

Trabajo de Fin de Grado

Métodos numéricos avanzados para el estudio electrostático de la unión P-N



Departamento de Electrónica y Electromagnetismo

Grado en Física

Universidad de Sevilla

Alumna: Andrea Romero Salazar

Fecha: 06/07/2023

ÍNDICE DE CONTENIDOS:

1	INTRODUCCIÓN	1
1.1	Objetivos y metodología.....	2
1.2	Sumario de contenidos.....	3
2	UNIÓN P-N	5
2.1	Propiedades de los semiconductores	5
2.1.1	Arrastre	6
2.1.2	Difusión	6
2.1.3	Generación y recombinación	7
2.1.4	Ecuaciones de continuidad	7
2.2	Estudio electrostático de la unión pn	8
2.3	Aproximación de empobrecimiento	10
3	ANÁLISIS NUMÉRICO EN CONDICIONES DE EQUILIBRIO	12
3.1	Revisión general del método de Newton-Raphson.....	12
3.2	Aplicación a la ecuación de Poisson.....	14
3.3	Consideraciones prácticas para la implementación del método en MATLAB.....	17
3.3.1	Constantes y parámetros.....	17
3.3.2	Parámetros de simulación.....	18
3.3.3	Estimación inicial	18
3.3.4	Newton-Raphson	19
3.3.5	Representación gráfica	21
4	ANÁLISIS NUMÉRICO EN CONDICIONES DE NO-EQUILIBRIO	22
4.1	Scharfetter-Gummel	22
4.2	Ecuaciones de continuidad	24
4.3	Newton-Raphson en no-equilibrio.....	26
4.4	Consideraciones prácticas para la implementación del método en MATLAB.....	27
5	CASOS PRÁCTICOS DE APLICACIÓN: COMPARATIVA CON LA APROXIMACIÓN CLÁSICA DE EMPOBRECIMIENTO	29
5.1	Dopaje simétrico – Equilibrio.....	30
5.2	Dopaje asimétrico – Equilibrio	33
5.3	Dopaje simétrico – No-Equilibrio	35
5.4	Dopaje asimétrico – No-Equilibrio.....	38
5.5	Otros fenómenos de interés	41

5.5.1	Fotogeneración	41
5.5.2	Unión gradual	42
5.5.3	Dependencia de μ	43
6	CONCLUSIONES	44
7	REFERENCIAS	48
8	ANEXO I.....	50

1 INTRODUCCIÓN

La unión P-N es uno de los pilares esenciales de la electrónica moderna, ya que constituye la base de multitud de dispositivos semiconductores, desde diodos [1] y transistores [2] a *chips* y circuitos integrados [3]. Como estructura básica de estos dispositivos, el estudio y análisis de la unión P-N es de vital importancia a la hora de lograr una comprensión adecuada de los mismos.

Pese a su simplicidad conceptual, la unión P-N es un sistema altamente complejo cuyo análisis conlleva considerar fenómenos físicos y matemáticos tales como distribución de la corriente y difusión de portadores [4], generación y recombinación de pares electrón-hueco [5], fotogeneración [6] ... Las ecuaciones que caracterizan la unión P-N, como la ecuación de Poisson y la de continuidad de portadores, son ecuaciones diferenciales parciales no lineales que resultan complejas de resolver o, en la mayoría de los casos, carecen por completo de resolución analítica [7]. Es por esto por lo que recurrir a métodos numéricos [8]-[9] se muestra como una alternativa eficaz a la hora de enfrentarse a este problema.

Para una descripción de alta precisión se suele recurrir a aproximaciones numéricas 3-D con programas específicos para el estudio de semiconductores, TCAD (Technology Computer-Aided Design), tales como COMSOL Multiphysics [10] y Sentaurus [11]. Sin embargo, su implementación y ejecución requieren recursos computacionales significativos y un conocimiento previo sobre cómo usar correctamente cada programa. Si bien proporcionan soluciones muy precisas, el análisis de los resultados puede ser complejo y mucho menos intuitivo. Por ello se hace evidente la necesidad de una simplificación de las expresiones, a menudo en forma de aproximación mono-dimensional [12], que permita generar una mayor comprensión sobre el problema y facilitar la interpretación de los resultados.

Una posible aproximación es la de empobrecimiento, que permite el estudio de la unión P-N mediante la resolución analítica de la ecuación de Poisson en el equilibrio [13]. Esta aproximación proporciona una solución sencilla bajo condiciones de equilibrio termodinámico, pero que no es capaz de contemplar determinados efectos de interés, como aquellos dependientes de la distribución de carga libre en la zona de empobrecimiento, la aplicación de un voltaje externo, la fotogeneración, o

dependencias de las propiedades del semiconductor con las densidades de carga o gradientes de temperatura y dopado.

Para solventar las limitaciones de la aproximación de empobrecimiento, pero sin llegar a la solución de “fuerza bruta” de las herramientas TCAD, en este trabajo se presenta un estudio de métodos numéricos avanzados para el estudio de la unión P-N en equilibrio [14]-[15], y su extensión a condiciones de fuera del equilibrio [16]. Este tipo de métodos de resolución mono-dimensional de las ecuaciones de Poisson y continuidad permiten una manipulación más sencilla de los parámetros y condiciones de contorno, y facilitan la visualización e interpretación de los perfiles de carga, los campos eléctricos y las distribuciones de portadores en la unión P-N.

Este tipo de análisis permite además la incorporación de diversas características y efectos, como la recombinación y generación de portadores, la difusión, la temperatura y los perfiles de dopaje, de forma que podemos alcanzar un nivel de personalización y adaptación mayor que con aproximaciones de menor orden, sin perder intuición en la resolución del problema.

Los resultados de este estudio de métodos numéricos se han plasmado en la implementación, dentro del entorno MATLAB, de los procedimientos necesarios para la resolución numérica de las ecuaciones diferenciales que describen el comportamiento de la unión P-N. Este trabajo abarca el análisis de numerosos casos de estudio, tanto en equilibrio, donde compararemos los resultados con la aproximación de empobrecimiento, como fuera del equilibrio, donde esta aproximación es menos precisa.

1.1 OBJETIVOS Y METODOLOGÍA

El objetivo último de este trabajo será la implementación de los métodos y técnica de computación numérica estudiados dentro del entorno MATLAB [17], para caracterizar y resolver las ecuaciones asociadas a la unión P-N tanto dentro como fuera del equilibrio. Para alcanzar este objetivo final se han planteado los siguientes objetivos específicos:

- Estudio bibliográfico: análisis de la bibliografía disponible sobre el modelado de la unión P-N siguiendo una aproximación mono-dimensional.

- Estudio de la unión P-N en el equilibrio: derivación de las ecuaciones de a resolver y estimación inicial de la solución mediante la aproximación de empobrecimiento.
- Estudio e implementación en MATLAB de métodos numéricos avanzados para la resolución de ecuaciones diferenciales:
 - Método iterativo de Newton-Raphson, tanto para funciones como para sistemas de ecuaciones.
 - Métodos de discretización de las ecuaciones diferenciales para su solución numérica.
- Aplicación e implementación de los métodos de estudiados a la resolución de la unión P-N en condiciones de equilibrio en el entorno de MATLAB.
- Extensión de los procedimientos desarrollados para la resolución de la unión P-N en condiciones de no-equilibrio.
- Análisis y discusión de resultados de varios casos prácticos en MATLAB, de forma que ilustren las ventajas de los procedimientos avanzados desarrollados frente a la aproximación clásica de empobrecimiento.

1.2 SUMARIO DE CONTENIDOS

Los contenidos de esta memoria se distribuyen de la siguiente forma:

- En la sección 2 “*Unión P-N*” vamos a mostrar los conceptos fundamentales la unión P-N, comenzando con la definición de un semiconductor y sus portadores, continuando con los efectos físicos que ocurren al perturbar un semiconductor, y finalizando con un estudio electrostático de la unión P-N en el que se definirá la zona de empobrecimiento y el potencial de contacto. En esta sección también se definirá la aproximación de empobrecimiento, y se examinarán los resultados que se obtienen con dicha aproximación.
- En la siguiente sección, “*Análisis Numérico en Condiciones de Equilibrio*”, se presentan los aspectos necesarios para el estudio de la unión P-N en

condiciones de equilibrio térmico (sin campo externo aplicado, sin iluminación, sin gradientes de temperatura, etc.):

- Comenzaremos analizando aspectos generales del método iterativo de Newton-Raphson, así como sus ventajas y limitaciones. Se obtendrán las fórmulas para resolver una ecuación con Newton-Raphson, y la extrapolación del método para un sistema de ecuaciones.
 - Posteriormente aplicaremos este procedimiento a nuestro modelo electrostático de la unión, poniendo énfasis en el proceso de discretización de las ecuaciones teóricas para la unión P-N y la particularización del sistema equivalente para trabajar con el método de Newton-Raphson [14]-[15].
 - Finalmente, indicaremos algunas consideraciones prácticas de la implementación en MATLAB, y se mostrará una explicación de las secciones principales del código generado.
- Continuaremos en la sección *“Análisis numérico en condiciones de no-equilibrio”* con el estudio de la unión P-N en condiciones fuera del equilibrio. Se introduce el método numérico de Scharfetter-Gummel [18]-[19] para la discretización de las ecuaciones de continuidad, de forma que se obtenga un sistema de tres ecuaciones que, unidas a las al procedimiento desarrollado para el caso de equilibrio, permitan resolver de forma iterativa la unión en no-equilibrio. De nuevo, pondremos énfasis en las consideraciones prácticas más relevantes para la implementación en MATLAB y el código generado.
 - Una vez presentado los procedimientos numéricos para el estudio de la unión P-N tanto dentro como fuera del equilibrio, pasaremos en la sección 5 a examinar algunos *“Casos Prácticos de Aplicación: Comparativa con la Aproximación Clásica de Empobrecimiento ”* con distintas condiciones de contorno, comparando cuando proceda con la aproximación de empobrecimiento.
 - Finalmente, en la sección 6 se presentarán las conclusiones principales del trabajo y posible desarrollo futuro.

2 UNIÓN P-N

En esta sección se muestran los conceptos principales relativos a nuestro objeto de estudio, la unión P-N, y se introduce la notación seguida en el resto de la memoria. Para ello vamos a considerar brevemente las propiedades de cada tipo de semiconductor (tipo p, dopado con impurezas aceptoras; y tipo n, dopado con impurezas donantes), además de los efectos que ocurren en los mismos cuando se introduce una perturbación. Finalmente se modelará la interacción en la unión P-N y la aproximación de empobrecimiento.

2.1 PROPIEDADES DE LOS SEMICONDUCTORES

Definimos un semiconductor como un material que posee características y comportamientos intermedios entre un material conductor y un aislante, de forma que propiedades como su resistividad y conductividad son muy variables frente a cambios en su entorno (temperatura, presión, campo eléctrico/magnético) y pueden exhibir comportamientos similares a los de aislantes y metales [12].

Los semiconductores extrínsecos (aquellos que han sido dopados con materiales aceptores o donantes) se pueden clasificar dos categorías, tipo p o tipo n, dependiendo de su portador mayoritario. La densidad de electrones por unidad de volumen es n , y de la misma manera la densidad de huecos por unidad de volumen es p . Se suelen expresar en cm^{-3} .

Tenemos que la concentración de portadores en equilibrio (corriente neta nula, a temperatura constante y sin efectos externos) para cada portador es:

$$n = n_i \cdot e^{\frac{E_F - E_i}{kT}} \quad p = n_i \cdot e^{\frac{E_i - E_F}{kT}} \quad (1)$$

siendo n_i la concentración de portadores intrínseca, E_F el nivel del Fermi, E_i el nivel de Fermi intrínseco y k la constante de Boltzmann. El valor de n_i para el silicio a 300K es aproximadamente de $10^{10} cm^{-3}$ [12].

Cuando se perturba el semiconductor pueden ocurrir una serie de efectos que también tendremos que modelar. Vamos a definir las corrientes de arrastre y difusión, y los procesos de generación y recombinación de portadores [3].

2.1.1 Arrastre

La corriente de arrastre consiste en el movimiento de partículas cargadas bajo la influencia de un campo eléctrico, de forma que los electrones se desplazan en el sentido contrario al campo eléctrico y los huecos a favor de este.

Bajo un campo eléctrico las partículas se aceleran, lo que llevaría a pensar que sufren un movimiento acelerado, pero a menudo esta aceleración se ve frenada por interacciones microscópicas como las colisiones e interacciones con la red cristalina. En general, podemos considerar que la aceleración se compensa con estas interacciones y los portadores se desplazan, aproximadamente, a velocidad constante. Así, la corriente de arrastre se define como [12]:

$$\vec{J}_p = q \cdot \mu_p \cdot p \cdot \vec{E} \quad \vec{J}_n = q \cdot \mu_n \cdot n \cdot \vec{E} \quad (2)$$

siendo q la carga de cada portador en valor absoluto, \vec{E} el campo eléctrico que provoca la corriente de arrastre, y μ la movilidad del portador. Esta última no solo depende del material semiconductor, sino que también tiene una dependencia compleja con la temperatura, el dopado, y el propio campo eléctrico.

Puesto que nos vamos a centrar en una aproximación mono-dimensional, a partir de este punto prescindiremos de la notación vectorial, y asumiremos en todo momento que $\vec{E}(x, y, z) = E(x)$, y así con el resto de las magnitudes vectoriales.

2.1.2 Difusión

La difusión es el proceso mediante el cual las partículas se redistribuyen al desplazarse debido a la agitación térmica, de forma que, al migrar desde zonas de alta densidad de partículas a zonas de baja densidad, tienden a distribuirse de manera uniforme en el espacio [4]. Este efecto será especialmente notable en la unión PN, ya que gracias a la difusión se establece la región de empobrecimiento.

La difusión depende de la existencia de una región con mayor densidad de portadores que otra, es decir, debe existir un gradiente de concentración de portadores para que exista corriente. Además, a mayor gradiente, mayor se espera que sea el efecto de difusión.

$$\begin{aligned} J_p &= -q \cdot D_p \cdot \frac{\partial p}{\partial x} \\ J_n &= -q \cdot D_n \cdot \frac{\partial n}{\partial x} \end{aligned} \quad (3)$$

donde D es el coeficiente de difusión, que depende principalmente del material y de la temperatura.

En general, las corrientes de difusión y de arrastre se pueden sumar para obtener la corriente neta total, $J_{total} = J_{difusión} + J_{arrastrre}$.

$$\begin{aligned} J_p &= q \cdot \mu_p \cdot p \cdot E - kT \cdot \mu_p \cdot \frac{\partial p}{\partial x} \\ J_n &= q \cdot \mu_n \cdot n \cdot E + kT \cdot \mu_n \cdot \frac{\partial n}{\partial x} \end{aligned} \quad (4)$$

Donde hemos tenido en cuenta la relación de Einstein $D = \frac{kT}{q} \mu$ [12].

2.1.3 Generación y recombinación

La generación y recombinación son el conjunto de procesos que crean o aniquilan pares electrón-hueco [5]. Se suelen considerar en conjunto ya que son procesos opuestos, y en general solo es relevante cuál de las dos tendencias, la generación o la recombinación, es dominante en el semiconductor a estudiar.

