

UNIVERSIDADE DE SANTIAGO DE COMPOSTELA
DEPARTAMENTO DE ELECTRÓNICA E COMPUTACIÓN



TESIS DOCTORAL

DESARROLLO DE HERRAMIENTAS PARA LA
SIMULACIÓN DE TRANSISTORES MOSFET

Presentada por:
Manuel Antonio Aldegunde Rodríguez

Dirigida por:
Antonio Jesús García Loureiro

Santiago de Compostela, Marzo de 2009

Dr. **Antonio Jesús García Loureiro**, Profesor Contratado Doctor del Área de Electrónica de la Universidad de Santiago de Compostela

HACE CONSTAR:

Que la memoria titulada **Desarrollo de herramientas para la simulación de transistores MOSFET** ha sido realizada por D. **Manuel Antonio Aldegunde Rodríguez** bajo mi dirección en el Departamento de Electrónica e Computación de la Universidad de Santiago de Compostela, y constituye la Tesis que presenta para optar al grado de Doctor en Ciencias Físicas.

Santiago de Compostela, Marzo de 2009

Fdo: **Antonio Jesús García Loureiro**
Director de la tesis

Fdo: **Javier Díaz Bruguera**
Director del Departamento de
Electrónica y Computación

Fdo: **Manuel Antonio Aldegunde Rodríguez**
Autor de la tesis

A mi familia

Agradecimientos

En primer lugar expresar mi agradecimiento a todas aquellas personas que, aunque no nombre explícitamente en los posteriores párrafos, me han ayudado a realizar esta memoria.

Especial agradecimiento merece mi director, Antonio García Loureiro, por su constante ayuda y apoyo a lo largo de los más de cuatro años de elaboración de esta tesis. Sin su esfuerzo y dedicación en los momentos de más dificultad no habría llegado hasta aquí.

A todos los miembros del Departamento de Electrónica y Computación, especialmente a los integrantes del Grupo de Arquitectura de Computadores, por proporcionarme el soporte necesario para la realización de este trabajo. Sin las facilidades que me han dado esta memoria tampoco habría sido posible. A mis compañeros (y ex-compañeros), en especial Óscar, Javi, Diego, Dani, Julio, Juan Ángel, Fabi, Marcos, Cris, Raúl y Enrique, por haber hecho agradables tantas horas en el laboratorio. Y a Natalia, por su ayuda desde el primer día, por su paciencia, por su alegría y por su amistad.

También tengo que agradecer a los miembros del Device Modelling Group de la Universidad de Glasgow su amabilidad y colaboración durante mis estancias en Escocia. Entre ellos, mencionar al Profesor Asen Asenov, por su contribución al desarrollo de esta tesis doctoral, y a Karol Kalna, por su constante ayuda, sus ideas y sus explicaciones, en especial en mis primeros pasos con el método Monte Carlo.

Al EPCC y sobre todo al CESGA por posibilitar y facilitar el acceso a sus recursos computacionales. A los Ministerios de Ciencia y Tecnología y posteriormente de Educación y Ciencia por la financiación a través de los proyectos TIN2004-07797-C02 y TIN 2007-67537-C03 y a través de una beca del programa de “Formación de Profesorado Universitario”. A la Xunta de Galicia por la financiación a través de los proyectos DXIDI07TIC01CT y INCITE08PXIB206094PR.

Por supuesto, a mi familia, por su constante apoyo y ayuda. Y a mis amigos, que consiguen que pueda escapar del trabajo.

Y de nuevo a todas las personas que no he nombrado y que han hecho posible que llegue hasta aquí.

Marzo de 2009

La ciencia será siempre una búsqueda, jamás un descubrimiento real. Es un viaje, nunca una llegada.

Karl Popper

–Me asustan los suelos.

–Querrás decir las alturas –dijo Conina–. Y deja de hacer el tonto.

–¡Ya sé lo que quiero decir! ¡Son los suelos los que te matan!

Rechicero. Terry Pratchett

Índice general

| | |
|--|-----------|
| Introducción | 1 |
| Introduction | 5 |
| 1. Dispositivos MOSFET | 9 |
| 1.1. El MOSFET convencional, escalado y estado actual de la tecnología | 10 |
| 1.2. Alternativas a los MOSFET estándar | 12 |
| 1.2.1. MOSFETs multipuerta | 13 |
| 1.2.2. Nuevos materiales | 15 |
| 1.3. Técnicas de simulación | 17 |
| 1.3.1. Arrastre-difusión | 18 |
| 1.3.2. Modelo hidrodinámico | 19 |
| 1.3.3. Monte Carlo | 20 |
| 1.3.4. Transporte cuántico | 20 |
| 2. Generación de mallas tetraédricas | 23 |
| 2.1. Clasificación de mallas | 24 |
| 2.2. Métodos de generación de mallas | 25 |
| 2.2.1. Triangulaciones de Delaunay | 26 |
| 2.2.2. Método <i>advancing front</i> | 31 |
| 2.2.3. Métodos basados en la descomposición recursiva del espacio | 33 |
| 2.3. Generación de mallas usando un <i>octree</i> | 36 |
| 2.3.1. Construcción del <i>octree</i> | 37 |
| 2.3.2. Descomposición en tetraedros de los octantes terminales | 41 |
| 2.3.3. Ejemplos: tiempos de generación de las mallas | 44 |
| 2.3.4. Ejemplos: calidad de las mallas | 48 |

| | | |
|-----------|---|-----------|
| 2.4. | Generación de mallas atomísticas | 52 |
| 2.4.1. | Teoría del funcional de la densidad | 57 |
| 2.4.2. | Regiones cristalinas | 60 |
| 2.4.3. | Regiones de interfaz | 61 |
| 2.4.4. | Regiones amorfas | 67 |
| 2.4.5. | Calidad de las mallas | 69 |
| 2.5. | Refinamiento adaptativo de mallas tetraédricas | 71 |
| 3. | El modelo de arrastre-difusión | 75 |
| 3.1. | El modelo de arrastre-difusión | 76 |
| 3.1.1. | Ecuación de Poisson | 76 |
| 3.1.2. | Ecuaciones de continuidad | 78 |
| 3.1.3. | Concentración de portadores | 79 |
| 3.1.4. | Factor de generación-recombinación | 85 |
| 3.2. | Correcciones cuánticas | 87 |
| 3.2.1. | El modelo <i>density gradient</i> | 88 |
| 3.2.2. | El modelo <i>effective conduction band edge</i> | 90 |
| 3.2.3. | Otras aproximaciones | 91 |
| 3.3. | Escalado de las variables | 92 |
| 3.4. | Discretización de las ecuaciones de arrastre-difusión y <i>density gradient</i> | 94 |
| 3.4.1. | El método de los elementos finitos | 95 |
| 3.4.2. | Ecuación de Poisson | 101 |
| 3.4.3. | Ecuaciones de continuidad | 103 |
| 3.4.4. | Ecuaciones del modelo <i>density gradient</i> | 107 |
| 3.5. | Resolución paralela de sistemas de ecuaciones | 110 |
| 3.5.1. | Ecuaciones del sistema discretizado | 111 |
| 3.5.2. | Método de Gummel | 112 |
| 3.5.3. | Método de Newton-Raphson | 114 |
| 3.5.4. | Resolución de sistemas lineales | 115 |
| 3.6. | Simulación de fluctuaciones en dispositivos MOSFET de silicio | 117 |
| 3.6.1. | Impacto de la granularidad del polisilicio | 117 |
| 3.6.2. | Impacto de algunas no idealidades de óxidos de alta permitividad | 119 |
| 3.6.3. | Impacto de las posiciones aleatorias de los dopantes | 123 |
| 3.6.4. | Impacto de las regiones de transición entre materiales | 127 |
| 3.6.5. | Simulación de fluctuaciones en dispositivos IF III-V MOSFET | 129 |

| | |
|--|----------------|
| 4. El método de simulación Monte Carlo | 137 |
| 4.1. Ecuación de transporte de Boltzmann | 138 |
| 4.2. El método Monte Carlo aplicado a la simulación del transporte en semiconductores | 141 |
| 4.2.1. Vuelo libre de los portadores | 143 |
| 4.2.2. Dispersión de los portadores | 144 |
| 4.3. Técnicas de paralelización de simuladores Monte Carlo | 144 |
| 4.4. Simulaciones Monte Carlo del transporte en silicio | 148 |
| 4.4.1. Estructura de bandas y aproximación analítica | 149 |
| 4.4.2. Principales mecanismos de dispersión | 151 |
| 4.4.3. Resultados | 156 |
| 4.5. Simulación de dispositivos: uso de mallas tetraédricas | 157 |
| 4.5.1. Correcciones cuánticas | 157 |
| 4.5.2. Autofuerzas: asignación de carga e interpolación del cam- po eléctrico | 158 |
| 4.5.3. Inyección de portadores sobre los contactos óhmicos | 165 |
| 4.5.4. Integración de las ecuaciones del movimiento | 166 |
| 4.6. Simulación de un DG MOSFET de 10 nm de puerta | 170 |
| Conclusiones | 175 |
| Conclusions | 181 |
| A. Development of tools for the numerical simulation of MOS- FETs | 187 |
| A.1. Modified octree mesh generation for Manhattan type structures | 187 |
| A.1.1. Standard octree generation | 188 |
| A.1.2. Modified octree generation | 188 |
| A.1.3. Tetrahedral mesh generation algorithms | 189 |
| A.1.4. Mesh quality | 192 |
| A.2. Atomistic mesh generation for the simulation of nanoscale MOS- FETs | 193 |
| A.2.1. Meshing approach for crystalline regions | 193 |
| A.2.2. Meshing of amorphous layers | 197 |
| A.2.3. Mesh quality | 199 |
| A.3. Study of the impact of grain boundaries in MOSFETs using drift-diffusion simulations | 200 |
| A.3.1. Polysilicon grain boundary | 200 |

