while(abs(U-Uaux)>=eps):
            Uaux=U
            N=2*((tau_d*fermi(U+homo-Ef, mu_s, T))
                +(tau_s*fermi(U+homo-Ef, mu_d, T))
                +(tau_d*fermi(U+lumo-Ef, mu_s, T))
                +(tau_s*fermi(U+lumo-Ef, mu_d, T)))/(tau_s+tau_d)
            U=q2_entre_C_ES*(N-N0)
            U=Uaux+alpha*(U-Uaux)
        #end-while
    I1[contador]=2*q*((fermi(U+homo-Ef, mu_s, T)
```

```

-fermi(U+homo-Ef,mu_d,T))
+(fermi(U+lumo-Ef,mu_s,T)
-fermi(U+lumo-Ef,mu_d,T)))/(tau_s+tau_d)
contador+=1

```

Como vemos, hay cambios en el valor de N , porque se evalúan tanto HOMO como LUMO y se suman. Igual ocurre con la intensidad I_1 .

Esta función la implementamos en el cuerpo del programa así:

```

#Si q^2/C_ES = 1 eV y E_f=-2.5 eV
q2_entre_C_ES=1
Ef=-2.5 #eV
I1=np.linspace(-rangoV,rangoV,n) #I como un array de n valores
algoritmo(n,rangoV,q2_entre_C_ES,I1,Ef)
conduct1=np.diff(I1)/np.diff(Vds)

#Si q^2/C_ES = 1 eV, E_F=-3.5
q2_entre_C_ES=1
Ef=-3.5 #eV
I2=np.linspace(-rangoV,rangoV,n) #I como un array de n valores
algoritmo(n,rangoV,q2_entre_C_ES,I2,Ef)
conduct2=np.diff(I2)/np.diff(Vds)

#Si q^2/C_ES = 1 eV, E_F=-5
q2_entre_C_ES=1
Ef=-5 #eV
I3=np.linspace(-rangoV,rangoV,n) #I como un array de n valores
algoritmo(n,rangoV,q2_entre_C_ES,I3,Ef)
conduct3=np.diff(I3)/np.diff(Vds)

```

Aquí asociamos I_1 con $E_F = -2.5$ eV, I_2 con $E_F = -3.5$ eV y I_3 con $E_F = -5$ eV y obtenemos lo que se ve en la figura 3.

Tras comparar con lo que aparece en el libro de Marc Baldo, vemos que los resultados son los esperados.

Además notamos que tanto en este ejercicio como el anterior los gaps son simétricos con respecto al origen, en todos los casos obtenemos funciones de simetría impar para la corriente y par para la conductancia.

2.1 Preguntas

- ¿Por qué las curvas intensidad-tensión son uniformes? Pista: ¿qué ocurriría si $\eta \neq 0.5$? Identifica el origen de las transiciones en la curva IV.

Las curvas son como así porque los contactos son simétricos, es decir, es una consecuencia de que $\eta = 0.5$.

Se define η como $\eta = \frac{C_D}{C_D + C_S}$, donde C_D y C_S son condensadores que modelan la conexión entre el QDot y los contactos. El valor $\eta = 0.5$ se da cuando la molécula tiene sus terminales equiespaciados, son simétricos, $C_D \sim C_S$. Si ocurriese que $C_S \gg C_D$ o al revés, la molécula sería antisimétrica y el gap no estaría centrado en 0. Como hemos visto antes $\mu_S = \eta q V_{DS}$, la cercanía de los niveles de Fermi a los niveles moleculares depende de η . Si $\eta = 0.5$, la distancia energética de μ_S a E_F y la que hay entre E_F y μ_D es idéntica, por lo tanto el gap estará centrado en $(I, V) = (0, 0)$. Es por esto por lo que son tan uniformes.

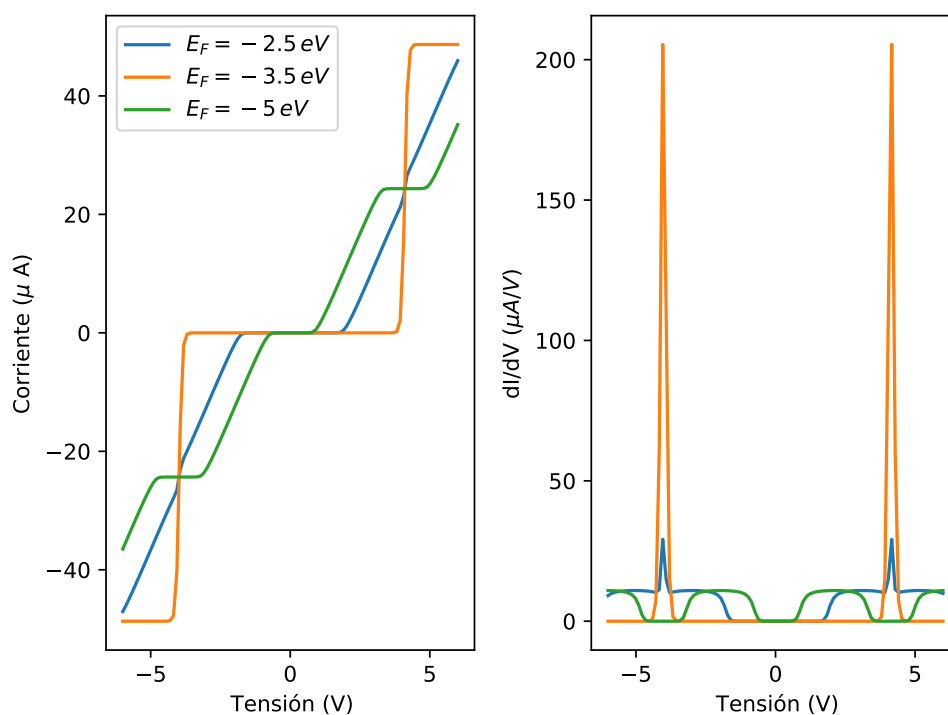


Figure 3: Corriente y conductancia en función de la tensión para el ejercicio 3.10.

En el ejercicio 3.9, viendo la figura 2, identificamos la transición en $V \sim -1, 1$ V para los dos casos.

En el ejercicio 3.10, figura 3, las transiciones para $E_F = -2.5$ eV se dan en $V \sim -2, 2$ V y $V \sim -4, 4$ V, para $E_F = -3.5$ eV se dan en $V \sim -4, 4$ V, para $E_F = -5$ eV se dan en $V \sim -1, 1$ V y en $V \sim -3, 3$ V

- **¿Por qué está ausente el efecto de carga cuando $E_F = -3.5$ eV**

Esto ocurre porque el nivel de Fermi de la molécula cae justo en la mitad, entre LUMO y HOMO. Para el HOMO $E_F - HOMO = -3.5 - (-5.5)$ eV = 2 eV y para el LUMO $LUMO - E_F = -1.5 - (-3.5)$ eV = 2 eV.

Esto potencia el crecimiento de la función intensidad en los puntos de tránsito, minimizando la pendiente causada por el efecto de carga (aunque este efecto siga estando ahí).