La recombinación sucede cuando un electrón y un hueco coinciden en el mismo espacio y se aniquilan, liberando energía en forma de fotón o energía térmica. Esto ocurre normalmente cuando un electrón en la banda de conducción se desplaza a la banda de valencia, aunque también pueden existir niveles añadidos por impurezas que actúen como centros de recombinación [12]. De manera análoga, si se proporciona suficiente energía estos procesos se pueden revertir, promocionando un electrón a la banda de valencia. Esto ocurre, por ejemplo, mediante el paso de partículas ionizantes a través del semiconductor.

2.1.4 Ecuaciones de continuidad

Los procesos de difusión, arrastre, generación y recombinación afectan a la concentración de portadores en el semiconductor, y su efecto se puede combinar para obtener una visión general de la evolución temporal de los portadores. Si

además consideramos que debe existir continuidad temporal y espacial, podemos definir ecuaciones que describan el comportamiento de estos portadores.

$$\frac{\partial p}{\partial t} = -\frac{1}{q} \frac{\partial J_p}{\partial x} + G_L - U \quad (5)$$

$$\frac{\partial n}{\partial t} = +\frac{1}{q} \frac{\partial J_p}{\partial x} + G_L - U \quad (6)$$

siendo la primera parte de esta ecuación la variación temporal de la densidad de portadores, J la densidad de corriente, G_L la tasa de fotogeneración y U la suma de las tasas de recombinación y generación.

Estas ecuaciones pueden simplificarse si consideramos estados estacionarios o si despreciamos los efectos de generación y recombinación.

2.2 ESTUDIO ELECTROSTÁTICO DE LA UNIÓN PN

La unión PN es la unión metalúrgica entre un semiconductor tipo P y uno tipo N, de forma que, teniendo un lado p con un dopaje N_a y el lado n un dopaje N_d , la unión se encuentra en el punto en el que $N_a(x) - N_d(x) = 0$.

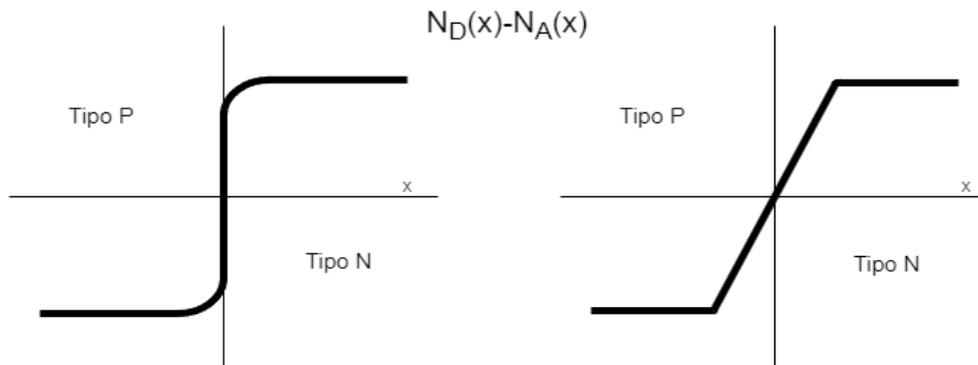


Fig. 1 Perfiles de dopado, Unión abrupta vs Unión gradual.

A la hora de realizar el estudio nos conviene realizar determinadas aproximaciones que nos permitan agilizar los cálculos pero que nos proporcionen una solución satisfactoria. En esta ocasión vamos a recurrir a un análisis unidimensional en dirección perpendicular a la unión. También vamos a considerar que tenemos una unión ideal y abrupta en $x = 0$. Hacemos un estudio electrostático (sin campo externo, ni procesos de generación/recombinación) porque nos proporciona un

marco de referencia desde el que podemos extrapolar comportamientos más complejos, como perturbaciones.

Al realizarse la unión PN, se rompe el equilibrio existente en ambos conductores y se genera una primera corriente de difusión debido al gradiente de portadores que existe a ambos lados de la unión. Los portadores mayoritarios de cada región atraviesan la unión y pasan a ser portadores minoritarios de la región opuesta, provocando dos efectos:

- Los portadores de carga libre se recombinan, reduciendo la carga libre en la unión (lo que llamamos empobrecimiento, $p \approx n \approx 0$)
- La carga fija (inmóvil) queda desbalanceada de forma que surge una corriente de arrastre que se opone a la de difusión.

Este proceso continúa hasta que se alcanza el equilibrio, cuando las corrientes de arrastre y de difusión se anulan mutuamente.

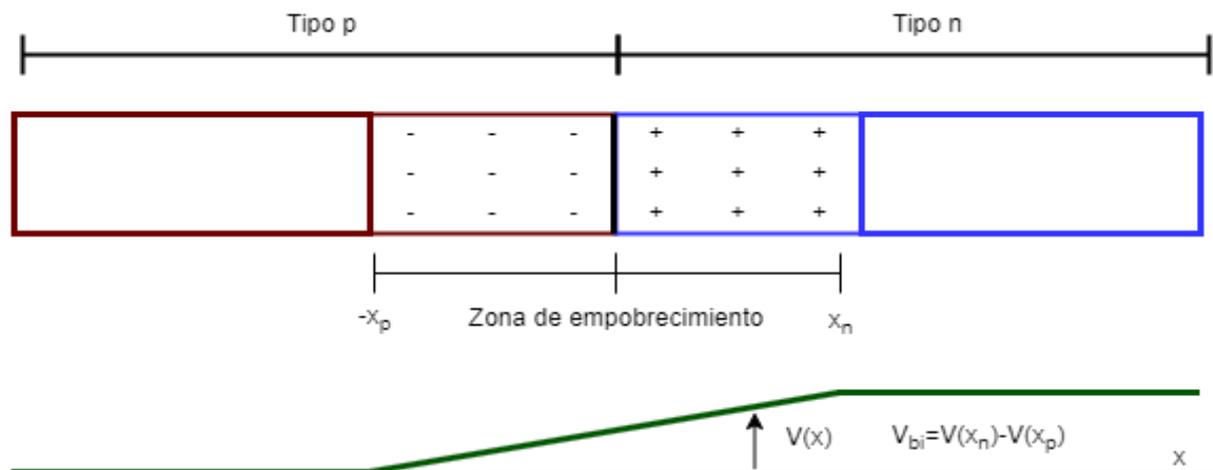


Fig. 2 Zona de empobrecimiento en equilibrio, comprendida entre los anchos de la región de empobrecimiento $-x_p$ y x_n , y junto con la caída de potencia generada en la unión, V_{bi} .

La caída de potencial que se aprecia en la unión se conoce como potencial de contacto, V_{bi} , y proviene de la existencia de un campo eléctrico no nulo. Si integramos este campo en la región de empobrecimiento:

$$\int_{-x_p}^{x_n} E \, dx = \int_{V(-x_p)}^{V(x_n)} dV = V_{bi} \quad (7)$$

Además, conocemos que la corriente neta en el semiconductor, la suma de la corriente de arrastre y la de difusión, es igual a 0, es decir:

$$J = q\mu_n E + qD_N \frac{\partial n}{\partial x} = 0 \quad (8)$$

Si resolvemos para \vec{E} y tenemos en cuenta la relación de Einstein $\frac{D_N}{\mu_N} = \frac{kT}{q}$ [12], obtenemos:

$$E = -\frac{D_N}{\mu_n} \frac{1}{n} \frac{\partial n}{\partial x} = -\frac{kT}{q} \frac{1}{n} \frac{\partial n}{\partial x} \quad (9)$$

Sustituyendo el campo \vec{E} en la integral (7) y resolviendo, finalmente obtenemos nuestra expresión para V_{bi} .

$$V_{bi} = \int_{-x_p}^{x_n} \bar{E} dx = \frac{kT}{q} \int_{n(-x_p)}^{n(x_n)} \frac{dn}{n} = \frac{kT}{q} \ln \left[\frac{n(x_n)}{n(-x_p)} \right] \quad (10)$$

Consideramos que en los extremos de la unión el número de portadores intrínsecos es despreciable con respecto a los extrínsecos, es decir, que podemos aproximar $n(x_n) = N_D$ y $n(-x_p) = \frac{n_i^2}{N_A}$. Podemos sustituir estos valores para obtener el potencial en función del dopaje:

$$V_{bi} = \frac{kT}{q} \ln \left[\frac{N_A N_D}{n_i^2} \right] \quad (11)$$

2.3 APROXIMACIÓN DE EMPOBRECIMIENTO

Para obtener una solución numérica para la unión P-N es necesario resolver su ecuación de Poisson en equilibrio, con condiciones de contorno $V(-x_p) = 0$ y $V(x_n) = V_{bi}$.

$$\frac{\partial^2 V}{\partial x^2} = \frac{q}{\epsilon_0} (p - n + N_D - N_A) \quad (12)$$

Para resolver esta expresión es necesario conocer la concentración de portadores p y n en función de x y el valor de V_{bi} (11). Para obtener la concentración de portadores

en función de la posición podemos hacer uso de las expresiones (1). Así, si derivamos (1) con respecto a x :

$$\frac{\partial n}{\partial x} = -\frac{1}{kT} n_i e^{\frac{EF-E_i}{kT}} \frac{\partial E_i}{\partial x} \rightarrow \frac{\partial E_i}{\partial x} = -kT \frac{1}{n} \frac{\partial n}{\partial x} \quad (13)$$

y tenemos en cuenta la expresión del campo (9), podemos obtener una expresión de p en función de x . El desarrollo es equivalente para obtener $n(x)$.

$$p(x) = N_a e^{-V(x)\frac{q}{kT}} \quad n(x) = N_D e^{V(x)\frac{q}{kT}} \quad (14)$$

Con estas expresiones, ya tenemos toda la información necesaria para resolver la ecuación de Poisson.

Una aproximación habitual que facilita su cálculo y ofrece una solución satisfactoria es la aproximación de empobrecimiento, es decir, considerar que la densidad de portadores es nula fuera de la zona de empobrecimiento, y dentro de la zona se puede aproximar como qN_A en la parte tipo p y $-qN_D$ en la de tipo n , es decir, un cambio abrupto en la unión $x = 0$. De esta manera la ecuación de Poisson se puede particularizar como sigue:

$$\frac{\partial E}{\partial x} = \frac{\rho(x)}{\varepsilon} \rightarrow \frac{\partial E}{\partial x} = \frac{q}{\varepsilon} \begin{cases} N_D & 0 < x < x_n \\ N_A & -x_p < x < 0 \end{cases} \quad (15)$$

Por último, resulta conveniente tener una expresión para los anchos de cada región de la zona de empobrecimiento, x_p y x_n . La podemos obtener a partir de la expresión del potencial de contacto (11) y sabiendo que $N_A \cdot x_p = N_D \cdot x_n$.

$$x_p = \left[\frac{2\varepsilon}{q} V_{bi} \frac{1}{N_a} \frac{N_d}{N_d + N_a} \right]^{\frac{1}{2}} \quad x_n = \left[\frac{2\varepsilon}{q} V_{bi} \frac{1}{N_d} \frac{N_a}{N_d + N_a} \right]^{\frac{1}{2}} \quad (16)$$

siendo la anchura total de la zona de empobrecimiento $W = x_p + x_n$.

3 ANÁLISIS NUMÉRICO EN CONDICIONES DE EQUILIBRIO

Para obtener una solución cuantitativa de la ecuación de Poisson (12) sin tener que recurrir a la aproximación de empobrecimiento es necesario utilizar métodos numéricos [8]-[9], y en este caso vamos a optar, por su robustez y precisión, por Newton-Raphson.

3.1 REVISIÓN GENERAL DEL MÉTODO DE NEWTON-RAPHSON

El método de Newton-Raphson es un algoritmo que se utiliza para encontrar los ceros de una función $f(x)$, de forma que, dadas unas condiciones iniciales, termine por converger en uno de los ceros de la ecuación. Para ello utiliza la fórmula de Newton-Raphson:

$$f'(x_i) = \frac{f(x_i) - 0}{x_i - x_{i+1}} \rightarrow x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)} \quad (17)$$

donde $f(x)$ es la función cuyos ceros queremos encontrar, y $f'(x)$ su derivada.

Es decir, dada una aproximación inicial a la solución x_0 , se tiene que el valor $x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$ se acerca más a la solución que nuestro x_0 inicial. Newton-Raphson es uno de los métodos conocidos como métodos de punto fijo [9], que se basan en encontrar una función iterativa que se aproxime al cero de una función dada una estimación inicial.

Newton-Raphson tiene, en general, un orden de convergencia 2 [9], así que se aproxima a la solución de manera cuadrática. Esto significa que el número de decimales correctos se dobla con cada iteración, lo que lo convierte en un método muy potente para obtener las raíces de una función.

Para aplicar este método es necesario disponer no solo de la función inicial, si no de su derivada. Además, la estimación inicial debe escogerse adecuadamente para evitar problemas de convergencia. Si x_0 se encuentra demasiado lejos de la solución, el algoritmo no garantiza la convergencia y en general tenderá a converger más lento, teniendo que realizar muchas más iteraciones (y, por tanto, consumir más memoria y tiempo) que si tuviera una estimación más cercana a la solución.

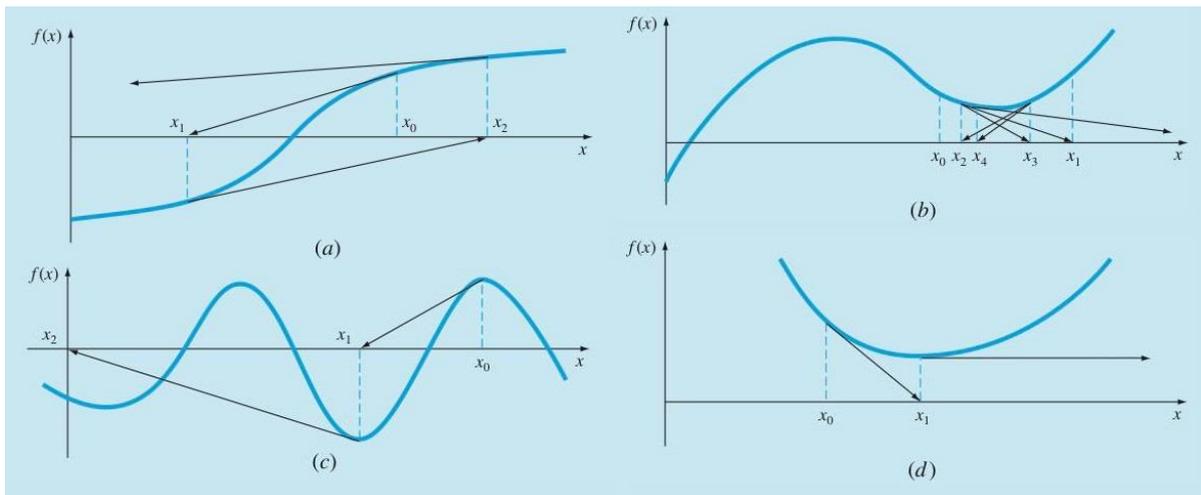


Fig. 3 Distintos casos de convergencia para Newton-Raphson [8].

Puesto que este método se basa en aproximar utilizando la tangente, puede darse el caso de que x_0 se encuentre cerca de un punto con una pendiente cercana a 0, y provoque que la solución se aleje demasiado de nuestra estimación inicial (Fig. 3a). Este método también tiende a oscilar alrededor de puntos extremos de la función (Fig. 3b). En el peor de los casos, si la pendiente es igual a 0, la tangente será paralela al eje x y la fórmula Newton-Raphson (17) efectuará una división entre 0 (Fig. 3d). Tampoco es adecuado para funciones con múltiples ceros, ya que la convergencia es mucho más lenta (lineal, en vez de cuadrática) y puede realizar múltiples iteraciones hasta acercarse suficientemente a uno de los ceros.