| | |
|--|-----|
| A.3.2. Gate dielectric granularity | 202 |
| A.4. Study of random dopant induced fluctuations using drift-diffusion simulations | 204 |
| A.5. Simulation of variability in InGaAs Implant-Free NanoMOS- FETs using drift-diffusion simulations | 208 |
| A.6. Efficient parallel Monte Carlo simulations using finite element tetrahedral meshes applied to novel MOSFET architectures . . . | 212 |
| A.7. Reduction of the self-forces for Monte Carlo simulation of semi- conductor devices on general unstructured meshes | 217 |
| A.7.1. Charge assignment | 219 |
| A.7.2. Electric field interpolation | 220 |
| A.7.3. Double gate MOSFET simulation | 221 |

Bibliografia

Índice de figuras

| | | |
|-------|--|----|
| 1.1. | Esquema básico del funcionamiento de un MOSFET. | 10 |
| 1.2. | Esquema de tres tipos básicos de MOSFETs de doble puerta: (a) canal horizontal, (b) canal vertical (FinFET) y (c) canal vertical con puertas independientes (MIGFET). | 14 |
| 1.3. | Esquema de la sección transversal de tres tipos de MOSFETs de triple puerta: (a) TriGate, (b) puerta Π y (c) puerta Ω | 14 |
| 1.4. | Esquema de la sección transversal de diferentes transistores con una puerta rodeando por completo el cuerpo de silicio: (a) sección transversal cuadrada, (b) sección transversal circular y (c) MBCFET. | 15 |
| 1.5. | Escalera de jerarquía para diferentes métodos de simulación en función de su complejidad computacional y su tiempo de computación. | 18 |
| 2.1. | Ejemplo de malla estructurada. | 24 |
| 2.2. | Ejemplo de malla no estructurada. | 25 |
| 2.3. | Ejemplo de malla de cuadriláteros. | 25 |
| 2.4. | Ejemplo de malla con cuadriláteros y triángulos. | 25 |
| 2.5. | Diagrama de Voronoï y triangulación de Delaunay asociada. | 27 |
| 2.6. | Triángulos que (a) verifican la condición del círculo vacío y que (b) no la verifican. | 27 |
| 2.7. | Ejemplo del algoritmo de giro de aristas. | 28 |
| 2.8. | Ejemplo del método del polígono de inserción. | 28 |
| 2.9. | Tetraedro <i>sliver</i> | 29 |
| 2.10. | Intercambio de aristas en 2D. | 30 |
| 2.11. | Eliminación de tripletes. | 30 |
| 2.12. | Ejemplo de inicio del proceso de triangulación usando el método <i>advancing front</i> | 31 |
| 2.13. | Triángulos generados (a) sin considerar la proximidad de otro segmento del frente y (b) considerando este factor. | 32 |
| 2.14. | Descomposición de un cuadrado y estructura en árbol equivalente. | 34 |
| 2.15. | Tres formas posibles de triangular cuadrantes terminales. | 34 |
| 2.16. | Descomposición de un cubo y árbol equivalente. | 35 |
| 2.17. | Descomposición de un cubo en cinco tetraedros. | 36 |

| | |
|---|----|
| 2.18. Descomposición de un octante en cuatro hijos. | 36 |
| 2.19. Regiones usadas para definir (a) un HEMT y (b) un MOSFET con puerta de polisilicio. | 37 |
| 2.20. Diagrama de flujo correspondiente a la generación del <i>octree</i> | 39 |
| 2.21. Ejemplo 2D de (a) una región y (b) los octantes terminales después del ajuste a las fronteras usando $n_D = 3$ | 40 |
| 2.22. Ejemplo de asignación errónea en caso no transmitir la deformación a los padres. | 40 |
| 2.23. Conectividades entre nodos en un octante terminal para (a) el algoritmo A, (b) el algoritmo B, (c) el algoritmo C, (d) el algoritmo D y (e) el algoritmo E. | 42 |
| 2.24. Tetraedros generados usando el algoritmo A en el caso de tener un vecino más refinado. | 42 |
| 2.25. Dependencia del tiempo de generación del <i>octree</i> con su tamaño. | 45 |
| 2.26. Dependencia con el tamaño del <i>octree</i> de los tiempos de (a) adaptación a las fronteras y (b) balanceo. | 45 |
| 2.27. Dependencia del tiempo de ajuste con el número de planos que definen la región. | 46 |
| 2.28. Tiempos de generación de nodos y tetraedros para diferentes tamaños del <i>octree</i> usando (a) el algoritmo A, (b) el algoritmo B, (c) el algoritmo C, (d) el algoritmo D y (e) el algoritmo E. | 47 |
| 2.29. Tiempos relativos de generación de (a) nodos y (b) tetraedros. | 48 |
| 2.30. Distribución de calidades de las mallas del HEMT usando (a) el algoritmo A, (b) el algoritmo B, (c) el algoritmo C, (d) el algoritmo D and (e) el algoritmo E. | 50 |
| 2.31. Distribución de calidades de las mallas del MOSFET usando (a) el algoritmo A, (b) el algoritmo B, (c) el algoritmo C, (d) el algoritmo D y (e) el algoritmo E. | 51 |
| 2.32. Malla final de (a) un HEMT usando el algoritmo A y (b) un MOSFET usando el algoritmo E. | 52 |
| 2.33. Distribución de las conexiones en una malla ejemplo para los diferentes algoritmos. | 53 |
| 2.34. Ilustración de la estructura atómica ideal de un transistor DG-MOSFET de silicio de longitud de puerta de 10 nm. Las esferas rojas representan átomos de silicio, mientras que las verdes representan los átomos de oxígeno del óxido de puerta. | 55 |
| 2.35. Estructura de dióxido de silicio amorfo con 192 átomos ($\rho = 2,39g/cm^{-3}$) obtenida de simulaciones de dinámica molecular clásicas [MSSS04]. | 55 |
| 2.36. Estructura atómica de la interfaz cristalina Si/SiO ₂ pasivada con hidrógeno [YKU ⁺ 01]. | 56 |
| 2.37. Malla del octante terminal usando los átomos en la celda unidad del silicio como nodos. | 61 |

| | |
|--|-----|
| 2.38. Patrón utilizado para construir la malla de SiO ₂ cristalino usando nodos en las posiciones atómicas. | 62 |
| 2.39. Malla de una lámina de silicio de 6.5 nm de grosor. | 62 |
| 2.40. Patrón utilizado para construir la malla de la interfaz entre silicio y dióxido de silicio cristalinos. | 63 |
| 2.41. Malla de un MOSFET de doble puerta con SiO ₂ cristalino como dieléctrico de puerta. | 64 |
| 2.42. Malla de un MOSFET con un dieléctrico de puerta compuesto por una capa de SiO ₂ cristalino y otra de HfO ₂ (monoclínico). | 64 |
| 2.43. Estructura atómica calculada de una interfaz Si/SiO ₂ /HfO ₂ | 65 |
| 2.44. Posibles crecimientos de dos materiales cristalinos con diferente constante de red. Dependiendo del grosor de la capa, la estructura resultante puede estar tensionada (abajo) o dislocada (arriba). | 66 |
| 2.45. Dos triángulos delimitando una superficie no convexa. | 68 |
| 2.46. Regiones de la superficie que pueden y no pueden ser utilizadas para mover el nodo. | 68 |
| 2.47. Discretización de un transistor con regiones amorfas y cristalinas obtenida de la malla de materiales cristalinos. | 69 |
| 2.48. Átomos en la estructura después del proceso de hacer el óxido amorfo. Las esferas rojas representan átomos de silicio y las verdes de oxígeno. Notar las posiciones irregulares de los átomos en la capa de óxido al compararlos con sus posiciones regulares en la figura 2.34. | 70 |
| 2.49. Refinamiento de un tetraedro usando bisección. | 71 |
| 2.50. Detalle del refinamiento cerca de un átomo de impureza. Los colores se corresponden con valores del potencial. | 72 |
| 2.51. Evolución de la norma del residuo de la ecuación de continuidad con el número de iteración para la malla original y una refinada. | 73 |
| | |
| 3.1. Niveles de energía del material de referencia junto con los niveles de energía de una región del semiconductor. | 82 |
| 3.2. Elemento maestro tetraédrico de cuatro nodos. | 99 |
| 3.3. Transformación entre el tetraedro maestro y un tetraedro arbitrario. | 100 |
| 3.4. Bandas de conducción y valencia del polisilicio en una línea que corta perpendicularmente el plano de la frontera entre cristales. | 118 |
| 3.5. Desplazamiento del voltaje umbral en el dispositivo de 80 nm de longitud de puerta producido por una frontera moviéndose desde el extremo de fuente (valores negativos) al de drenador (valores positivos). | 119 |
| 3.6. Desplazamiento del voltaje umbral en el dispositivo de 25 nm de longitud de puerta producido por una frontera moviéndose desde el extremo de fuente (valores negativos) al de drenador (valores positivos). | 120 |
| 3.7. Potencial electrostático bajo la capa de dieléctrico cuando hay una frontera entre cristales de polisilicio en el plano $x = -2,5$ nm. | 121 |