Newton-Raphson se puede extrapolar para sistemas de ecuaciones, calculando las raíces de todas las ecuaciones de manera simultánea. Si estamos tratando con ecuaciones lineales, pueden expresarse de la siguiente forma:

$$f_i(V) = 0, \quad i = 1, 2, \dots, m. \quad (18)$$

siendo V un vector de m dimensiones que contiene las ecuaciones del sistema:

$$V = \begin{pmatrix} f_1(x_1, x_2, \dots, x_m) \\ f_2(x_1, x_2, \dots, x_m) \\ \vdots \\ f_m(x_1, x_2, \dots, x_m) \end{pmatrix} \quad (19)$$

Para aplicar Newton-Raphson necesitamos la derivada de la función, que al tratarse de un sistema de ecuaciones equivale a obtener su Jacobiano.

$$J = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \dots & \frac{\partial f_1}{\partial x_m} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \dots & \frac{\partial f_2}{\partial x_m} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \frac{\partial f_m}{\partial x_2} & \dots & \frac{\partial f_m}{\partial x_m} \end{pmatrix} \quad (20)$$

Por último, para aplicar el método iterativo será esencial definir el vector de corrección ΔV , que servirá para mejorar nuestra estimación inicial en cada iteración:

$$\Delta V = J^{-1}V \quad (21)$$

De forma que, siguiendo la fórmula de Newton Raphson (17), podemos sumar este vector de corrección a nuestra primera estimación V_0 para obtener un nuevo vector, $V_1 = V_0 + \Delta V$, que estará más cerca de la solución. El proceso se podrá repetir hasta que el vector de corrección sea suficientemente pequeño, o hasta que consideremos que la solución V_i es adecuada.

3.2 APLICACIÓN A LA ECUACIÓN DE POISSON

Para resolver la ecuación de Poisson (12) mediante el método de Newton-Raphson vamos a implementar el algoritmo en MATLAB, de forma que, dados unos valores iniciales, sea capaz de obtener una solución numérica a nuestro problema. Para ello será necesario discretizar nuestras ecuaciones. Utilizando el método de las diferencias finitas, que se describe de manera generalizada en la referencia [8], podemos derivar las ecuaciones obtenidas en la sección 2 y, una vez tengamos las expresiones, escribir un código que lleve a cabo el proceso y nos devuelva los resultados.

Para discretizar la ecuación de Poisson (12) utilizaremos la aproximación de derivadas en diferencias finitas [9]. Tenemos que la primera y segunda derivada de una función $f(x_i)$ se pueden representar de la siguiente manera:

$$f'(x_i) = \frac{f(x_{i+1}) - f(x_i)}{\Delta x} \quad (22)$$

$$f''(x_i) = \frac{f(x_{i+2}) + 2f(x_{i+1}) - f(x_i)}{\Delta x^2} \quad (23)$$

Siendo Δx la longitud del intervalo sobre el que estamos haciendo la aproximación, más conocido como *step size*, y el término $f(x_{i+1}) - f(x_i)$ la diferencia progresiva. Vamos a escoger V_i como nuestra variable, es decir, nuestra función será $f(V_i)$. Haciendo uso de la ecuación (10) podemos integrar el campo en nuestro intervalo $\Delta x = x_{i+1} - x_i$ y así obtener:

$$V_{i+1} - V_i = \frac{kT}{q} \ln \left[\frac{n(x_{i+1})}{n(x_i)} \right] \quad V_{i+1} - V_i = \frac{kT}{q} \ln \left[\frac{p(x_i)}{p(x_{i+1})} \right] \quad (24)$$

de donde podemos despejar las expresiones discretizadas de $n(x_i)$ y $p(x_i)$ que luego podremos sustituir en Poisson, teniendo en cuenta que $\frac{kT}{q} = V_T$.

$$n_{i+1} = n_i e^{\frac{V_{i+1} - V_i}{V_T}} \quad p_{i+1} = p_i e^{\frac{V_i - V_{i+1}}{V_T}} \quad (25)$$

Lo único que nos queda es la segunda derivada $\frac{d^2V}{dx^2}$, que obtenemos sustituyendo directamente en la expresión (23).

$$f''(V_i) = \frac{f(V_{i+2}) - 2f(V_{i+1}) + f(V_i)}{\Delta x^2} \quad (26)$$

Combinando los resultados de (25) y (26), y sustituyendo en nuestra ecuación de Poisson, finalmente nos queda:

$$f(V_i) = \frac{V_{i+2} - 2V_{i+1} + V_i}{\Delta x^2} - \frac{q}{\varepsilon} \left(p_i \cdot e^{\frac{V_{i+1} - V_i}{V_T}} - n_i \cdot e^{\frac{V_{i+1} - V_i}{V_T}} + N_D - N_A \right) = 0 \quad (27)$$

Esta función es la que tenemos que derivar a la hora de aplicar la fórmula de Newton-Raphson (17). Para simplificar los cálculos, vamos a derivar los dos términos de la ecuación (27) por separado.

$$\begin{aligned}
 \frac{\partial f(V)}{\partial V} &= \frac{\Delta V_{i+2} - 2\Delta V_{i+1} + \Delta V_i}{\Delta x^2} - \frac{q}{\varepsilon} \left(\frac{\partial p}{\partial V} - \frac{\partial n}{\partial V} \right) \\
 &\downarrow \\
 \frac{q}{\varepsilon} \left(\frac{\partial p}{\partial V} - \frac{\partial n}{\partial V} \right) &= \frac{q}{\varepsilon} \left(\frac{\Delta V_i - \Delta V_{i+1}}{V_T} \cdot p_i e^{\frac{V_i - V_{i+1}}{V_T}} - \frac{\Delta V_{i+1} - \Delta V_i}{V_T} \cdot n_i e^{\frac{V_{i+1} - V_{i-1}}{V_T}} \right) = \\
 &= \frac{q}{\varepsilon} \frac{\Delta V_i - \Delta V_{i+1}}{V_T} \left(p_i e^{\frac{V_i - V_{i+1}}{V_T}} + n_i e^{\frac{V_{i+1} - V_{i-1}}{V_T}} \right) \\
 &\downarrow \\
 \frac{\partial f(V)}{\partial V} &= \frac{\Delta V_{i+2} - 2\Delta V_{i+1} + \Delta V_i}{\Delta x^2} - \frac{q}{\varepsilon} \frac{\Delta V_i - \Delta V_{i+1}}{V_T} \left(p_i e^{\frac{V_i - V_{i+1}}{V_T}} + n_i e^{\frac{V_{i+1} - V_{i-1}}{V_T}} \right)
 \end{aligned} \tag{28}$$

Al sustituir (27) y (28) en la fórmula de Newton-Raphson nos queda una ecuación que podemos representar de manera matricial:

$$\begin{bmatrix} D_1 & D_1^{up} & 0 & 0 & 0 \\ D_2^{down} & D_2 & D_2^{up} & 0 & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & D_{m-3}^{down} & D_{m-3} & D_{m-3}^{up} \\ 0 & 0 & 0 & D_{m-2}^{down} & D_{m-2} \end{bmatrix} \begin{bmatrix} \Delta V_1 \\ \Delta V_2 \\ \vdots \\ \Delta V_{m-3} \\ \Delta V_{m-2} \end{bmatrix} = \begin{bmatrix} f(V_1) \\ f(V_2) \\ \vdots \\ f(V_{m-3}) \\ f(V_{m-2}) \end{bmatrix} \tag{29}$$

donde m será nuestro parámetro de mallado, que calcularemos más tarde a partir de Δx y el ancho de nuestra región de empobrecimiento. Los coeficientes de la matriz tridiagonal son los siguientes:

$$\begin{aligned}
 D_i &= \frac{-2}{\Delta x^2} + \frac{q}{\varepsilon} \frac{1}{V_T} \left(p_i e^{\frac{V_i - V_{i+1}}{V_T}} + n_i e^{\frac{V_{i+1} - V_{i-1}}{V_T}} \right) \\
 D_i^{up} &= \frac{1}{\Delta x^2}, \quad D_i^{down} = \frac{1}{\Delta x^2} - \frac{q}{\varepsilon} \frac{1}{V_T} \left(p_i e^{\frac{V_i - V_{i+1}}{V_T}} + n_i e^{\frac{V_{i+1} - V_{i-1}}{V_T}} \right)
 \end{aligned} \tag{30}$$

Una vez determinadas todas las expresiones, nuestro algoritmo de Newton-Raphson se encuentra perfectamente definido para su implementación en MATLAB.

3.3 CONSIDERACIONES PRÁCTICAS PARA LA IMPLEMENTACIÓN DEL MÉTODO EN MATLAB

Una vez conformado el problema para su implementación en MATLAB, en esta sección vamos a introducir algunas consideraciones prácticas en su implementación. El código cuenta con cinco partes diferenciadas:

- Un apartado inicial donde se definen las constantes a utilizar en la simulación, y se calculan parámetros de interés a partir de las mismas.
- Creación del mallado a partir de las condiciones de contorno, entendiendo como mallado el paso mínimo para la discretización de las ecuaciones.
- Cálculo de una primera aproximación para la tensión V y la concentración de portadores p y n .
- Proceso iterativo que contendrá el algoritmo de Newton-Raphson.
- Apartado final donde se realizan cálculos a partir de los resultados, y se representan gráficamente.

En el Anexo I se encuentra un diagrama de flujo que explica la esencia del código, tanto la sección de equilibrio como la posterior ampliación para no equilibrio.

3.3.1 Constantes y parámetros

El primer lugar habrá que definir todas las constantes necesarias, cuyo valor variará para distintos materiales o condiciones iniciales. Nosotros nos vamos a centrar en silicio a 300K. Las concentraciones de impurezas donantes y aceptoras, N_d y N_a , también se encuentran en este apartado. Además se calculará el potencial de contacto (11), el ancho de la región de empobrecimiento (16), y la longitud de Debye L_D [20]-[21] de cada portador:

$$L_D = \sqrt{\frac{\varepsilon \cdot k_b T}{q \cdot N}} \quad (31)$$

Siendo N el dopaje neto, que equivale a N_A para el semiconductor tipo p, N_D para el tipo n, n_i si se trata de un semiconductor intrínseco. La longitud de Debye es relevante.

3.3.2 Parámetros de simulación

Una vez definidos los parámetros del problema, es importante asegurarse de que nuestro intervalo de integración Δx sea bastante menor que la anchura de la zona de empobrecimiento (ya que no podríamos apreciar correctamente dicha región) y menor a la longitud de Debye de los portadores. Como compromiso, definimos: 1) Δx como un tercio de la menor longitud entre estos parámetros, 2) los límites de simulación a tres veces el ancho de cada zona de empobrecimiento, y 3) el mallado como el número entero de intervalos Δx que caben en nuestros límites de simulación. El parámetro de mallado, m , será el que determine la longitud de los vectores.

```
dx = min([Ldn, Ldp, Ldi, Wn, Wp])/3;  
xmin = - Wp * 3;  
xmax = Wn * 3;  
m = round((xmax - xmin) / dx);  
x = linspace(xmin, xmax, m)';
```

Fig. 4 Definición de parámetros de simulación en función de la zona de empobrecimiento.

3.3.3 Estimación inicial

La estimación inicial para la tensión V se obtiene de la expresión (11), considerando que en la parte tipo p de la unión $N_D \approx n_i$ y en la parte tipo n, $N_A \approx n_i$. Se puede diferenciar en qué parte de la unión metalúrgica nos encontramos fijándonos en el signo de x (negativo en la parte p, positivo en la n).

```
for i = 1:m  
    if x(i) < 0  
        NA(i) = Na; V(i) = Vp;  
    else  
        ND(i) = Nd; V(i) = Vn;  
    end  
end
```

Fig. 5 Bucle for que establece la estimación inicial de la tensión V y el dopaje N_A y N_D .

Para el valor inicial de la concentración de portadores solo necesitamos las condiciones de contorno, que en nuestro caso corresponde a definir el primer elemento de cada vector. Asumimos que en el extremo p de la unión se cumple que $p \approx N_A$, y por tanto $n = n_i^2/N_A$. El resto de la concentración se evaluará dentro del bucle, ya que depende de la tensión (25).

```
n = zeros(m,1);  
p = zeros(m,1);  
n(1) = (ni.^2)/Na;  
p(1) = Na;
```

Fig. 6 Condiciones de contorno para la concentración de portadores.

3.3.4 Newton-Raphson

Para implementar Newton-Raphson vamos a crear un bucle que repita el proceso hasta que el error, que depende de la magnitud del vector de corrección (21), sea lo suficientemente pequeño, es decir, cuando consideremos que las correcciones que se introducen con el algoritmo son demasiado pequeñas. El valor de tolerancia no está en principio definido y dependerá de cómo de precisa deseamos que sea nuestra solución y del número de iteraciones máximas deseadas. Inicialmente hemos considerado una tolerancia de error de 10^{-6} y un número máximo de iteraciones de 100.

El primer paso será evaluar la concentración de portadores utilizando la expresión obtenida en (25). Es decir, para obtener el punto n_{i+1} necesitamos el punto n_i , y lo mismo ocurre para p_{i+1} . Repetimos el proceso $i = 1, 2, \dots, m - 1$ hasta llegar a n_m y p_m .

```
for i = 1:m-1  
n(i+1) = n(i) * exp( (V(i+1) - V(i))/Vt);  
p(i+1) = p(i) * exp(-(V(i+1) - V(i))/Vt);  
end
```

Fig. 7 Estimación inicial de la concentración de portadores.

Para obtener la función $f(V_i)$ que discretizamos anteriormente (27), vamos a calcular, por comodidad, cada término por separado, y finalmente lo sumaremos.

```
d2V_by_dx2 = (V(1:m-2)-2*V(2:m-1)+V(3:m,1))/dx2;
rho = q*(ND(2:m-1)-NA(2:m-1)+p(2:m-1)-n(2:m-1));
F = d2V_by_dx2 + rho/eps;
```

Fig. 8 Definición de la función $f(V_i) = F$ como suma del primer y segundo término (27).

Por último, la matriz del sistema (29) se obtiene calculando cada una de las diagonales (30) y sumándolas. Conviene utilizar la función *sparse* para ahorrar memoria, ya que genera una matriz que sólo almacena los términos distintos a 0.

```
dp_dV = - p(1:m-2) .* exp(-(V(1:m-2) - V(2:m-1))/Vt);
dn_dV = n(1:m-2) .* exp( (V(1:m-2) - V(2:m-1))/Vt);
drho_dV = q * (dp_dV - dn_dV) / Vt;

dup = sparse(diag(1/dx2 + drho_dV(2:m-2)/eps , 1));
d = sparse(diag(2/dx2 - drho_dV/eps));
down = sparse(diag(ones(m-3,1)/dx2 , -1));
MD = d-(dup+down);
```

Fig. 9 Definición de la matriz (29) mediante suma de sus diagonales.

La resolución del sistema de ecuaciones (29) se obtiene fácilmente multiplicando la matriz D por la inversa del vector F. Así obtenemos nuestro vector de corrección ΔV , que podemos sumar a nuestra estimación inicial V_i para obtener una nueva solución V_{i+1} , para que se utilice en la siguiente iteración.

```
DV = MD\F;
V(2:m-1) = V(2:m-1) + DV;
```

Fig. 10 Obtención y aplicación del vector de corrección ΔV .