| | |
|--|-----|
| 3.8. Desplazamiento del voltaje umbral en el MOSFET de 25 nm de longitud de puerta producido por una frontera en el dieléctrico moviéndose del extremo de fuente (valores negativos) al de drenador (valores positivos). Los valores del desplazamiento son relativos al de un dispositivo con Si_3N_4 como dieléctrico de puerta. | 122 |
| 3.9. Dependencia del desplazamiento de la tensión umbral con el ángulo de la frontera entre los dos cristales y el eje x en el dispositivo de 25 nm de longitud de puerta. | 122 |
| 3.10. Geometría del MOSFET de doble puerta simulado. Las dimensiones están dadas en unidades de la constante de red del silicio (1 uc = 0,543 nm). | 123 |
| 3.11. Potencial electrostático para una configuración aleatoria de dopantes para un voltaje de drenador de 0,05 V y de puerta de 1,1 V. | 124 |
| 3.12. Curvas características I_D-V_G para un voltaje de drenador de 0,05 para un grupo de transistores simulados. Las corrientes de drenador usando dopado continuo con las mallas atomística (círculos) y no atomística (cruces) se muestran por comparación. | 124 |
| 3.13. Desviación estándar normalizada de la corriente de drenador como función de la tensión de puerta para una polarización de drenador de 0,05 V. | 125 |
| 3.14. Variación relativa de la corriente de drenador como función de la tensión de puerta para una tensión de drenador de 0,05 V. | 126 |
| 3.15. Curvas características I_D-V_G de los dispositivos con las corrientes mayor (max) y menor (min) usando la malla atomística y la malla uniforme con dos métodos de asignación de carga, CIC y NGP. | 127 |
| 3.16. Isosuperficies del ancho de la banda prohibida en la región de transición del SiO_2 | 128 |
| 3.17. Curvas características I_D-V_G para los dispositivos con interfaz abrupta y realista. | 129 |
| 3.18. Sección transversal de un IF $\text{In}_{0,75}\text{Ga}_{0,25}\text{As}$ MOSFET tipo n con la capa de dopado δ debajo del canal. | 130 |
| 3.19. Calibración de las curvas I_D-V_G frente a simulaciones Monte Carlo con voltajes de drenador de 0,1 V y 0,8 V para el IF $\text{In}_{0,75}\text{Ga}_{0,25}\text{As}$ MOSFET con longitud de puerta de 15 nm. | 131 |
| 3.20. Ejemplo de la distribución de dopantes en la capa de dopado δ para un dopado nominal de $1.5 \times 10^{19} \text{ cm}^{-3}$. El número total de átomos en la región es aproximadamente 60000. | 132 |
| 3.21. Distribución de potencial en equilibrio en la capa de dopado δ y en el canal para una configuración de dopantes generada aleatoriamente. | 133 |
| 3.22. Concentración de electrones en equilibrio en la capa de dopado δ y en el canal para una configuración de dopantes generada aleatoriamente. | 133 |

| | |
|---|-----|
| 3.23. Diagrama de dispersión que muestra la tensión umbral y el número de dopantes para cada simulación del conjunto de transistores. En los lados izquierdo y superior se muestran los histogramas de cada variable junto con la función de densidad de probabilidad esperada y el valor de la variable en el caso de la simulación continua. | 134 |
| 3.24. Correlación entre la posición de los dopantes a lo largo del eje x y el valor de la tensión umbral. | 135 |
| 4.1. Flujo de un programa Monte Carlo típico. | 142 |
| 4.2. Diagrama de la subrutina Monte Carlo paralelizada. | 146 |
| 4.3. Aceleración (cruces) y eficiencia paralela (círculos) de un paso temporal. Los valores de escalado ideal (líneas discontinuas) se muestran por comparación. | 147 |
| 4.4. (a) Estructura de bandas del Si calculada usando la aproximación de electrones fuertemente ligados. (b) Detalle de la región próxima a la banda de energías prohibidas. En ambos casos las primeras bandas de valencia y conducción están resaltadas en rojo. El resto de las bandas de conducción están en azul y las de valencia en negro. | 149 |
| 4.5. Primera zona de Brillouin para una red fcc en la que se muestran puntos de alta simetría. | 150 |
| 4.6. Superficies de energía constante próximas al mínimo de la banda de conducción. | 151 |
| 4.7. Velocidad media de los electrones frente al campo eléctrico para el silicio en las direcciones $\langle 100 \rangle$ y $\langle 111 \rangle$ comparadas con datos experimentales en escalas logarítmica y lineal. | 155 |
| 4.8. Movilidad del silicio para diferentes concentraciones de impurezas comparadas con datos experimentales (cuadrados) y simulaciones con la dispersión por impurezas ionizadas introducida <i>ab initio</i> (estrellas) [ARA08]. Los datos con los modelos de apantallamiento se corresponden con: modelo de Debye-Hückel (límite no-degenerado, círculos), modelo de Thomas-Fermi (límite degenerado, rombos) y modelo estático descrito en el texto (triángulos). | 156 |
| 4.9. Concentración media de portadores en dos secciones transversales de un TriGate MOSFET de 40 nm de puerta ($V_D=0,05$ V y $V_G=1,1$ V) usando un potencial cuántico constante en toda la simulación. | 159 |
| 4.10. Interpretación geométrica de la asignación de carga para un elemento cartesiano (izquierda) y uno triangular (derecha). W_i es el peso asociado al nodo i y corresponde al área del subelemento opuesto al nodo. Los subelementos están definidos por un punto que sería la posición de la carga. | 162 |
| 4.11. Valor de la integral usada en el cálculo de las trayectorias con un paso temporal de 0.2 fs. Para las aproximaciones parabólica y de γ constante también se muestran los errores. | 169 |

| | |
|---|-----|
| 4.12. Isosuperficies (a) de la componente x y (b) de la componente y del campo eléctrico actuando sobre una partícula situada en el interior del dispositivo. | 171 |
| 4.13. Isosuperficies (a) de la componente z y (b) del módulo del campo eléctrico actuando sobre una partícula situada en el interior del dispositivo. . | 171 |
| 4.14. Reducción del campo eléctrico, en órdenes de magnitud, sobre la posición de una partícula en el transistor MOSFET de doble puerta de 10 nm de longitud de canal. | 172 |
| 4.15. Curvas I_D - V_G a $V_D = 0,05$ V del DG MOSFET de 10 nm de longitud de puerta obtenidas con diferentes métodos de simulación: arrastre-difusión (Medici [med]), Monte Carlo no autoconsistente (FF MC) y Monte Carlo autoconsistente (SC MC-1,2,3). | 173 |
| | |
| A.1. Conectividades among nodes in a terminal octant for (a) algorithm A, (b) algorithm B, (c) algorithm C, (d) algorithm D and (e) algorithm E | 191 |
| A.2. Mesh of the terminal octant using the atoms in the unit cell of (a) Si and (b) crystalline SiO_2 as nodes. | 194 |
| A.3. Pattern used to build the mesh of the interface between crystalline SiO_2 and crystalline Si. | 195 |
| A.4. Mesh of a DG-MOSFET with crystalline SiO_2 as gate dielectric using patterns. | 196 |
| A.5. Mesh of a MOSFET with gate dielectric composed of one layer of crystalline SiO_2 and monoclinic HfO_2 | 196 |
| A.6. (a) Two triangles delimiting a non-convex surface. (b) Regions of the surface which can and cannot be used to move the node. | 198 |
| A.7. Mesh with amorphous regions obtained from the mesh of the crystalline materials. | 199 |
| A.8. Conduction and valence bands of the polysilicon in a line crossing perpendicularly to the plane of the boundary. | 201 |
| A.9. Threshold voltage displacement in the 80 nm physical gate length device produced by a boundary moving from the source end (negative values) to the drain end (positive values) of the gate. | 201 |
| A.10. Threshold voltage shift in the 25 nm gate length device produced by a grain boundary moving from the source end (negative values) to the drain end (positive values) of the gate. | 202 |
| A.11. Threshold voltage displacement in the 25 nm gate length device produced by a boundary in the dielectric moving from the source end (negative values) to the drain end (positive values) of the gate. Displacement values are relative to the device with a Si_3N_4 gate dielectric. | 203 |
| A.12. Dependence of the threshold voltage displacement on the angle of the boundary between the two crystals and the x axis in the 25 nm gate length device. | 204 |

| | | |
|-------|--|-----|
| A.13. | Geometry of the simulated 10 nm gate length double gate MOSFET. Dimensions are in units of the Si unit cell (1 uc = 0,543 nm). | 205 |
| A.14. | I_D - V_G characteristics at a drain voltage of 0,05 V for an ensemble of simulated transistors. The drain current obtained using continuous doping with the non-atomistic (crosses) and atomistic (circles) meshes are highlighted. | 205 |
| A.15. | Normalised standard deviation of the current as a function of the gate voltage for a drain bias of 0,05 V. | 206 |
| A.16. | Relative variation of the drain current for a drain bias of 0,05 V. . . . | 207 |
| A.17. | I_D - V_G characteristics for the devices showing the lowest (min) and highest (max) currents using the atomistic mesh and the uniform mesh with two different charge assignments: cloud-in-cell (CIC) and the nearest grid point (NGP) scheme. | 208 |
| A.18. | Cross-section of n -type implant free $\text{In}_{0,75}\text{Ga}_{0,25}\text{As}$ MOSFET with a δ -doping layer below the channel. | 209 |
| A.19. | Calibration of I_D - V_G characteristics at drain biases of 0,1 V and 0,8 V for the $\text{In}_{0,75}\text{Ga}_{0,25}\text{As}$ implant free MOSFET with a gate length of 15 nm against Monte Carlo device simulations. | 210 |
| A.20. | Electron concentration, at equilibrium, in the δ -doping layer and at the bottom of the channel for a randomly generated pattern of dopants. . | 211 |
| A.21. | Scatter plot showing the threshold voltage and the dopant number variability in the ensemble of simulated transistors. Histograms for the individual distributions are shown on the left and top sides of the figure. The expected probability density functions and the continuous device values are also indicated for comparison. | 212 |
| A.22. | Speedup (crosses) and parallel efficiency (circles) of one time step. We also show the ideal values (dashed lines) for comparison. | 215 |
| A.23. | Geometrical meaning of charge assignment schemes for a Cartesian element (left) and a simplex element (right) in 2D. w_i is the weight associated to node i and corresponds to the area of the sub-element the label is on. Sub-elements are defined by the point x , which would be the position of the charge. | 219 |
| A.24. | Reduction of the electric field in orders of magnitude at the positions of a particle in the 10 nm gate length, 6,1 nm body thick, DG Si MOSFET. | 222 |
| A.25. | Isosurfaces of the (a) x component and (b) y component of the electric field on a particle positioned in different places inside the device. . . . | 223 |
| A.26. | Isosurfaces of the (a) x component and (b) absolute value of the electric field on a particle positioned in different places inside the device. . . . | 223 |
| A.27. | I_D - V_G characteristics at a drain voltage of $V_D = 0,05$ V for different charge assignment/electric field interpolation schemes | 224 |

Índice de tablas

| | |
|---|-----|
| 2.1. Nodos y elementos generados por los diferentes algoritmos de descomposición en tetraedros para un mismo <i>octree</i> | 44 |
| 2.2. Parámetros que describen la distribución de calidades de los tetraedros para las mallas estudiadas. | 49 |
| 2.3. Parámetros para describir la distribución de calidades de los tetraedros para las mallas de diferentes materiales. Los parámetros dados para las mallas amorfas son $\beta_1 = 0,05$ y $\beta_2 = 1,0$ | 70 |
| 3.1. Factores de escalado | 94 |
| 4.1. Tiempos medios, en segundos, de la simulación de un paso temporal ($t_{iteration}$), de una resolución de la ecuación de Poisson ($t_{Poisson}$), de los vuelos libres (t_{flight}), de la etapa de inyección en los contactos ($t_{cont.}$), de la comunicación de la concentración de electrones ($t_{reduc.}$), de la comunicación del potencial ($t_{com.pot.}$) y de la actualización del campo eléctrico (\vec{E}_{update}) para diferente número de procesadores (n_p). | 148 |
| A.1. Parameters describing quality distributions of the tetrahedra for the studied meshes | 193 |
| A.2. Parameters describing quality distributions of the tetrahedra for the meshes of different materials. The parameters given for the amorphous meshes are $\beta_1 = 0,05$ and $\beta_2 = 1,0$ | 200 |
| A.3. Mean simulation times (s) for one time step ($t_{iteration}$), linear Poisson solver ($t_{Poisson}$), particle flights (t_{flight}), contacts stage ($t_{contacts}$), reduction of electron concentration ($t_{reduction}$), communication of the potentials ($t_{comun.poten.}$) and electric field update (\vec{E}_{update}) for different number of processors (n_p) | 216 |

Introducción

En los últimos años el escalado de los transistores MOSFET, los más importantes para la industria de los dispositivos semiconductores, ha llevado la tecnología a un punto en el que se vislumbra el final del camino seguido hasta el momento. Algunos efectos intrínsecos a las propiedades de los materiales, como pueden ser las fluctuaciones debidas a la naturaleza discreta de la carga y la materia, hoy en día tienen que ser incorporados en el diseño de los sistemas para asegurar su correcto funcionamiento. Además de estos hay otros factores de vital importancia para mantener el escalado como por ejemplo la reducción de las corrientes de fuga a través del óxido de puerta.

En el pasado, las principales fuentes de fluctuaciones estaban asociadas a los procesos de fabricación, y algunas de ellas podían ser modeladas y compensadas. Sin embargo, con el escalado de los dispositivos a dimensiones del orden de los nanómetros, las fluctuaciones intrínsecas a la naturaleza de los materiales pasan a jugar el papel más importante. Estas fluctuaciones son independientes de los procesos litográficos y no pueden ser eliminadas mediante mejoras en los procesos de fabricación. Por lo tanto, sus efectos deben ser cuantificados y tenidos en cuenta en los diseños de los circuitos de los que formarán parte. Todo esto supone la necesidad de simular no un único dispositivo con propiedades continuas, sino un grupo de dispositivos microscópicamente diferentes que resultarán en variaciones en los valores de la corriente.

Además de esto, el escalado de los transistores MOSFET convencionales también lleva a los llamados efectos de canal corto, que básicamente provocan una pérdida en el control de la carga por parte de la puerta. Otros problemas con el escalado tradicional son el rendimiento y las corrientes de fuga, que no se pueden mantener en los niveles necesarios sin introducir cambios en los materiales de fabricación. Estos efectos llevan a pensar en nuevas alternativas al escalado tradicional que permitan prolongar el escalado durante más años. Algunas soluciones ya se han introducido en la industria, como pueden ser el

uso de silicio tenso para aumentar la movilidad o la introducción de materiales de alta permitividad como óxido de puerta para reducir las corrientes de fuga. Sin embargo, esto no es suficiente para sustentar el escalado hasta el final de la hoja de ruta [roa] y son necesarias nuevas innovaciones en el diseño de los dispositivos, desde nuevas arquitecturas hasta nuevos materiales para el canal.

Estas dos problemáticas que afectan a los dispositivos hacen que sea necesario el uso de simulaciones tridimensionales para poder predecir el funcionamiento de algunos de los dispositivos candidatos a formar parte de la industria de los semiconductores. Sin embargo, las simulaciones tridimensionales son computacionalmente muy costosas, y ha de tenerse mucho cuidado en la elección del modelo de simulación más apropiado para cada caso. Por un lado, si queremos hacer simulaciones estadísticas de las fluctuaciones en un dispositivo, el modelo más conveniente es el de arrastre-difusión, ya que su coste computacional es el más bajo permitiendo todavía una simulación predictiva de ciertos parámetros, especialmente aquellos asociados con la electrostática del problema. Por otro, en el caso de nuevas arquitecturas debemos evaluar el rendimiento antes de hacer cualquier estudio de fluctuaciones de parámetros intrínsecos, lo que necesita, en general, una herramienta de mayor poder predictivo para las propiedades de transporte del dispositivo. Los modelos que resuelven directamente las ecuaciones del transporte cuántico son todavía muy costosos y están restringidos a dispositivos de tamaño muy reducido. Por lo tanto, la herramienta más apropiada en estas condiciones es un simulador que resuelva la ecuación de transporte de Boltzmann, y el método Monte Carlo es el más extendido para este fin.

El gran coste que supone la simulación tridimensional, en especial en el caso de simulaciones estadísticas o de simulaciones Monte Carlo, hace que sea deseable, y a veces necesario, el uso de la computación paralela para aprovechar al máximo los recursos computacionales disponibles. Por este motivo, el simulador 3D de dispositivos utilizado a lo largo de este trabajo está paralelizado a través de la librería MPI de paso de mensajes, usando los lenguajes de programación C y Fortran, lo que garantiza la portabilidad del código entre numerosos sistemas de computación paralela.

Esta memoria se enmarca en una de las líneas de investigación del grupo de Arquitectura de Computadores del Departamento de Electrónica y Computación de la Universidad de Santiago de Compostela. Este trabajo de investigación parte de un simulador para transistores bipolares de homounión (BJT) y de heterounión (HBT) posteriormente extendido a la simulación de transistores

HEMT y MOSFET convencionales. A partir de este proyecto¹, desarrollamos una extensión de este simulador para:

- la inclusión de nuevos efectos de variabilidad entre transistores,
- la simulación de fluctuaciones con resolución espacial definida por la estructura atómica del material,
- el estudio de dispositivos MOSFET multipuerta,
- la simulación de MOSFETs mediante el método Monte Carlo con correcciones cuánticas.