Es importante señalar que este vector de corrección se calcula de forma que no afecte al primer y último punto del vector V, ya que corresponden a las condiciones de contorno y deben mantenerse fijas por toda la simulación.

Por último, vamos a evaluar el error para considerar si es necesario seguir con la siguiente iteración. Además, por precaución, hemos añadido una condición para

evitar que, en caso de que la solución no sea convergente o converja muy despacio, se detenga el proceso y nos proporcione un aviso de error, que deberemos considerar a la hora de ajustar las condiciones iniciales.

```
Error=norm(DV,2)/sqrt(m);  
  
if k_iter>=100  
    error('Demasiadas iteraciones, es probable que no haya  
convergencia')  
end
```

Fig. 11 Evaluación de las condiciones para continuar el bucle.

El proceso se repetirá desde el primer paso hasta que el error sea inferior al umbral de tolerancia.

3.3.5 Representación gráfica

Para visualizar gráficamente cada iteración de Newton-Raphson, se puede introducir dentro del bucle una función que dibuje V y ΔV para cada iteración. Esto además nos sirve para comprobar de manera intuitiva si existe convergencia, ya que vemos la función evolucionar en tiempo real, pero incrementa considerablemente el tiempo de simulación.

Finalmente, y una vez hayamos obtenido nuestro potencial V al alcanzar la convergencia, podemos representarlo y utilizar esta solución para obtener también la concentración de portadores p y n , la densidad de carga total y el campo eléctrico a través de la unión. También calcularemos V mediante la aproximación de empobrecimiento para compararlo con nuestro valor obtenido numéricamente.

4 ANÁLISIS NUMÉRICO EN CONDICIONES DE NO-EQUILIBRIO

Hasta ahora hemos asumido que nos encontrábamos en una situación de equilibrio, es decir, aquella en la que no existe corriente en la unión y no hay efectos de generación y recombinación.

Podemos ampliar nuestro modelo para tener en cuenta estas perturbaciones, para lo que tendremos que discretizar las ecuaciones de continuidad (5) y (6). Sin embargo, si intentamos discretizar la corriente neta (4) para sustituirla en la ecuación, nos encontramos con inestabilidad numérica cuando la variación de carga entre dos puntos x_i y x_{i+1} es mayor que $2V_T$ [18]. Para solventar este problema se podría incrementar el número de puntos m del mallado para hacerla más fina y evitar inestabilidades, u optar por un método de discretización más robusto. En esta ocasión vamos a apoyarnos en el método de Scharfetter-Gummel [18].

4.1 SCHARFETTER-GUMMEL

El método de Scharfetter-Gummel aplicado a la unión P-N [18][19] consiste en discretizar las ecuaciones de corriente bajo dos premisas:

- El valor de J es constante entre los puntos del mallado.
- El potencial V es lineal entre los puntos del mallado y, por tanto, $\frac{dV}{dx}$ es constante.

De esta forma, al discretizar obtenemos una ecuación diferencial de coeficientes constantes.

Para implementar este método, comenzamos expresando J utilizando el método de diferencias finitas, pero tomando como referencia su valor en el punto intermedio entre x_i y x_{i+1} :

$$f(x_i) = \frac{J_{i+1/2} - J_{i-1/2}}{\Delta x} \quad (32)$$

Antes de continuar vamos a reescribir la ecuación (4) teniendo en cuenta que nos encontramos en una dimensión (es decir, que $\nabla p = \frac{dp}{dx}$) y que podemos expresar el

campo en función del potencial (9), que ya tenemos discretizado. En este escenario, nos quedaría una ecuación diferencial para cada portador en la forma:

$$\begin{aligned} J_{p,i+1/2} &= -q \cdot \mu_p \cdot \frac{kT}{q} \left(p(x) \cdot \frac{q}{kT} \frac{V_{i+1} - V_i}{\Delta x} + \frac{\partial p}{\partial x} \right) \\ J_{n,i+1/2} &= -q \cdot \mu_n \cdot \frac{kT}{q} \left(n(x) \cdot \frac{q}{kT} \frac{V_{i+1} - V_i}{\Delta x} - \frac{\partial n}{\partial x} \right) \end{aligned} \quad (33)$$

Es decir, una ecuación diferencial lineal de la forma $f(x) = p(x) \cdot G(x) + \frac{\partial p(x)}{\partial x}$ con coeficientes no constantes. Este tipo de ecuaciones pueden resolverse utilizando el factor integrante $\mu(x) = \exp \int G(x) \partial x$ [22], que, al multiplicarse por la ecuación diferencial, permite que pueda ser resuelta mediante métodos de integración clásicos. En nuestro caso, el factor integrante equivale a $\exp \left(\frac{q}{kT} \frac{V_{i+1} - V_i}{\Delta x} (x - x_i) \right)$. Si lo multiplicamos a ambos lados de la ecuación:

$$J \cdot e^{\left(\frac{q}{kT} \frac{V_{i+1} - V_i}{\Delta x} (x - x_i) \right)} = -q \cdot \mu_p \cdot \frac{kT}{q} \left(p(x) \cdot \frac{q}{kT} \frac{V_{i+1} - V_i}{\Delta x} + \frac{\partial p}{\partial x} \right) \cdot e^{\left(\frac{q}{kT} \frac{V_{i+1} - V_i}{\Delta x} (x - x_i) \right)}$$

e integramos ambas partes por separado, empezando por J:

$$\begin{aligned} \int J_i \cdot \mu(x) \partial x &= J_i \int_{x_i}^{x_{i+1}} e^{\frac{q}{kT} \frac{V_{i+1} - V_i}{\Delta x} (x - x_i)} \partial x = J_i \cdot e^{\frac{q}{kT} \frac{V_{i+1} - V_i}{\Delta x} (-x_i)} \int_{x_i}^{x_{i+1}} e^{\frac{q}{kT} \frac{V_{i+1} - V_i}{\Delta x} (x)} \partial x \\ &= J_i \cdot e^{\frac{q}{kT} \frac{V_{i+1} - V_i}{\Delta x} (-x_i)} \cdot \frac{kT}{q} \frac{\Delta x}{V_{i+1} - V_i} \int_{x_i}^{x_{i+1}} \frac{q}{kT} \frac{V_{i+1} - V_i}{\Delta x} e^{\frac{q}{kT} \frac{V_{i+1} - V_i}{\Delta x} (x)} \partial x \\ &= J_i \cdot e^{\frac{q}{kT} \frac{V_{i+1} - V_i}{\Delta x} (-x_i)} \cdot \frac{kT}{q} \frac{\Delta x}{V_{i+1} - V_i} \cdot \left(e^{\frac{q}{kT} \frac{V_{i+1} - V_i}{\Delta x} (x)} \right) \Bigg|_{x_i}^{x_{i+1}} \\ &= J_i \cdot \frac{kT}{q} \frac{\Delta x}{V_{i+1} - V_i} \cdot e^{\frac{q}{kT} \frac{V_{i+1} - V_i}{\Delta x} (-x_i)} \cdot \left(e^{\frac{q}{kT} \frac{V_{i+1} - V_i}{\Delta x} (x_{i+1})} - e^{\frac{q}{kT} \frac{V_{i+1} - V_i}{\Delta x} (x_i)} \right) \\ &= J_i \cdot \frac{kT}{q} \frac{\Delta x}{V_{i+1} - V_i} \cdot \left(e^{\frac{q}{kT} \frac{V_{i+1} - V_i}{\Delta x} \frac{\Delta x}{(x_{i+1} - x_i)}} - 1 \right) \end{aligned} \quad (34)$$

Si en esta expresión, continuamos con la parte que corresponde a los portadores, podemos aprovechar la propiedad de la derivada de la multiplicación de dos funciones, ya que $\frac{\partial \mu}{\partial x} = \frac{q}{kT} \frac{V_{i+1} - V_i}{\Delta x} \cdot \mu(x)$:

$$\begin{aligned}
 & -q\mu_p V_T \int_{x_i}^{x_{i+1}} \left(p(x) \cdot \frac{q}{kT} \frac{V_{i+1} - V_i}{\Delta x} \cdot \mu(x) + \frac{\partial p}{\partial x} \cdot \mu(x) \right) \partial x = -q\mu_p V_T \int_{x_i}^{x_{i+1}} \left(p(x) \cdot \frac{\partial \mu}{\partial x} + \frac{\partial p}{\partial x} \cdot \mu(x) \right) \partial x \\
 & = -q\mu_p V_T \int_{x_i}^{x_{i+1}} \frac{\partial (p(x) \cdot \mu(x))}{\partial x} \partial x = -q\mu_p V_T [p(x) \cdot \mu(x)]_{x_i}^{x_{i+1}} = -q\mu_p V \cdot e^{\frac{V_{i+1} - V_i}{V_T}} \cdot (p_i - p_{i-1})
 \end{aligned} \tag{35}$$

Combinando las expresiones (34) y (35), y despejando J, se obtiene para un caso general en el que $V_{i+1} \neq V_i$:

$$J_{p,i+1/2} = \frac{q\mu_p \cdot V_T}{\Delta x} \frac{V_{i+1} - V_i}{V_T} \left(\frac{p_{i-1}}{\exp\left(\frac{V_i - V_{i+1}}{V_T}\right) - 1} + \frac{p_i}{\exp\left(\frac{V_{i+1} - V_i}{V_T}\right) - 1} \right) \tag{36}$$

En caso particular de que se cumpla $V_{i+1} = V_i$, se simplifica:

$$J_{p,i+1/2} = \frac{q\mu_p \cdot V_T}{\Delta x} (p_i - p_{i-1}) \tag{37}$$

Podemos combinar ambas soluciones si hacemos uso de la función de Bernoulli

$B(x) = \frac{x}{e^x - 1}$, $B(0) = 1$. Nos quedaría entonces:

$$J_{p,i+1/2} = \frac{q\mu_p \cdot V_T}{\Delta x} \left(B\left(\frac{V_{i+1} - V_i}{V_T}\right) p_i - B\left(\frac{V_i - V_{i+1}}{V_T}\right) p_{i+1} \right) \tag{38}$$

Y para la función $J_{n,i+1/2}$ se realiza el mismo procedimiento, con cuidado de respetar los signos:

$$J_{n,i+1/2} = \frac{q\mu_n \cdot V_T}{\Delta x} \left(B\left(\frac{V_{i+1} - V_i}{V_T}\right) n_{i+1} - B\left(\frac{V_i - V_{i+1}}{V_T}\right) n_i \right) \tag{39}$$

Una vez hemos obtenido la expresión correctamente discretizada para las ecuaciones de corrientes, podremos sustituirlas en las ecuaciones de continuidad.

4.2 ECUACIONES DE CONTINUIDAD

Tenemos las ecuaciones de continuidad de los portadores (5) y (6) en función de la corriente, ahora ya discretizada, y bastaría con sustituir (38) y (39) en la ecuación

correspondiente. En una primera aproximación al problema, vamos a considerar despreciable la fotogeneración ($G_L = 0$), además de considerar que nos encontramos en régimen estacionario, es decir, $\frac{\partial p}{\partial t} = 0$. Nos quedarían entonces unas ecuaciones de continuidad simplificadas:

$$\begin{aligned} 0 &= -\frac{1}{q} \frac{\partial J_p(x)}{\partial x} - U \\ 0 &= \frac{1}{q} \frac{\partial J_n(x)}{\partial x} - U \end{aligned} \quad (40)$$

Donde podemos sustituir la corriente para obtener, finalmente, nuestras ecuaciones de continuidad discretizadas.

$$\begin{aligned} U_i &= \left[\frac{V_T}{\Delta x^2} \mu_{n,i-1/2} \cdot B \left(\frac{V_{i-1} - V_i}{V_T} \right) \right] n_{i-1} \\ &- \left[\frac{V_T}{\Delta x^2} \mu_{n,i-1/2} \cdot B \left(\frac{V_i - V_{i-1}}{V_T} \right) + \frac{V_T}{\Delta x^2} \mu_{n,i+1/2} \cdot B \left(\frac{V_i - V_{i+1}}{V_T} \right) \right] n_i \\ &+ \left[\frac{V_T}{\Delta x^2} \mu_{n,i+1/2} \cdot B \left(\frac{V_{i+1} - V_i}{V_T} \right) \right] n_{i+1} \end{aligned} \quad (41)$$

$$\begin{aligned} U_i &= \left[\frac{V_T}{\Delta x^2} \mu_{p,i-1/2} \cdot B \left(\frac{V_i - V_{i-1}}{V_T} \right) \right] p_{i-1} \\ &- \left[\frac{V_T}{\Delta x^2} \mu_{p,i-1/2} \cdot B \left(\frac{V_{i-1} - V_i}{V_T} \right) + \frac{V_T}{\Delta x^2} \mu_{p,i+1/2} \cdot B \left(\frac{V_{i+1} - V_i}{V_T} \right) \right] p_i \\ &+ \left[\frac{V_T}{\Delta x^2} \mu_{p,i+1/2} \cdot B \left(\frac{V_i - V_{i+1}}{V_T} \right) \right] p_{i+1} \end{aligned} \quad (42)$$

La última ecuación que nos falta es la ecuación de Poisson, ya discretizada en (27), con la única diferencia de que ahora p_i y n_i no tienen dependencia con V .

$$f(V_i) = \frac{V_{i+2} - 2V_{i+1} + V_i}{\Delta x^2} - \frac{q}{\epsilon} (p_i - n_i + N_D - N_A) = 0 \quad (43)$$

4.3 NEWTON-RAPHSON EN NO-EQUILIBRIO

Con las ecuaciones (41), (42) y (43) tenemos un sistema de tres ecuaciones, de forma que podemos resolverlo mediante el método de Newton-Raphson para sistemas de ecuaciones que se explicó previamente en la sección 0. Obtenemos un vector V y una matriz J de esta forma:

$$V = \begin{pmatrix} f_1(n, p, V_i) \\ f_2(n, p, V_i) \\ f_3(n, p, V_i) \end{pmatrix} \quad (44)$$

$$J = \begin{pmatrix} \frac{\partial f_1}{\partial n} & \frac{\partial f_1}{\partial p} & \frac{\partial f_1}{\partial V_i} \\ \frac{\partial f_2}{\partial n} & \frac{\partial f_2}{\partial p} & \frac{\partial f_2}{\partial V_i} \\ \frac{\partial f_3}{\partial n} & \frac{\partial f_3}{\partial p} & \frac{\partial f_3}{\partial V_i} \end{pmatrix} \quad (45)$$

Siendo f_1 la ecuación de continuidad de los electrones, f_2 la de los huecos y f_3 la ecuación de Poisson. Aplicar Newton-Raphson sin más conllevaría resolver una matriz de $3N \times 3N$, que es 27 veces más costosa que nuestra matriz original de $N \times N$. Sin embargo, este sistema puede resolverse desacoplando las ecuaciones en tres sistemas más sencillos que se resuelven sucesivamente, lo que se conoce como *block successive over relaxation Newton method*, o Block-SOR-Newton [23].

$$\begin{aligned} \frac{\partial f_{1,k}}{\partial V_{i,k}} \cdot \Delta V_i &= f_{1,k}(V_k, p_k, n_k) \\ \frac{\partial f_{2,k}}{\partial p} \cdot \Delta p &= f_{2,k}(V_{k+1}, p_k, n_k), \quad \frac{\partial f_{3,k}}{\partial n} \cdot \Delta n = f_{3,k}(V_{k+1}, p_{k+1}, n_k) \end{aligned} \quad (46)$$

Es decir, resolvemos la ecuación de Poisson, utilizamos esta solución para resolver la continuidad de los huecos, y así sucesivamente. Con este método se obtiene una convergencia más lenta (lineal, en vez de cuadrática) [23], pero reduce el coste de procesamiento y simplifica su implementación.