Para esto, además, desarrollamos dos generadores de mallas que nos permiten la flexibilidad necesaria para acometer estas tareas. Uno de ellos, basado en el uso de árboles octales, permite discretizar regiones compuestas por paralelepípedos rectangulares y definir refinados arbitrarios. El otro generador de mallas sitúa un nodo en la posición de cada átomo del dispositivo, lo que permite una resolución espacial exacta de, por ejemplo, las posiciones de los dopantes.

Hemos dividido el trabajo en cuatro capítulos en los que se describen, tras una breve introducción a los dispositivos MOSFET y su simulación, las tres principales líneas de este trabajo.

En el primer capítulo resumimos, inicialmente, el estado de la tecnología MOSFET actual, tal como ha sido guiada por el escalado y las modificaciones que ha sido necesario introducir para llegar al estado actual de la tecnología. A continuación describimos dos de las alternativas para continuar con el escalado más allá de los que la tecnología tradicional permite: nuevas arquitecturas para los dispositivos y nuevos materiales para el canal de los transistores. Finalmente, terminamos el capítulo con un resumen de los principales métodos de simulación disponibles actualmente.

En el segundo capítulo empezamos con una breve introducción a la clasificación de las mallas de simulación según sus propiedades. A continuación describimos los principales métodos de generación de mallas utilizados actualmente. Tras esta sección introductoria describimos los dos generadores de

¹Información sobre este proyecto se encuentra en las tesis doctorales de Antonio Jesús García Loureiro, de título: **BIPS3D: Un simulador 3D paralelo de dispositivos bipolares BJT y HBT**, y de Natalia Seoane Iglesias, de título **Optimización de un simulador 3D paralelo aplicado al estudio de fluctuaciones de parámetros intrínsecos en dispositivos HEMT**, realizadas en el Departamento de Electrónica y Computación de la Universidad de Santiago de Compostela.

mallas desarrollados y que emplearemos posteriormente en las simulaciones. En primer lugar describimos un generador basado en el uso de árboles octales (*octrees*) que permite el mallado de estructuras de tipo Manhattan y que nos permite una gran libertad en la especificación del refinado. A continuación introducimos la generación de mallas atomísticas tras una breve introducción a la teoría del funcional de la densidad, ya que es el método que utilizaremos para obtener las distribuciones de átomos en las estructuras que posteriormente utilizamos para generar las mallas.

En el tercer capítulo presentamos los resultados obtenidos utilizando el método de arrastre-difusión. Empezamos describiendo los modelos físicos utilizados, incluyendo los métodos para la inclusión de correcciones cuánticas. A continuación describimos la discretización de todas las ecuaciones utilizadas mediante el método de los elementos finitos. Finalmente, tras la descripción de los métodos de resolución de las ecuaciones, presentamos los diferentes estudios de fluctuaciones realizados: impacto de la granularidad del polisilicio y de diferentes constantes dieléctricas en el óxido de puerta de un MOSFET de arquitectura convencional, impacto de la posición aleatoria de los dopantes y la región de transición entre óxido y silicio en un transistor MOSFET de doble puerta e impacto de la posición aleatoria de los dopantes en la capa de dopado δ de un MOSFET con arquitectura *Implant Free* con canal de InGaAs.

En el cuarto capítulo empezamos describiendo la ecuación de transporte de Boltzmann y la aplicación del método Monte Carlo a su resolución. Después describimos la paralelización del sistema completo acoplado con la ecuación de Poisson. A continuación describimos los mecanismos de dispersión incluidos y resultados de simulaciones de silicio cristalino para validar los modelos. Una vez terminada esta descripción, pasamos a describir los principales problemas que surgen en la implementación del método Monte Carlo sobre dominios discretizados mediante mallas tetraédricas. Finalmente, terminamos el capítulo con resultados de simulación de un transistor de doble puerta de canal de silicio.

Tras los cuatro capítulos, en los que se describe el trabajo realizado, se indican las principales conclusiones y aportaciones de este trabajo, así como las líneas de investigación abiertas por el mismo.

Introduction

In the last years, the scaling of MOSFETs (Metal Oxide Semiconductor Field Effect Transistors), key devices for the semiconductor industry, has led the technology to a point where the present concepts are reaching an end. Some of the intrinsic properties of the materials, such as fluctuations induced by the discrete nature of charge and matter, nowadays have to be included in the design of the systems to ensure its proper operation. On top of this, there are other factors of vital importance to sustain scaling, such as the reduction of leakage currents through the gate oxide.

In the past, the main sources of variability were associated to fabrication processes, and some of them could be modelled and compensated. However, with the scaling of devices to the nanometre regime, intrinsic fluctuations originated in the discreteness of materials begin to play the dominant role. These fluctuations are independent of lithographic processes and cannot be eliminated with improvements in the fabrication. Therefore, their effects must be quantified and accounted for in the design of the circuits which will integrate them. This means that simulations should deal with a whole ensemble of microscopically different devices, and not only with one with continuous properties. This will lead to a variability in the simulated values of the current-voltage characteristics which can be used to predict the variability in the fabricated devices.

In addition, the scaling of conventional MOSFETs leads to the so called short channel effects, which basically cause that the gate terminal loses the control on the charge in the device. Other problems are, for example, performance and leakage currents, which cannot be kept in the required levels without introducing changes in the fabrication materials. These problems necessitate the consideration of new alternatives to traditional scaling to sustain it for a longer period. Some solutions have already been introduced in the industry. This is the case of strained silicon to improve mobilities or high- κ

dielectrics for the gate stack to reduce leakage currents through the gate. However, this is not enough to maintain scaling to the end of the roadmap [roa] and new revolutions in the design of devices are required, from new architectures to new channel materials.

These two different problems affecting semiconductor devices make necessary the use of three-dimensional (3D) simulations to predict the performance of some of the new candidates to become major contenders in the semiconductor industry. 3D simulations are computationally very expensive and the simulation model must be chosen with great care. On the one hand, if we need to carry out statistical simulations of fluctuation effects in a device, the most convenient model is drift-diffusion, since it has the lowest computational cost and the ability to predict some parameters, specially those associated with the electrostatics of the device. On the other hand, to evaluate the performance of new architectures we need, in general, a predictive tool which capture the transport properties of the device. Quantum transport simulations are still too expensive and are restricted to very small devices. In these conditions, the most appropriate tool would be a simulator which solves the Boltzmann Transport Equation (BTE), and Monte Carlo methods are the most extended to handle this problem.

The computational cost of 3D simulations, specially those involving statistical samples or the Monte Carlo method, makes desirable, and sometimes mandatory, the use of parallel computing to take advantage of all the available computational resources. The 3D device simulator used throughout this work has been parallelised using the MPI library on C and Fortran programming languages, which guarantees the portability of the code.

This dissertation is part of one of the research lines of the Grupo de Arquitectura de Computadores from the Departamento de Electrónica e Computación of the Universidade de Santiago de Compostela. This research work is based on a 3D finite element simulator for bipolar junction transistors (BJTs) and heterojunction bipolar transistors (HBTs) later extended for high electron mobility transistors (HEMTs) and single gate (SG) MOSFETs. From this project, we have developed an extension of the simulator for:

- the inclusion of new sources of variability between transistors,
- the simulation of fluctuations with spatial resolution given by the atomic structure of the materials,
- the study of multigate MOSFET architectures,

- the simulation of MOSFETs using a Monte Carlo method with quantum corrections.

Furthermore, we have developed two mesh generators with the necessary flexibility to achieve these objectives. One of them, based on the use of octrees, allows us to discretise Manhattan type structures and define arbitrary refinements. The other mesh generator places one node in every atom position of the device, which gives us great spatial accuracy, for example, in the positions of the dopants.

We have divided this work in four chapters. In the first one, we begin summarising the present status of MOSFET technology, how it has been driven by scaling and the innovations which were necessary to reach the present technology. Then we describe two of the alternatives to continue scaling beyond the possibilities of the present technology: new device architectures and new channel materials. Finally, we give a brief description of the main simulation methods available nowadays.

In the second chapter we begin with a description of some possible classification of meshes according to their properties. We continue describing the main mesh generation methods employed: Delaunay meshes, advancing front techniques and octree based methods. After these introductory sections, we describe the two mesh generators we have developed and which we will use in the simulations. First we describe an algorithm based on the use of octrees which allows the meshing of Manhattan-type structures with a great flexibility in the refinement specification. Finally, we introduce the generation of atomistic meshes after a brief introduction to the density functional theory (DFT), since it will be the method used to obtain the atom distribution in the structures that we will use to generate the meshes.

In the third chapter, we present the results obtained using the drift-diffusion approximation. Firstly, we describe the physical models employed, including different approximations to include quantum corrections. Then, we show the discretisation of all the equations using the finite element method. Finally, after the description of the resolution methods for the systems of equations, we present different studies of parameter fluctuations. These include the study of granularity in the polysilicon gate and gate oxide in a conventional SG MOSFET, the impact of random dopants and the Si/SiO₂ transition region in a double gate (DG) MOSFET and the impact of random dopants in the δ -doping layer of an Implant Free MOSFET with an InGaAs channel.