Siguiendo el mismo procedimiento que se utilizó en la sección 3.2 solo nos queda derivar las correspondientes ecuaciones del sistema (46) para obtener las expresiones a introducir en MATLAB.

$$\begin{aligned} \frac{df_1}{dn} = & \left[\frac{V_T}{\Delta x^2} \mu_{n,i-1/2} \cdot B \left(\frac{V_{i-1} - V_i}{V_T} \right) \right] \Delta n_{i-1} \\ & - \left[\frac{V_T}{\Delta x^2} \mu_{n,i-1/2} \cdot B \left(\frac{V_i - V_{i-1}}{V_T} \right) + \frac{V_T}{\Delta x^2} \mu_{n,i+1/2} \cdot B \left(\frac{V_i - V_{i+1}}{V_T} \right) \right] \Delta n_i \\ & + \left[\frac{V_T}{\Delta x^2} \mu_{n,i+1/2} \cdot B \left(\frac{V_{i+1} - V_i}{V_T} \right) \right] \Delta n_{i+1} \end{aligned} \quad (47)$$

$$\begin{aligned} \frac{\partial f_2}{\partial p} = & \left[\frac{V_T}{\Delta x^2} \mu_{p,i-1/2} \cdot B \left(\frac{V_i - V_{i-1}}{V_T} \right) \right] \Delta p_{i-1} \\ & - \left[\frac{V_T}{\Delta x^2} \mu_{p,i-1/2} \cdot B \left(\frac{V_{i-1} - V_i}{V_T} \right) + \frac{V_T}{\Delta x^2} \mu_{p,i+1/2} \cdot B \left(\frac{V_{i+1} - V_i}{V_T} \right) \right] \Delta p_i \\ & + \left[\frac{V_T}{\Delta x^2} \mu_{p,i+1/2} \cdot B \left(\frac{V_i - V_{i+1}}{V_T} \right) \right] \Delta p_{i+1} \end{aligned} \quad (48)$$

Para derivar Poisson (43) tenemos que hacer uso de la propiedad (9) para obtener

$$\frac{\partial p}{\partial V} = \frac{\partial p}{\partial x} \frac{\partial x}{\partial V} \text{ y } \frac{\partial n}{\partial V} = \frac{\partial n}{\partial x} \frac{\partial x}{\partial V}.$$

$$\frac{df_3}{dn} = \left[\frac{1}{\Delta x^2} \right] \Delta V_{i-1} + \left[\frac{-2}{\Delta x^2} + \frac{q}{\varepsilon} \frac{n_i + p_i}{V_T} \right] \Delta V_i + \left[\frac{1}{\Delta x^2} \right] \Delta V_{i+1} \quad (49)$$

4.4 CONSIDERACIONES PRÁCTICAS PARA LA IMPLEMENTACIÓN DEL MÉTODO EN MATLAB

Para realizar esta extensión en no equilibrio vamos a programar otro bucle inmediatamente después del original, de forma que nuestra solución V en Newton-Raphson en equilibrio servirá como la solución inicial V_0 para el *Block-SOR-Newton* fuera del equilibrio.

```
Dn_min = Vt*mun;  
Dn_plus = Dn_min;  
Dp_min = Vt*mup;  
Dp_plus = Dp_min;  
  
Vint = V;  
V(m) = V(m)+per;
```

Fig. 12 Definición de coeficientes de difusión en función de la movilidad y de la perturbación en el voltaje para sacar el sistema del equilibrio.

Vamos a definir los coeficientes de difusión D_{min} y D_{max} , que en primera aproximación no vamos a diferenciar, además de introducir una pequeña perturbación en el voltaje per para sacar nuestra unión del equilibrio. Es recomendable que esta perturbación sea pequeña (alrededor de $3V_T$) para asegurar la convergencia de la solución. Si se desea una perturbación mayor, siempre se puede repetir el proceso con la nueva solución, añadiendo la perturbación de forma incremental.

La resolución del sistema de Newton-Raphson con las ecuaciones f_1 , f_2 y f_3 es idéntica que en el apartado 3.3.4:

- Se define la función, los coeficientes de la matriz correspondiente, y se resuelve el sistema para obtener ΔF .
- En primer lugar, se resuelve la continuidad de ambos portadores.
- Y es esta solución la que se utiliza para obtener el nuevo potencial, siendo la continuidad de los huecos análoga y la ecuación de Poisson igual que en el equilibrio, a excepción de la matriz de derivadas que se forma con los coeficientes obtenidos en (49).

```
% Continuidad de electrones
an = zeros(m,1);    bn = zeros(m,1);    cn = zeros(m,1);
for i = (2:m-1)
an(i) = (Dn_min/dx2)*Ber(phi(i-1)-phi(i));
bn(i) = (Dn_plus/dx2)*Ber(phi(i)-phi(i+1)) +
(Dn_min/dx2)*Ber(phi(i)-phi(i-1));
cn(i) = (Dn_plus/dx2)*Ber(phi(i+1)-phi(i));
end
an(1) = an(2);    an(m) = an(m-1);    bn(1) = bn(2);
bn(m) = bn(m-1);    cn(1) = cn(2);    cn(m) = cn(m-1);

FN = zeros(m-2,1);
for i = (1:m-2)
FN(i) = an(i+1)*n(i)-bn(i+1)*n(i+1)+cn(i+1)*n(i+2);
end

dup = sparse(diag(cn(2:m-2),1));
d = sparse(diag(bn(2:m-1)));
down = sparse(diag(an(3:m-1),-1));
clear an bn cn
MDN = dup-d+down;
clear d dup down

DN=-MDN\FN;

n(2:m-1)=n(2:m-1)+DN;
n(1) = n(2);
n(m) = n(m-1);
```

Fig. 13 Implementación de la función $f_1(n)$, la matriz de derivadas $\frac{\partial f_1}{\partial n}$, y la obtención del vector de corrección para la ecuación de continuidad de los electrones.

5 CASOS PRÁCTICOS DE APLICACIÓN: COMPARATIVA CON LA APROXIMACIÓN CLÁSICA DE EMPOBRECIMIENTO

Para ilustrar el funcionamiento de nuestro simulador vamos a examinar los resultados para distintas distribuciones de dopaje, tanto en equilibrio como fuera del mismo. En primer lugar, realizaremos una simulación con un dopaje $N_A = N_D = 10^{16}$, y en segundo lugar otra con un dopaje $N_A > N_D$, siendo $N_A = 10^{18}$ y $N_D = 10^{16}$. El resto de las constantes físicas se han escogido de forma que representen el semiconductor de silicio a 300K.

5.1 DOPAJE SIMÉTRICO – EQUILIBRIO

Al ser el dopaje idéntico en ambos lados de la unión, podemos esperar ver una situación simétrica para la distribución de carga y de potencial, además de un nivel de Fermi constante. La simulación completa tuvo una duración de 0.035 segundos y 22 iteraciones.

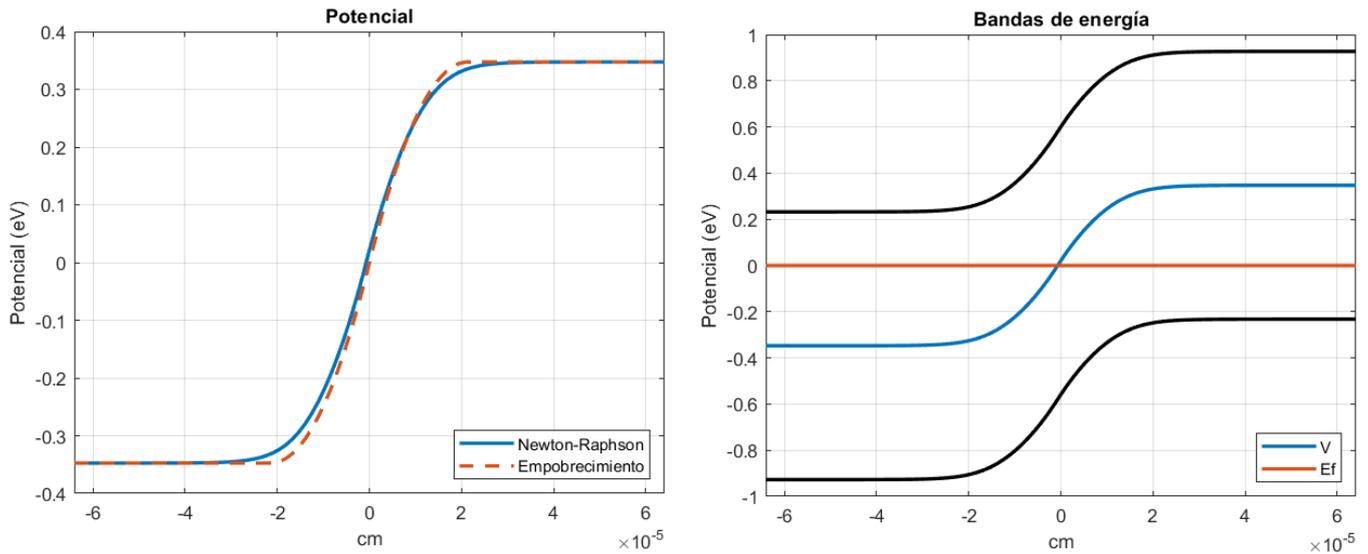


Fig. 14 Potencial obtenido mediante Newton-Raphson (línea azul) y estimación inicial de empobrecimiento (línea naranja) para $N_A = N_D = 10^{16}$.

Fig. 15 Diagrama de bandas (Potencial V (azul) y nivel de Fermi (naranja) junto a los niveles de conducción y valencia (negro)) para $N_A = N_D = 10^{16}$.

Si comparamos el potencial que hemos obtenido con métodos numéricos frente a la aproximación de empobrecimiento vemos que la diferencia entre ambas es bastante pequeña, siendo la mayor desviación de $0.0263V$ (7.45%) en los extremos de la zona de empobrecimiento.

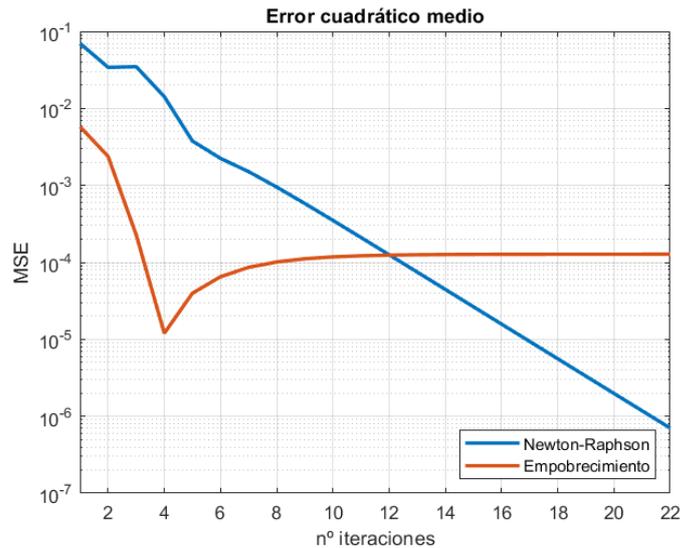


Fig. 16 MSE de la solución de Newton-Raphson (línea azul) y MSE de la diferencia con la aproximación de empobrecimiento (línea naranja) para $N_A = N_D = 10^{16}$.

Podemos representar el error cuadrático medio (MSE) de la solución de Newton-Raphson frente al número de iteraciones para comprobar su evolución. Vemos que el error se reduce a la mitad cada iteración (a excepción de la iteración número 3, en la que el método introdujo una sobrecorrección que luego compensa), y que a partir de la iteración 12 ya se obtiene una solución mejor que la de empobrecimiento.

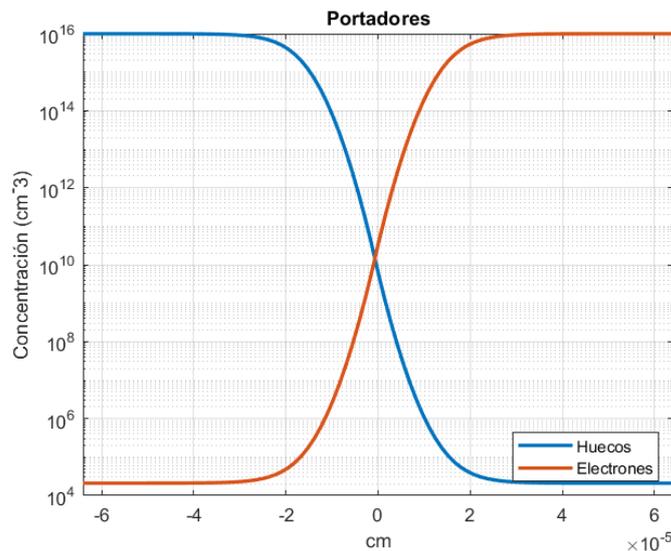


Fig. 17 Densidad de carga para $N_A = N_D = 10^{16}$.

En esta ocasión vemos que la aproximación de empobrecimiento para la densidad de carga es adecuada para el área que corresponde a las cercanías de la unión

metalúrgica, pero no es lo suficientemente precisa en los extremos de la zona de empobrecimiento.

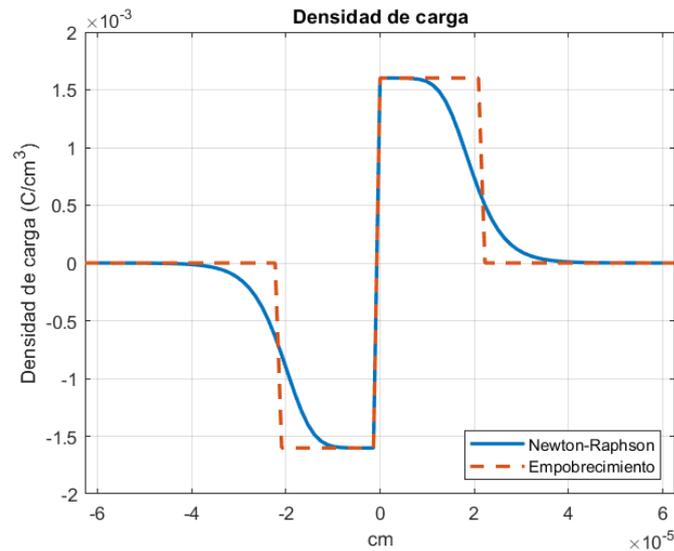


Fig. 18 Concentración de portadores a lo largo de la unión PN para $N_A = N_D = 10^{16}$.