In the fourth chapter we begin describing the BTE and the application

of the Monte Carlo method to obtain its solution. After this, we describe the parallelisation strategy of the complete coupled BTE-Poisson system. We then describe the different scattering mechanisms included in the code and the simulation results for bulk silicon to validate the models. Once the models are described, we comment the problems that arise in the implementation of the Monte Carlo method on domains discretised using a tetrahedral mesh. Finally, we finish the chapter with the results obtained from a simulation of a silicon channel DG MOSFET.

Finally, we summarise the main conclusions of this dissertation, with special emphasis in the main contributions of this work, and we describe some possible future research topics to continue this work.

Capítulo 1

Dispositivos MOSFET

Durante más de 30 años, el transistor de efecto campo metal-óxido-semiconductor (MOSFET) ha sido la base de los circuitos integrados. Ya en el año 1965, Gordon E. Moore, observando que hasta el momento existía un crecimiento exponencial en el número de transistores en un chip, postuló que esta tendencia continuaría en el futuro [Moo65]. Esta afirmación, conocida hoy en día como la ley de Moore, expresa que cada tres años se duplica el rendimiento de los transistores y se cuadruplica el número de dispositivos presentes en un chip [Won02]. El progreso que supone esta ley ha sido permitido gracias al escalado agresivo de los dispositivos, que permite mejorar la velocidad de operación y la densidad de integración. Sin embargo, con la reducción de las dimensiones a escalas nanométricas que ocurre hoy en día, las reglas de escalado convencionales dejan de ser aplicables y es necesario cambiarlas. Por ejemplo, las corrientes de fuga imponen un límite al escalado de la tensión de alimentación y es necesario el uso de un principio de escalado generalizado [BWD84] que disminuye el cambio en la tensión de alimentación pero aumenta el dopado del canal más rápido para balancear los campos lateral y vertical.

En la primera parte de este capítulo describiremos los transistores MOSFET. Empezamos con una breve descripción del dispositivo en su estructura convencional, estudiando a continuación las limitaciones que aparecen con el escalado y como influyen en su diseño. A continuación describimos algunos de los diseños alternativos que se han propuesto para superar las limitaciones de la arquitectura convencional, como son los transistores multipuerta y el reemplazo del material del canal. Finalmente, para terminar el capítulo presentaremos una descripción de las principales técnicas de simulación utilizadas para tratar

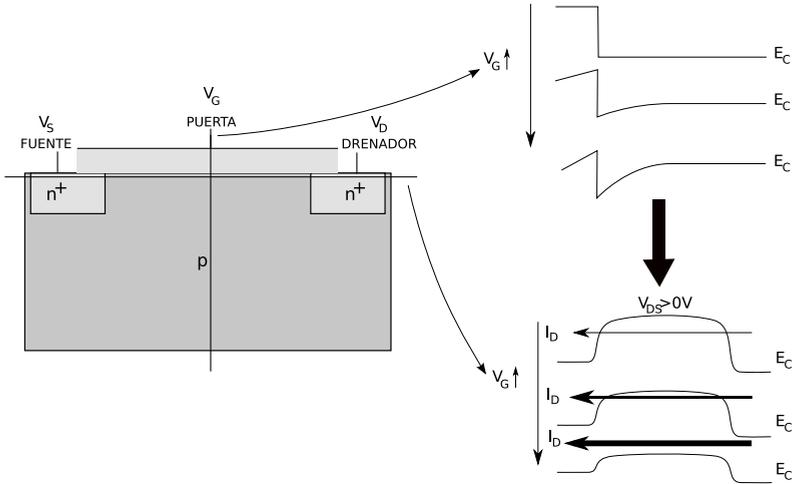


Figura 1.1: Esquema básico del funcionamiento de un MOSFET.

de predecir los diseños y arquitecturas óptimas, así como sus limitaciones.

1.1. El MOSFET convencional, escalado y estado actual de la tecnología

Los transistores de efecto campo son dispositivos unipolares que involucran principalmente el transporte de los portadores mayoritarios, ya sean electrones o huecos, en una capa paralela a la superficie. El más utilizado hoy en día es el MOSFET, que se caracteriza por aislar la capa de transporte del metal (o polisilicio) de puerta mediante la utilización de un óxido. El funcionamiento del dispositivo está basado en la modulación de la conductividad del canal mediante la aplicación de una tensión en la puerta. Este principio se ilustra, para un MOSFET de canal n , en la figura 1.1 mediante una representación esquemática del transistor. En este caso hay dos regiones con dopado tipo n alto, la fuente y el drenador, y el resto del dispositivo está dopado con impurezas aceptoras. Al aumentar el voltaje aplicado en la puerta disminuye la barrera que ven los electrones para pasar de fuente a drenador, lo que hace que aumente la conductividad.

En las tecnologías convencionales el semiconductor utilizado es el silicio. Este material tiene la ventaja de ser oxidado fácilmente para formar SiO_2 de una manera altamente controlable y reproducible. Además, la superficie de

separación Si-SiO₂ se puede crear con una muy buena regularidad, produciendo muy pocos defectos. Esto permite que los MOSFETs de Si se puedan fabricar en grandes cantidades, siendo fácilmente integrables para formar circuitos a gran escala.

La teoría original de escalado fue presentada en 1972 [DGKY72, DGR⁺74], ofreciendo parámetros de diseño concretos para la fabricación de MOSFETs de dimensiones reducidas. De acuerdo con esta teoría, si las dimensiones físicas y el potencial aplicado se escalan con un factor $1/k$ ($k > 1$) y la concentración de impurezas se aumenta un factor k , entonces la forma del campo eléctrico permanece constante. Sin embargo, al diseñar dispositivos con longitud de canal por debajo de $1 \mu\text{m}$, el escalado del voltaje umbral y de la alimentación pasa a ser demasiado agresivo y se presenta una nueva teoría de escalado generalizado [BWD84] que permite escalar los potenciales de forma independiente, dando una mayor flexibilidad al diseño. Una vez más, al disminuir las dimensiones físicas de los dispositivos, la teoría del escalado empieza a acercarse a su límite por motivos prácticos. Muchos de los problemas que ahora surgen son intrínsecos a la naturaleza de los semiconductores y no pueden ser eliminados por medio de mejoras en el procesamiento o en el equipamiento [Ase98]. Por ejemplo, el escalado de los dispositivos MOSFET a nodos tecnológicos inferiores al de 50 nm requiere la superación de barreras de naturaleza física que limitan su rendimiento [Key01, TBCF97, FDE⁺01, Fis03]. Algunos de los problemas más frecuentes son:

- Corrientes de fuga a través del óxido de puerta.
- Efecto túnel de portadores desde la fuente al drenador o desde el drenador hacia el interior del dispositivo.
- Control de la densidad y posición de los átomos dopantes en el canal y en las regiones de fuente y drenador para poder proporcionar una elevada relación entre las corrientes de conducción y de corte.

En los nodos tecnológicos actuales se han introducido numerosas variaciones sobre el MOSFET convencional. Entre estos cambios está el uso de puertas metálicas, óxidos de alta permitividad en la puerta o el uso de silicio tenso para aumentar la movilidad [MAA⁺07]. Además, compañías como IBM o AMD también introducen en nodos actuales diseños que van un paso más allá del MOSFET convencional utilizando la tecnología de silicio sobre aislante (SOI) [IN07, SCB⁺08] que, entre otras ventajas, permite reducir los efectos de canal corto [CC03].

Pese a las innovaciones introducidas para mejorar el rendimiento y continuar el escalado, hay dificultades tecnológicas que, en el estado actual de la tecnología, no tienen solución. El documento conocido como *SIA's International Technology Roadmap for Semiconductors* (ITRS) [roa] es una guía para la industria de los semiconductores que revisa los parámetros de diseño necesarios para poder ajustarse tanto a la ley de Moore como a las barreras tecnológicas a superar. En la última versión (*2008 Update*) la desaparición de la tecnología convencional sería en el nodo de 22 nm, aunque no hay soluciones de fabricación actualmente para los requerimientos pedidos desde el nodo de 32 nm. En los nodos posteriores es necesaria la introducción de tecnologías y materiales nuevos para continuar el escalado.

1.2. Alternativas a los MOSFET estándar

Una de las principales limitaciones de la tecnología convencional se encuentra en que el Si es un material de baja movilidad. La mayoría de los semiconductores compuestos (GaAs, InP, etc.) tienen movilidades mayores que la del silicio pero en la actualidad, a nivel comercial, no son utilizados en la fabricación de MOSFETs principalmente a causa de la dificultad de obtener aislantes válidos [YWK⁺03], pese al progreso realizado en los últimos años [Dat07, SKK⁺07, Pas05].

Los límites asociados con el escalado y el rango de aplicación de los MOSFETs convencionales han dado un impulso a la investigación y desarrollo de propuestas alternativas en el diseño de transistores y en el uso de nuevos materiales. Estas propuestas pueden ser clasificadas en función de la técnica que usan para intentar mejorar el rendimiento del dispositivo. Así es posible:

- Inducir una densidad de carga más elevada para una tensión de puerta dada. Por ejemplo, esto puede lograrse reduciendo la temperatura de funcionamiento del sistema [Won02] o usando un dispositivo FET de doble puerta (*Double Gate FET*) [SGZ⁺03, IWE⁺02].
- Aumentar el transporte de portadores elevando la movilidad, la velocidad de saturación o el transporte balístico. Para ello, entre otras técnicas, es posible disminuir la temperatura de funcionamiento, reducir la influencia de factores que degradan la movilidad, minimizando el campo eléctrico transversal o la dispersión coulombiana debida a átomos dopantes, o utilizar materiales de alta movilidad y velocidad de satu-

ración, como pueden ser Ge, InGaAs o InP. Entre los ejemplos de dispositivos que utilizan estas técnicas se encuentran los HEMT [YES02], PHEMT [KA01, KA04], MHEMT [WMH00], SiGe MOSFET [BOC02], HBT [WML⁺03] y DHBT [IKW02].