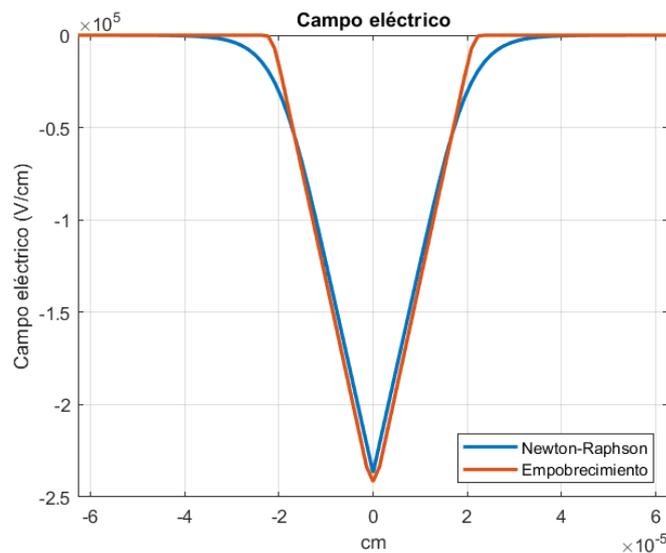


Fig. 19 Campo eléctrico en la unión PN para $N_A = N_D = 10^{16}$.

La aproximación de empobrecimiento también parece efectiva para calcular el campo eléctrico a través de la unión. Obtenemos un valor un poco menor utilizando el método Newton-Raphson, con una diferencia de $0.12765 \cdot 10^5 \text{ V/cm}$ o un error máximo del 5.58%.

5.2 DOPAJE ASIMÉTRICO – EQUILIBRIO

Para un dopaje asimétrico $N_A = 10^{18}$ y $N_D = 10^{16}$, la simulación ha tomado 18 iteraciones y 0.407 segundos.

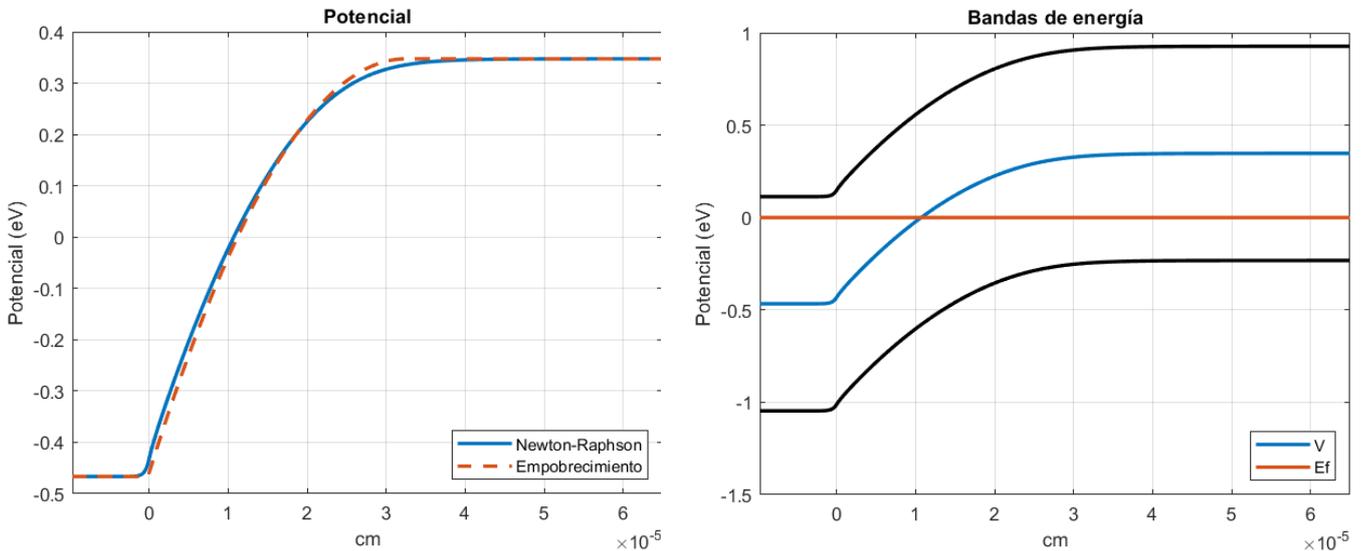


Fig. 20 Potencial obtenido mediante Newton-Raphson (línea azul) y estimación inicial de empobrecimiento (línea naranja) para $N_A = 10^{18}$ y $N_D = 10^{16}$.

Fig. 21 Diagrama de bandas (Potencial V (azul) y nivel de Fermi (naranja) junto a los niveles de conducción y valencia(negro)) para $N_A = 10^{18}$ y $N_D = 10^{16}$.

De nuevo comprobamos que la aproximación de empobrecimiento es bastante efectiva incluso para distribuciones de dopaje asimétricas, si bien se obtiene un resultado más preciso con Newton-Raphson. El mayor error obtenido es de $0.0334V$, o un error del 9.1%.

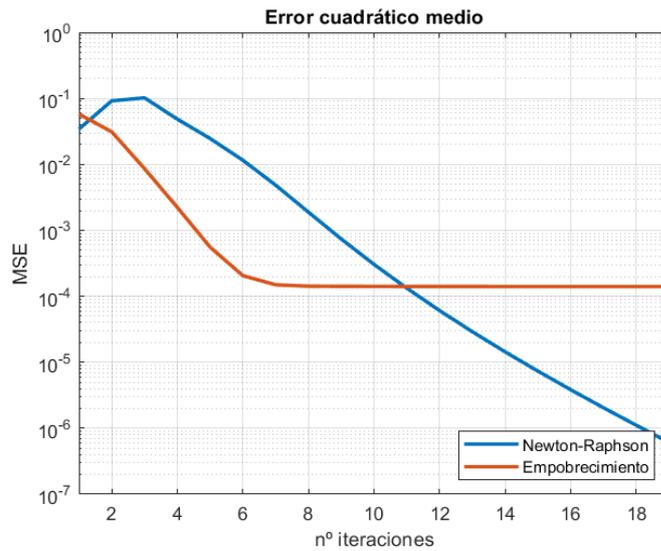


Fig. 22 Error cuadrático medio de la solución de Newton-Raphson (línea azul) y error cuadrático medio de la diferencia con la aproximación de empobrecimiento (línea naranja) para $N_A = 10^{18}$ y $N_D = 10^{16}$.

De manera análoga al dopaje simétrico, vemos que el error de la solución de Newton-Raphson se reduce con cada iteración, y que a partir de la iteración 11 tenemos una solución mejor que la de empobrecimiento.

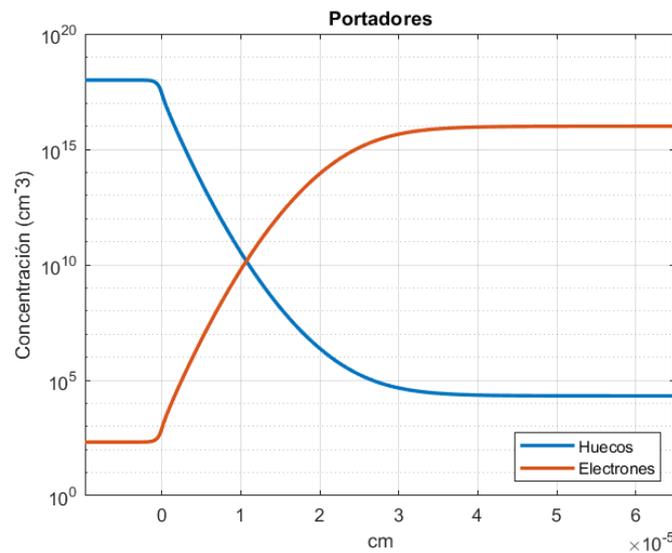


Fig. 23 Concentración de portadores a lo largo de la unión PN para $N_A = 10^{18}$ y $N_D = 10^{16}$.

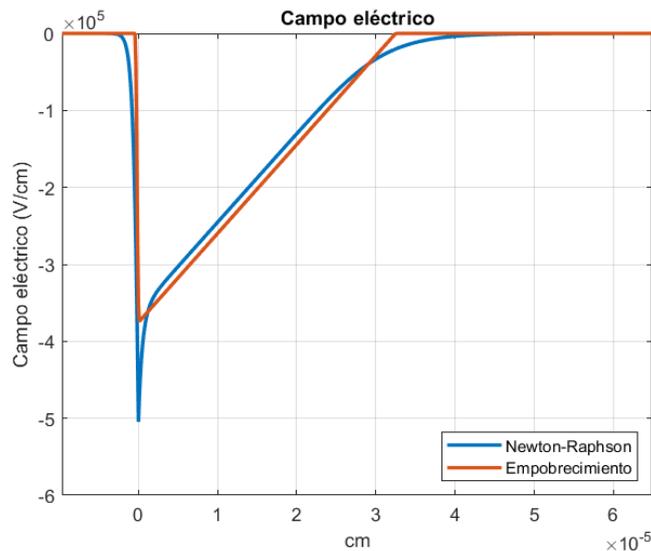


Fig. 24 Campo eléctrico obtenido mediante Newton-Raphson (línea azul) y estimación inicial de empobrecimiento (línea naranja) para $N_A = 10^{18}$ y $N_D = 10^{16}$.

Aquí se ve que, si bien ambos modelos han sido capaces de estimar las zonas de empobrecimiento correctamente, los valores obtenidos para el campo eléctrico son distintos, siendo el obtenido mediante Newton-Raphson mucho más definido. La diferencia entre ambos es de $2.9183 \cdot 10^5$ V/cm, un error relativo de casi un 60%.

5.3 DOPAJE SIMÉTRICO – NO-EQUILIBRIO

Realizamos la misma simulación que en el apartado 5.1, con la diferencia de que vamos a añadir una perturbación en forma de $V_{per} = 3V_T$ en el extremo n de la unión. Obtenemos los resultados tras 0.095 segundos y 24 iteraciones.

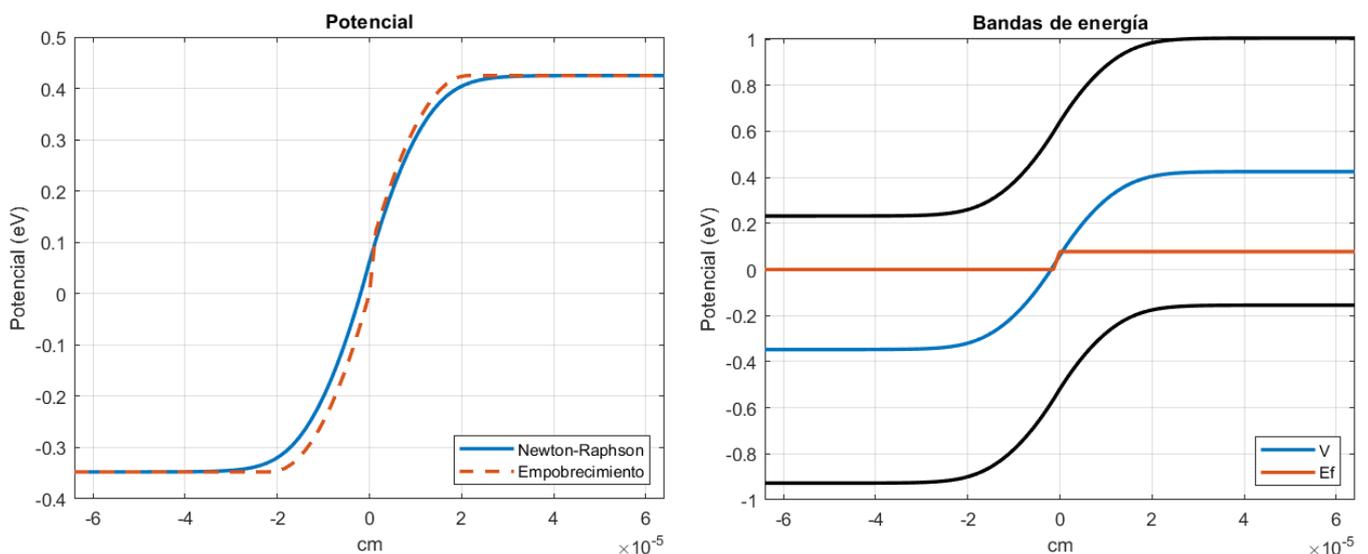


Fig. 25 Potencial obtenido mediante Newton-Raphson (línea azul) y estimación inicial de empobrecimiento (línea naranja) para $N_A = N_D = 10^{16}$.

Fig. 26 Diagrama de bandas (Potencial V (azul) y nivel de Fermi (naranja) junto a los niveles de conducción y valencia(negro)) para $N_A = N_D = 10^{16}$.

En este caso se aprecia la curvatura del nivel de Fermi, debido a la ausencia de equilibrio. La diferencia entre la aproximación de empobrecimiento tiene una desviación máxima de $0.0615V$ en el centro de la unión.

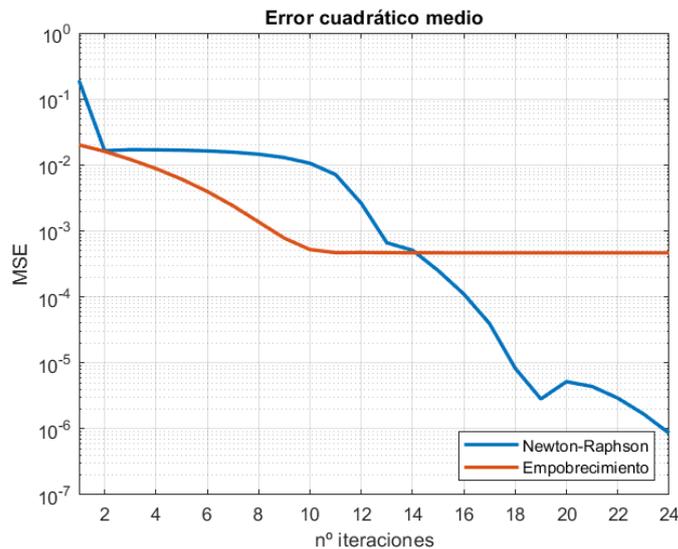


Fig. 27 Error cuadrático medio de la solución de Newton-Raphson (línea azul) y error cuadrático medio de la diferencia con la aproximación de empobrecimiento (línea naranja) para $N_A = N_D = 10^{16}$.

Al introducir la perturbación, el método de Newton-Raphson tiende a exagerar la curva para alcanzar el nuevo potencial en las primeras iteraciones, de forma que su error permanece por encima de la aproximación de empobrecimiento hasta alcanzar la iteración 14. Se aprecia, sobre todo al principio de la simulación, la principal desventaja del método de *Block-SOR-Newton*, que nos proporciona una convergencia algo más lenta.

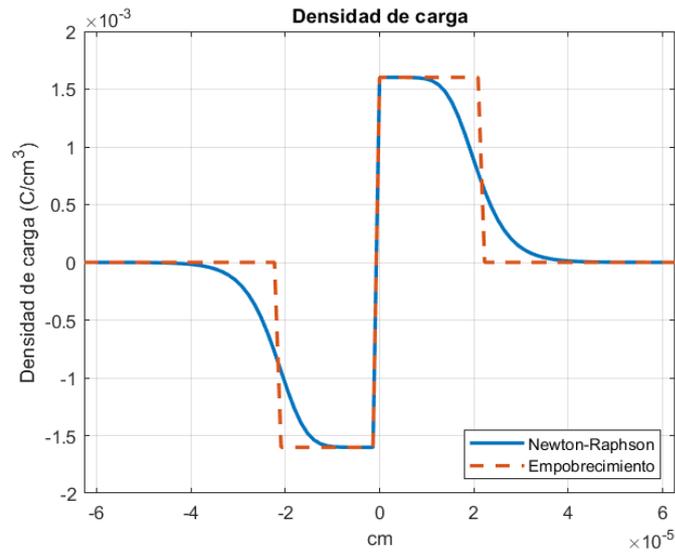


Fig. 28 Densidad de carga para $N_A = N_D = 10^{16}$.