- Asegurar la escalabilidad del dispositivo a una longitud de puerta más pequeña. Esto se puede conseguir utilizando perfiles de dopado más bruscos, a través de una capacidad de puerta elevada o manteniendo un buen control electrostático del potencial en el canal. Esto ocurre en dispositivos FET de doble puerta, *Ground Plane FET* y *Ultra Thin Body SOI MOSFET* [PC02] entre otros.
- Reducir capacidades y resistencias parásitas. Estas técnicas son empleadas en dispositivos SOI [WFS98] y FET de doble puerta.

A continuación describiremos con más detalle dos caminos posibles para conseguir alcanzar los requerimientos descritos en el ITRS: los MOSFET multipuerta y la introducción de nuevos materiales en el canal [Vog07].

1.2.1. MOSFETs multipuerta

La disminución de la longitud de canal de los transistores que implica el escalado lleva a la aparición de los efectos de canal corto, que básicamente consisten en la pérdida del control de la carga por parte del terminal de puerta. Una posibilidad para mejorar esto es el uso de varias puertas para controlar la carga. Hay principalmente dos variedades: transistores multipuerta verticales y planares. Los primeros contienen, por ejemplo, a los FinFET [YCA⁺02, HLK⁺00] o los TriGate [KDD⁺06], mientras que los segundos son principalmente derivados de la tecnología SOI [CC03].

El primer transistor de doble puerta, descrito en 1984, fue llamado XMOS por su parecido con la letra Ξ [SH84] y en la publicación ya se predecían sus buenas propiedades frente a efectos de canal corto. El primer transistor de doble puerta fabricado fue el DELTA (*fully DEpleted Lean-channel TrAnsistor*) [HKKT89], en 1989. La estructura de este transistor era vertical y se puede considerar como el predecesor del FinFET [HLK⁺00], una de las arquitecturas más extendidas por su sencillez de fabricación usando tecnología para MOSFETs convencionales. En la figura 1.2 mostramos un esquema de tres tipos de transistor de doble puerta: el de canal horizontal, el de canal vertical (FinFET) y el de canal vertical con puertas independientes (MIGFET).

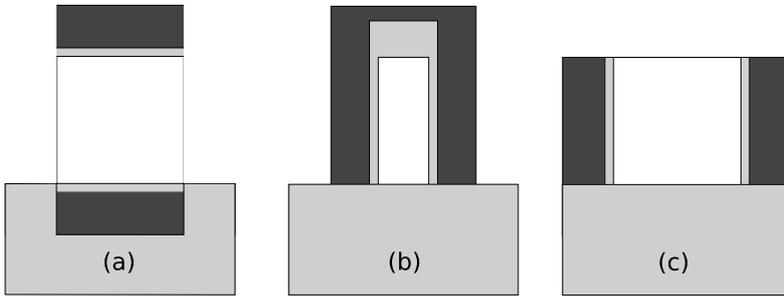


Figura 1.2: Esquema de tres tipos básicos de MOSFETs de doble puerta: (a) canal horizontal, (b) canal vertical (FinFET) y (c) canal vertical con puertas independientes (MIGFET).

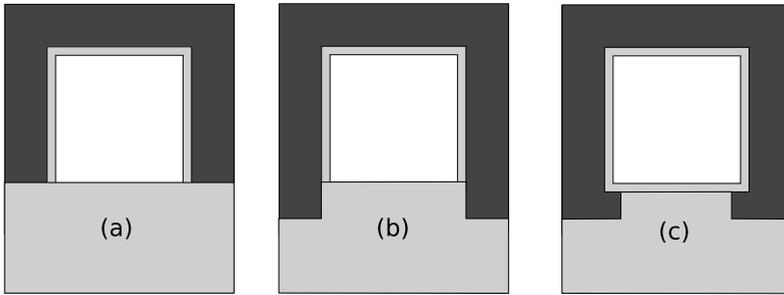


Figura 1.3: Esquema de la sección transversal de tres tipos de MOSFETs de triple puerta: (a) TriGate, (b) puerta II y (c) puerta Ω .

El paso lógico siguiente es un MOSFET con tres puertas para aumentar el control sobre la carga en el dispositivo. En este caso tenemos un hilo de silicio cubierto con óxido y un contacto de puerta en tres de sus lados. Entre las implementaciones están el hilo cuántico SOI MOSFET [CBBG96] y el TriGate MOSFET [KDD⁺06]. Además, existen otras versiones con un número efectivo de puertas mayor, como son los MOSFET de puerta II [PCD01] o de puerta Ω [YCC⁺02]. En el primer caso la puerta penetra verticalmente en el óxido, mientras que en el segundo la extensión de la puerta se dobla sobre el cuerpo de silicio, dejando la cara inferior parcialmente cubierta por el contacto de puerta. En la figura 1.3 mostramos los esquemas de las secciones transversales estos tres tipos de transistores de triple puerta: TriGate, puerta II y puerta Ω .

Finalmente, los dispositivos que ofrecen el mejor control sobre la región de canal son los MOSFETs con una puerta que los rodea por completo. Estos transistores pueden tener diferentes secciones transversales [NTO⁺91,

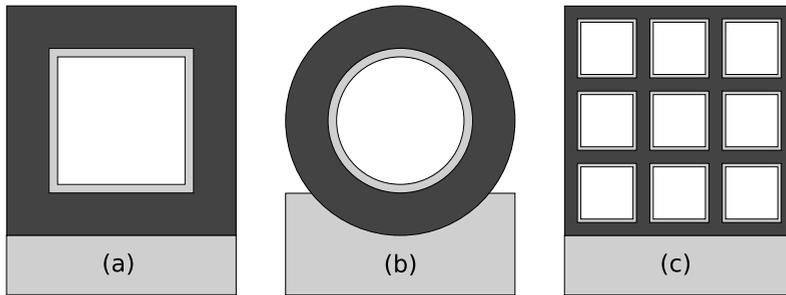


Figura 1.4: Esquema de la sección transversal de diferentes transistores con una puerta rodeando por completo el cuerpo de silicio: (a) sección transversal cuadrada, (b) sección transversal circular y (c) MBCFET.

MHM92] y ser integrados tanto en horizontal [CZW⁺03] como en vertical [GHFY06]. Además, para aumentar la corriente por unidad de área, pueden fabricarse varios transistores de tal forma que compartan fuente, drenador y puerta. Ejemplos de estos grupos son el *Multi-Bridge Channel* MOSFET (MBCFET) [YLK⁺04] y el *Twin-Silicon-Nanowire* MOSFET (TSNW-FET) [CSY⁺07]. En la figura 1.4 mostramos el esquema de secciones transversales de este tipo de transistores.

1.2.2. Nuevos materiales

Aunque en las últimas décadas la tecnología CMOS basada en silicio ha sido la dominante, en el escalado por debajo de los 100 nm las mejoras en rendimiento se han hecho cada vez más difíciles mediante el escalado tradicional. Además de las nuevas arquitecturas, la industria también ha empezado a utilizar nuevos materiales para lograr mejoras en el rendimiento. Un ejemplo es la introducción de materiales de alta permitividad para sustituir al dióxido de silicio como óxido de puerta y disminuir así las corrientes de fuga. Sin embargo, la introducción de nuevos materiales en el canal de los transistores supone un cambio mucho más drástico. Actualmente las mejoras en la movilidad se obtienen mediante la aplicación de tensión al silicio del canal, bien inducida en el proceso de fabricación [TAA⁺04], bien mediante ingeniería del sustrato [RKH⁺01]. Desafortunadamente, este beneficio es limitado y disminuye con el escalado [SKK⁺07], por lo que existe un gran interés en la introducción de otros materiales para alcanzar movilidades todavía mayores.

La introducción de nuevos materiales de alta movilidad como el germanio

para MOSFETs tipo p [SCB⁺04] y semiconductores III-V para MOSFETs tipo n [Dat07] y su integración en la plataforma del silicio es, actualmente, una solución ampliamente reconocida.

Hay tres problemas principales que dificultan la introducción de los transistores III-V en la tecnología CMOS [SKK⁺07]:

1. El problema más importante es que, a diferencia del silicio, no existe una buena solución para el óxido de puerta. Pese a los progresos de los últimos años [Dat07, SKK⁺07, Pas05], sigue siendo un problema para la implementación de aplicaciones CMOS.
2. Los sustratos de materiales III-V son caros, frágiles y difíciles de hacer en un tamaño grande. Además, el éxito de una tecnología no basada en silicio depende de su compatibilidad con los procesos utilizados para este material. Por este motivo hacen falta métodos para integrar los nuevos materiales sobre sustratos de silicio.
3. La movilidad de los huecos en los materiales III-V no es mejor que la del silicio. Hace falta un esquema de integración que permita combinar, por ejemplo, transistores III-V para n -MOSFETs y de germanio para p -MOSFETs. Además, la baja densidad de estados en los materiales III-V también podría ser un problema [PKK⁺06].