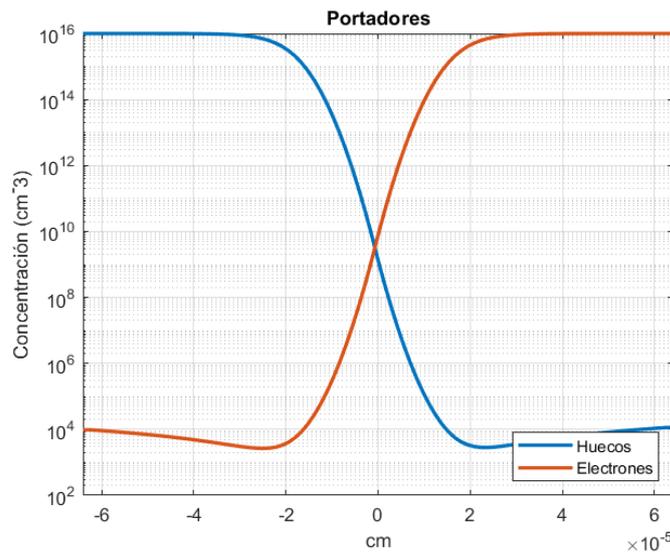


Fig. 29 Concentración de portadores a lo largo de la unión PN para $N_A = N_D = 10^{16}$.

La distribución de portadores es parecida a la obtenida en equilibrio, pero sí que podemos apreciar un pequeño incremento de los portadores minoritarios en los extremos de la unión.

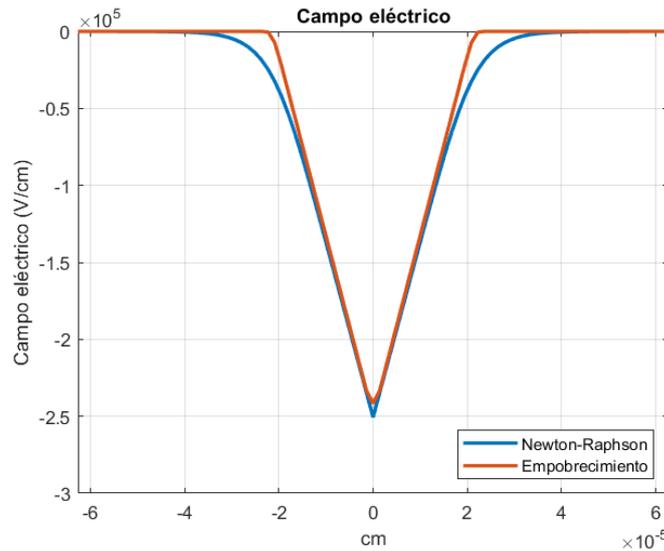


Fig. 30 Campo eléctrico en la unión PN para $N_A = N_D = 10^{16}$.

La diferencia entre el máximo de la aproximación de empobrecimiento y el máximo obtenido con Newton-Raphson es bastante pequeña, de 9026 V/cm, lo que nos deja con un error relativo del 3,6%. Se puede apreciar que el campo obtenido con Newton-Raphson muestra una zona de empobrecimiento mucho menos definida, lo que se ajusta más a una situación real, y no ideal, de la unión.

5.4 DOPAJE ASIMÉTRICO – NO-EQUILIBRIO

Vamos a considerar un dopaje asimétrico de $N_A = 10^{17}$ y $N_D = 10^{16}$, y la misma perturbación que el apartado anterior, de $V_{per} = 3V_T$. La simulación ha tomado 79 iteraciones y 0.4075 segundos.

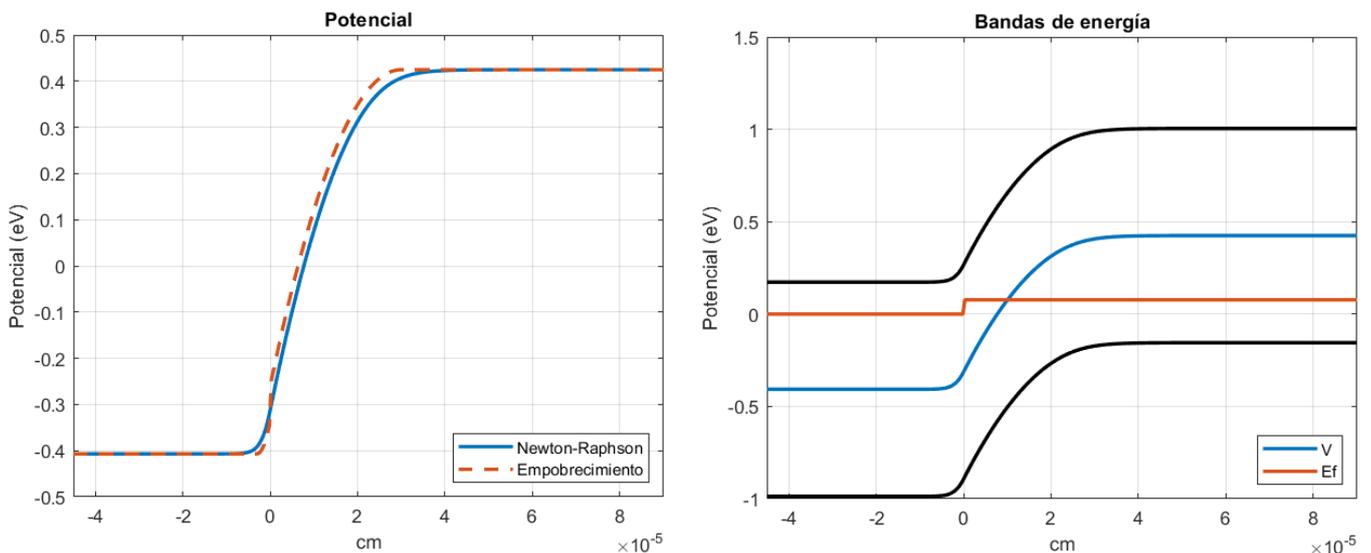


Fig. 31 Potencial obtenido mediante Newton-Raphson (línea azul) y estimación inicial de empobrecimiento (línea naranja) para $N_A = 10^{17}$ y $N_D = 10^{16}$.

Fig. 32 Diagrama de bandas (Potencial V (azul) y nivel de Fermi (naranja) junto a los niveles de conducción y valencia(negro)) para $N_A = 10^{17}$ y $N_D = 10^{16}$.

De nuevo la aproximación de empobrecimiento no es capaz de estimar correctamente el voltaje en la unión, ya que tiene una desviación máxima de $0.0522V$ en $x = 0$. Se aprecia también el nivel de Fermi, que no es constante.

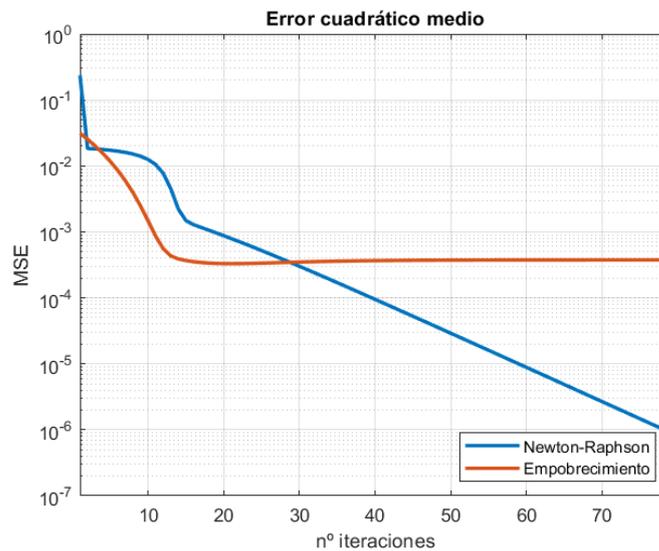


Fig. 33 Error cuadrático medio de la solución de Newton-Raphson (línea azul) y error cuadrático medio de la diferencia con la aproximación de empobrecimiento (línea naranja) para $N_A = 10^{17}$ y $N_D = 10^{16}$.

Si bien la convergencia en este caso ha sido considerablemente más lenta que en los ejemplos previos, vemos que con 29 iteraciones ya obtenemos un resultado mejor que con la aproximación clásica de empobrecimiento.

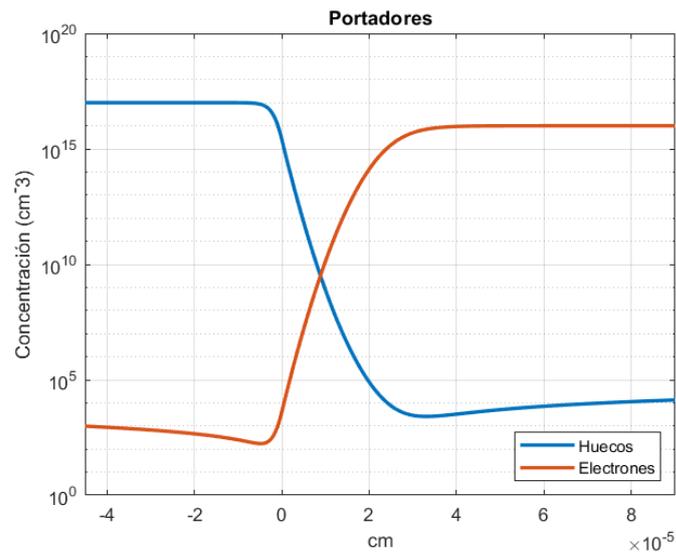


Fig. 34 Concentración de portadores a lo largo de la unión PN para $N_A = 10^{17}$ y $N_D = 10^{16}$.

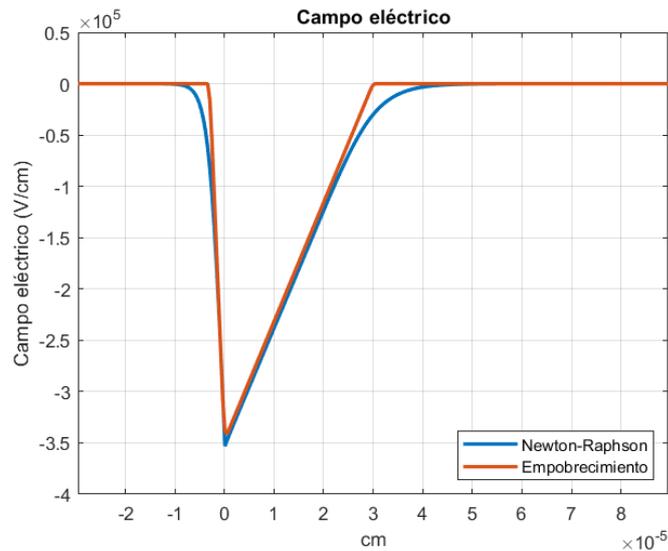


Fig. 35 Campo eléctrico obtenido mediante Newton-Raphson (línea azul) y estimación inicial de empobrecimiento (línea naranja) para $N_A = 10^{17}$ y $N_D = 10^{16}$.

Del mismo modo que el apartado anterior, obtenemos una distribución de portadores más realista, y un campo eléctrico similar. El error obtenido en el máximo del campo es del 4.12%, o una desviación de $0.1454 \cdot 10^5$ V/cm .

5.5 OTROS FENÓMENOS DE INTERÉS

Finalmente, para ilustrar el potencial del modelo construido, vamos a analizar cómo se implementarían otros fenómenos electrónicos como la fotogeneración, distintos dopajes (unión gradual), y dependencia de μ con la temperatura, el número de portadores, y el campo eléctrico.

5.5.1 Fotogeneración

En el desarrollo de las ecuaciones de continuidad para fuera del equilibrio se utilizaron ecuaciones simplificadas (40) que consideraban despreciable la fotogeneración G_L . Sin embargo, este efecto puede incluirse en nuestras ecuaciones, discretizarlas con esta nueva variable, y sustituirlas en nuestro desarrollo actual.

La fotogeneración [6]-[24] es el proceso mediante el cual se generan pares electrón-hueco debido a la energía incidente de un haz de fotones. Si esta energía es mayor que la de la banda prohibida, E_G , es absorbida por los electrones de la banda de valencia y promocionan a la banda de conducción, generando el par electrón-hueco. En general G_L dependerá de la distancia que la luz penetra en el semiconductor y de la longitud de onda de la luz incidente. Sin embargo, podemos considerar G_L constante para representar una iluminación uniforme del semiconductor.

$$G_L - U = \frac{1}{q} \frac{\partial J_n}{\partial x} \quad (50)$$

Es decir, que a las ecuaciones (41) y (42) solo hay que añadirles la constante G_L . Sus derivadas permanecen iguales, ya que hemos definido la fotogeneración de forma que no dependa de ninguna variable, así que solo tenemos que introducir la siguiente modificación en el código tanto para la continuidad de los electrones como la de los huecos:

```
G_L = GL*ones(m-2,1)

FN = zeros(m-2,1);
for i = (1:m-2)
FN(i) = an(i+1)*n(i)-bn(i+1)*n(i+1)+cn(i+1)*n(i+2)-G_L(i);
end
```

Fig. 36 Modificación necesaria para añadir el efecto de fotogeneración con G_L constante.

De la misma manera que estamos creando el vector G_L como constante, también sería posible modificarlo para añadir dependencia con x (iluminación irregular) definiendo cada punto con un bucle *for*. Por ejemplo, si la iluminación fuera distinta para la zona tipo p y tipo n:

```
for i = 1:m-2
    if x(i) < 0
        G_L(i) = GLp
    else
        G_L(i) = GLn
    end
end
```

Fig. 37 Modificación necesaria para añadir efecto fotoeléctrico con G_L constante distinto para cada parte del semiconductor.

5.5.2 Unión gradual

También podemos considerar perfiles de dopado graduales (Figura 8) si, después de definir el dopaje de la forma que se especifica en el apartado 3.3.3, que equivale a una unión abrupta, definimos una variable conjunta $dop = N_D - N_A$:

```
xtp=Wp/3; xtn=Wn/3;

dop = ND - NA;
for i = 1:m
    if x(i) > -xtp && x(i)<0
        dop(i) = x(i)*Na/xtp;
    elseif x(i) < xtn && x(i)>=0
        dop(i) = x(i)*Nd/xtn;
    end
end
```

Fig. 38 Modificación del dopaje para considerar un dopado lineal en una región de transición $(-x_{tp}, x_{tn})$.

De esta manera obtenemos un dopaje con una pendiente lineal en la región de transición $(-x_{tp}, x_{tn})$, que hemos definido inicialmente como 1/3 del ancho de la región de empobrecimiento. Es importante señalar que será necesario sustituir $(N_D - N_A)$ por *dop* en la ecuación de Poisson.

5.5.3 Dependencia de μ

Hemos considerado una movilidad constante, pero en realidad este valor depende de la temperatura, el dopado, y el campo eléctrico. Podemos definir la movilidad de los electrones como [21]:

$$\mu_n = \frac{q \cdot \tau_c}{m_e} \quad (51)$$

, siendo τ_c el tiempo medio entre colisiones y m_e la masa efectiva. A mayor dopado existirá un menor tiempo entre colisiones, ya que la probabilidad de interactuar con alguna impureza será mayor. La dependencia con la temperatura se hace evidente tanto a bajas temperaturas (donde domina el mecanismo de dispersión por impurezas) como a altas temperaturas (la agitación térmica de los átomos de la red cristalina disminuye el tiempo entre colisiones).

A la hora de obtener un valor de movilidad adecuado se suele recurrir a un ajuste empírico [25], tomando valores experimentales de referencia y ajustándolos mediante la siguiente fórmula:

$$\mu = \mu_{\min} + \frac{\mu_0}{1 + (N / N_{ref})^\alpha} \quad (52)$$

Siendo N el dopaje de cada sección del semiconductor (N_A en la parte tipo p y N_D en la tipo n), y el resto de parámetros (μ_{\min} , μ_0 , N_{ref} y α) son los parámetros de ajuste. Estos parámetros, que se toman como referencia a 300K, se podrán modificar para la temperatura mediante la siguiente relación:

$$A = A_0 \left(\frac{T}{300K} \right)^\eta \quad (53)$$

Es decir, que utilizando los valores de referencia (Tabla 1) e implementando las fórmulas (52) y (51) en el código, se podría obtener una movilidad más exacta.

Parámetro	Valores		η
	Electrones	Huecos	
$N_{ref}(cm^{-3})$	$1.30 \cdot 10^{17}$	$2.35 \cdot 10^{17}$	2.4
$\mu_{min}(cm^2/V \cdot s)$	92	54.3	-0.57
$\mu_0(cm^2/V \cdot s)$	1628	406.9	-2.33(electrones) y -2.23(huecos)
α	0.91	0.88	-0.146

Tabla 1 Parámetros de ajuste para la movilidad (Si a 300K) [25].

6 CONCLUSIONES

A continuación, se hace un sumario de las conclusiones principales de este Trabajo Fin de Grado:

- Estudio bibliográfico: se ha realizado un estudio de la bibliografía en el campo de modelado de uniones P-N. Como conclusión principal, se ha puesto de manifiesto la necesidad de desarrollar un método que aúne la sencillez y la intuición en los resultados de los métodos de bajo nivel, con la flexibilidad y soluciones más realistas de las aproximaciones numéricas 3-D que proporcionan las herramientas de TCAD. Para abordarlo, se decide recurrir a métodos numéricos avanzados [8]-[9] para obtener una resolución de la unión P-N que nos permita el estudio de fenómenos fuera del equilibrio y de efectos dependientes de la concentración de portadores.
- Estudio de la unión P-N: se han estudiado las ecuaciones que definen el sistema tales como la ecuación de Poisson (12) que define las variaciones del campo eléctrico en función de la concentración de carga espacial, como las ecuaciones de continuidad (5) y (6), que modelan el transporte de dichos portadores a lo largo de la unión [2]-[3].

Como parte del análisis se han incluido en la memoria una introducción a los distintos procesos de generación y recombinación, como pueden ser la fotogeneración o el tiempo de vida medio [4].

Para concluir, se ha presentado la solución clásica de empobrecimiento de portadores en las cercanías de la superficie de contacto de la unión, llegando a expresiones para el nivel intrínseco de energía (11) integrables en el espacio y que permiten aproximar tanto el campo eléctrico como la concentración de portadores.

- Estudio e implementación en MATLAB de métodos numéricos avanzados para la resolución de ecuaciones diferenciales:
 - Se han analizado las ventajas y desventajas del método de Newton-Raphson [8]-[9], y se ha llegado a la conclusión de que se trata de un método robusto y con una alta convergencia, pero que requiere de una estimación inicial adecuada y de la necesidad de obtener la derivada de la función a estudiar. Para solventar estos problemas, se ha considerado las siguientes estrategias:
 - Se toma como estimación inicial la aproximación de empobrecimiento (que, como hemos demostrado, resulta muy adecuada en condiciones de equilibrio).
 - Se discretizan las ecuaciones obtenidas para obtener sus derivadas.
 - La discretización se realiza siguiendo la aproximación de derivadas en diferencias finitas [13]-[14], donde se ha tenido que discretizar la variable V_i , aplicarla a la ecuación de Poisson (27), y derivarla para alcanzar los requisitos necesarios para aplicar el método de Newton-Raphson.
- Aplicación e implementación de los métodos de estudiados a la resolución de la unión P-N en condiciones de equilibrio en el entorno de MATLAB:
 - Los estudios y análisis previos se han plasmado en la implementación en MATLAB de procedimientos de análisis de la unión P-N.

- Como punto de partida se consideró inicialmente la aproximación en [15]. Sin embargo, este artículo utiliza una ecuación de Poisson simplificada que no evalúa la concentración de portadores, sino que la sustituye para obtener una ecuación con una sola variable. Nosotros hemos desarrollado la ecuación de Poisson (27) completa, de forma que tenemos que implementar nuevas variables, p y n , y sustituir las ecuaciones en el método. Además, si bien en el artículo se toma un parámetro de mallado fijo, en nuestro desarrollo se ha tenido en cuenta la necesidad de calcular un mallado a partir de la longitud de Debye, para poder apreciar correctamente el efecto de interacción entre las cargas.
- Extensión de los procedimientos desarrollados para la resolución de la unión P-N en condiciones de no-equilibrio:
 - A la hora de modificar nuestro modelo para la inclusión de efectos fuera del equilibrio, se ha tomado como guía el libro [16], que realiza un análisis de la unión P-N tanto dentro como fuera del equilibrio.
 - Se ha propuesto el método de Scharfetter-Gummel [18]-[19] para discretizar las ecuaciones de corriente, lo que nos deja con dos ecuaciones diferenciales no lineales que hemos resuelto mediante técnicas de métodos matemáticos [22] como la aplicación de un factor integrante, que también se ha tenido que calcular.
 - Una vez resuelta la discretización de la corriente, se ha sustituido en las ecuaciones de continuidad (41) y (42), y se procede a la implementación en MATLAB.
 - Las principales diferencias con la realización de partida en el libro [16] se pueden resumir en: 1) en nuestra implementación se utiliza Matlab en lugar de Fortran, 2) como técnica de resolución del sistema de ecuaciones se considera un método adaptativo basado en Newton-Raphson, frente a la implementación original más lenta utilizando la descomposición LU de la matriz.
- Análisis y discusión de resultados:

- Se demuestra que la aproximación de empobrecimiento es adecuada como condición inicial, y que en general es capaz de predecir adecuadamente el potencial en la unión, así como las zonas empobrecidas y la distribución de portadores, pero resulta inexacta a la hora de calcular el campo eléctrico para dopajes asimétricos o insuficiente a la hora de representar efectos causados por perturbaciones o interacciones de portadores. Nuestro modelo, desarrollado mediante las ecuaciones de Poisson y continuidad resueltas con el método de Newton-Raphson ofrece una solución numérica más exacta en pocas iteraciones y en menos de un segundo, que puede ajustarse mediante la modificación de la tolerancia para hacerla más o menos precisa dependiendo de la necesidad correspondiente.
- Se ha mostrado el funcionamiento del simulador utilizando dos configuraciones de dopaje, una simétrica y otra asimétrica, y en dos estados distintos: equilibrio y no-equilibrio con una perturbación específica.
- También se ha sugerido un desarrollo y modificaciones del código para analizar otros casos de interés, como dopajes graduales, efectos de fotogeneración, y dependencia de la movilidad con otros parámetros.
- Trabajo futuro:
 - Se deja la puerta abierta a mejorar el modelo, ya que tanto la solución en equilibrio como de no equilibrio pueden expandirse con pequeñas modificaciones en las ecuaciones a utilizar, mostradas en el apartado 5.5, utilizando como base la resolución ya obtenida con Newton-Raphson.
 - Al depender el modelo de las ecuaciones de continuidad completas y ofrecer una solución altamente precisa, es posible modelar efectos más complejos si se realiza el desarrollo correspondiente.

7 REFERENCIAS

- [1] Riordan, M., & Hoddeson, L. (1997). *Origins of the pn junction*. IEEE Spectrum, 34(6), 46-51.
- [2] Tsividis Y., McAndrew C. (2011). *Operation and Modeling of the MOS Transistor* (3th ed). Oxford University Press.
- [3] Samar K. Saha. (2020). *FinFET Devices for VLSI Circuits and Systems* (1st ed.). CRC Press.
- [4] A. S. Grove. (1967). *Physics and Technology of Semiconductor Devices*. John Wiley & Sons.
- [5] C.-t. Sah, R. N. Noyce and W. Shockley. (1957). *Carrier Generation and Recombination in P-N Junctions and P-N Junction Characteristics*. Proceedings of the IRE, 45(9), 1228-1243.
- [6] J. Piprek. (2003). *Semiconductor Optoelectronic Devices: Introduction to Physics and Simulation*. Academic Press.
- [7] Ferguson, R. C. (1996). *Numerical techniques for the drift-diffusion semiconductor equations*. University of Bath.
- [8] Chapra, S. (2005). *Numerical methods for engineers* (5th ed). (R. Canale, Ed.) Boston: McGraw-Hill.
- [9] Gil, A. (2007). *Numerical methods for special functions* (J. Segura, & N. Temme, Edits.) Philadelphia (Pennsylvania): Society for Industrial and Applied Mathematics.
- [10] Multiphysics, C. (1998). *Introduction to COMSOL Multiphysics extregistered*. COMSOL Multiphysics, Burlington, MA.
- [11] Synopsys, Inc. (2009). *Sentaurus Device Reference Manual* (6th ed). Mountain View, CA.
- [12] Pierret, R. (2006). *Semiconductor device fundamentals: with computer-based exercises and homework problems*. New Delhi, India: Pearson.
- [13] Patil M., Ravaoli U., Kerkhoven T. (1998). *Numerical Evaluation of Iterative Schemes for Drift-diffusion Simulation*. VLSI Design.
- [14] Shafiro, F. (1995). *The numerical solution of Poisson's equation in a pn diode using a spreadsheet*. IEEE Transactions on Education, (1995), 380-384.
- [15] Jabr R., Hamad M., Mohanna Y. (2007). *Newton-Raphson Solution of Poisson's Equation in a Pn Diode*. International Journal of Electrical Engineering & Education, (2007), 23-33.
- [16] Vasileska, D., Goodnick, S.M., & Klimeck, G. (2010). *Computational Electronics: Semiclassical and Quantum Device Modeling and Simulation* (1st ed.). CRC Press.
- [17] The MathWorks, Inc. (2021). *Deep Learning Toolbox: User's Guide (r2018a)*.
- [18] D. L. Scharfetter and H. K. Gummel. (1969). *Large-signal analysis of a silicon Read diode oscillator*. IEEE Transactions on Electron Devices, (1969), 64-77.
- [19] Paul A. Farrel, Eugene C. Gartland. (1991). *On the Scharfetter-Gummel discretization for drift-diffusion continuity equations*. Computational Methods for Boundary and Interior Layers in Several Dimensions, John J. H. Miller.
- [20] Ben G. Streetman, Sanjay Banerjee (2015). *Solid State Electronic Devices, Global Edition*. Pearson.

- [21] J. P. Colinge, C. A. Colinge. (2002). *Physics of Semiconductor Devices*. Springer New York, NY.
- [22] Juan M. Aguirregabiria (2000). *Ecuaciones diferenciales ordinarias para estudiantes de Física*. Servicio Editorial de la Universidad del País Vasco.
- [23] Selberherr, S. (1984). *Analysis and Simulation of Semiconductor Devices*. Vienna: Springer Vienna.
- [24] Pierret. (1994). *Fundamentos de semiconductores* (2a ed.). Addison-Wesley Iberoamericana,.
- [25] Robert F. Pierret. (1987). *Advanced semiconductor fundamentals*. Addison-Wesley Longman Publishing Co., Inc., USA.

8 ANEXO I

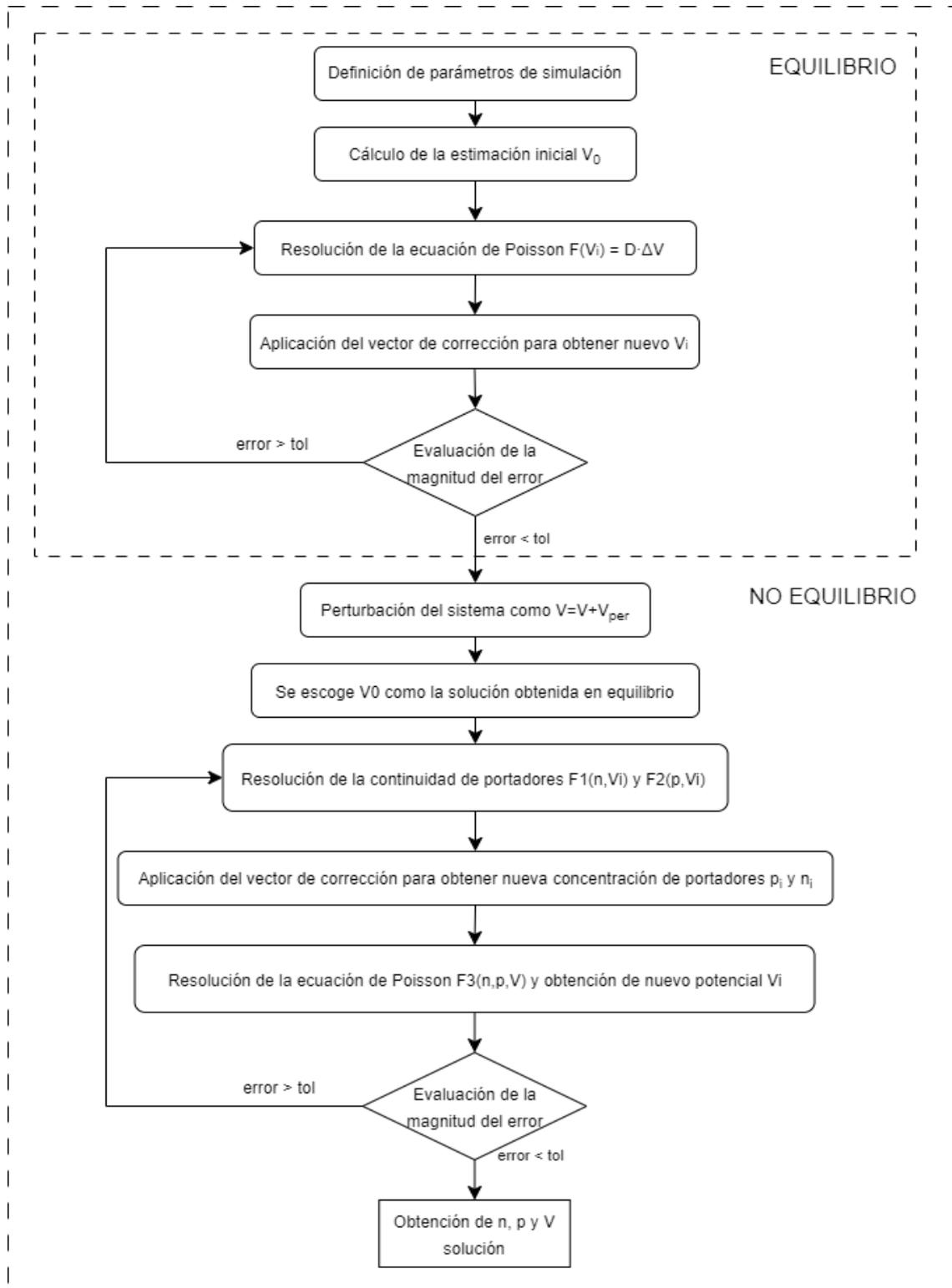


Fig. 39 Diagrama de flujo de implementación del algoritmo Newton-Raphson para la resolución de la unión P-N.