Hay principalmente dos estructuras que se estudian para MOSFETs III-V para aplicaciones digitales: de canal superficial y de canal enterrado. La primera estructura sería equivalente a los MOSFETs de silicio convencionales, mientras que en los MOSFETs de canal enterrado el canal suele estar entre dos capas de materiales III-V con un ancho de la banda prohibida mayor de manera que se forma un pozo cuántico. La estructura de canal superficial requiere la formación de una interfaz semiconductor-óxido de muy alta calidad para mantener una densidad de defectos baja cerca de la capa superficial de conducción.

El potencial para el escalado a dimensiones por debajo de los 20 nm de estas estructuras fue estudiado recientemente mediante simulaciones Monte Carlo [KAAM⁺08]. En este estudio se ve que los MOSFETs de canal superficial ofrecen una buena escalabilidad hasta longitudes de puerta del orden de los 20 nm, pero el óxido de puerta es una limitación fundamental para un mejor rendimiento de los transistores escalados. Por otro lado, en la simulación de una arquitectura con canal enterrado (*Implant Free MOSFET*), el rendimiento

escala adecuadamente hasta longitudes de puerta de 15 nm si se emplea un material de canal apropiado, en caso contrario la baja densidad de estados del material de canal puede ser el factor limitante en el escalado del rendimiento.

1.3. Técnicas de simulación

En la actualidad existe una jerarquía bien establecida de técnicas que pueden ser utilizadas en la simulación de dispositivos semiconductores modernos [Rav98]. En la figura 1.5 presentamos una escalera de jerarquía que los clasifica en base a dos parámetros, la complejidad computacional y el tiempo de simulación.

En la parte inferior de la escalera de jerarquía están los modelos compactos que contienen muy poca información sobre la física presente en el sistema y en general imitan el comportamiento del dispositivo utilizando aproximaciones analíticas y parámetros empíricos. Estos modelos requieren poco tiempo de cálculo pero su validez es limitada puesto que ignoran la naturaleza distribuida de los parámetros y la compleja geometría del dispositivo.

El siguiente nivel de técnicas de simulación es la aproximación de arrastre-difusión (DD, *drift-diffusion*) a la ecuación de transporte de Boltzmann (BTE) [Sel84]. La aproximación DD considera sólo los primeros dos momentos de la BTE, la ecuación de continuidad de corriente y la ecuación de conservación del momento. Estas ecuaciones están acopladas a la ecuación de Poisson por el potencial electrostático. La aproximación DD incluye una relación local entre la velocidad y el campo eléctrico y no puede representar apropiadamente efectos de transporte fuera del equilibrio.

El siguiente nivel por encima del modelo DD es la aproximación hidrodinámica a la ecuación de transporte de Boltzmann. En este caso se incluye el tercer momento de la BTE, la ecuación de conservación de la energía, lo que a su vez hace más compleja la ecuación de conservación del momento. Esto permite el tratamiento de efectos fuera del equilibrio ya que incluye una relación no local entre el campo eléctrico y la velocidad.

El siguiente nivel consiste en utilizar una técnica Monte Carlo para la resolución de la BTE. En esta técnica un conjunto de partículas evoluciona a través de un espacio de aceleración real y de eventos de dispersión escogidos aleatoriamente. Estos métodos requieren un elevado tiempo de computación, por lo que son utilizados principalmente para dispositivos de dimensiones muy pequeñas en los que los modelos de arrastre-difusión no son válidos.

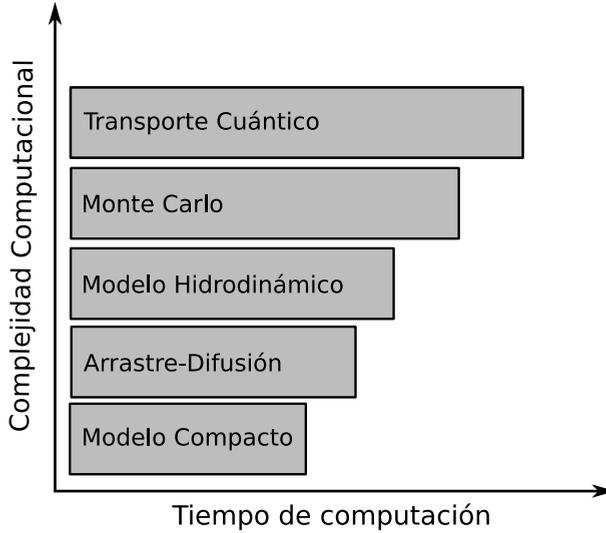


Figura 1.5: Escalera de jerarquía para diferentes métodos de simulación en función de su complejidad computacional y su tiempo de computación.

El nivel más elevado en la jerarquía está ocupado por las aproximaciones de transporte cuántico que utilizan las ecuaciones acopladas de Poisson y de Schrödinger independiente del tiempo, la matriz de densidad o la función de distribución Wigner. Todas estas técnicas son extremadamente costosas computacionalmente. Otra técnica que está ganando popularidad es el uso del formalismo de las funciones de Green fuera del equilibrio (NEGF *Non-Equilibrium Green Functions*), que permite la inclusión de la dispersión en la formulación del transporte cuántico. A continuación describimos brevemente las diferentes técnicas de simulación numéricas, destacando sus puntos fuertes y sus limitaciones para el estudio de fluctuaciones de parámetros intrínsecos, ya que este es uno de los principales objetivos de esta tesis.

1.3.1. Arrastre-difusión

El modelo de arrastre-difusión es el más simple empleado en simulaciones numéricas multidimensionales [GLKA05, ZWK⁺00]. Utiliza los dos momentos más bajos de la ecuación de transporte de Boltzmann. Las densidades de corriente de electrones J_n y huecos J_p se obtienen por medio de la suma de

dos componentes, una de arrastre gobernada por el campo eléctrico y otra de difusión dirigida por el gradiente de la densidad de portadores. Esta aproximación no tiene en cuenta la temperatura de los portadores y de forma estricta es solamente válida para campos en los que la velocidad de los portadores esté directamente relacionada con el campo eléctrico. Sin embargo, la validez de la aproximación de arrastre-difusión puede extenderse empíricamente introduciendo modelos de movilidad dependientes del campo eléctrico, lo que permite el uso de campos eléctricos más elevados en el proceso de simulación. Aunque incluso con esta extensión el sistema arrastre-difusión sólo funciona en el cuasi-equilibrio, donde el campo eléctrico varía lentamente y la velocidad está relacionada localmente con el campo. La principal ventaja de la aproximación de arrastre-difusión es su menor coste computacional en comparación con las otras técnicas de simulación, lo que la hace adecuada para realizar simulaciones estadísticas tridimensionales a gran escala, necesarias para la caracterización del impacto de diversas fuentes de fluctuaciones de parámetros intrínsecos. Además, para las ecuaciones de arrastre-difusión hay una gran variedad de técnicas numéricas plenamente desarrolladas.

1.3.2. Modelo hidrodinámico

Con el escalado de los dispositivos a dimensiones muy reducidas se produce una disparidad entre el rápido descenso de las dimensiones físicas y la mucho más lenta reducción de la tensión aplicada. Esto provoca un aumento en los campos en el interior de los dispositivos. Para tener en cuenta estos campos elevados el modelo de arrastre-difusión introduce modelos de movilidad dependientes del campo que relacionan de un modo local la velocidad de los portadores con la componente del campo en la dirección del flujo de corriente. Esta aproximación ignora fenómenos de transporte no locales en los que la velocidad de portadores en un punto está determinada por la distribución del campo a lo largo del camino de corriente. Una aproximación que supera estas limitaciones es el modelo hidrodinámico [ALD96, Gar94] al considerar el tercer momento de la ecuación de transporte de Boltzmann. En este caso la corriente tiene un término adicional proporcional al gradiente de la temperatura de los electrones y además se añade al sistema una ecuación de balance energético. Un problema que supone añadir momentos más elevados de la BTE es la estabilidad numérica de la solución, lo que lleva a un gran aumento de la complejidad computacional.

1.3.3. Monte Carlo

Un método alternativo de simular el transporte de portadores que no implica la discretización de la BTE o de sus momentos es la aproximación Monte Carlo (MC) [JL89, JR83a]. El método MC es una técnica estocástica que utiliza números aleatorios para obtener una aproximación estadística a la solución exacta de la BTE. Este método traza las trayectorias clásicas de los portadores en un dispositivo simulado por medio de la dispersión de cada partícula después de un período de vuelo libre determinado estocásticamente a través de las tasas de dispersión acumulativas. Estas tasas de dispersión se calculan cuánticamente e incluyen, entre otras, interacciones electrón-fonón, electrón-impureza y electrón-electrón. Una simulación Monte Carlo es por lo tanto una serie de vuelos libres intercalados con eventos de dispersión que cambian el momento y posiblemente la energía de la partículas. El movimiento de las partículas está acoplado a la solución de la ecuación de Poisson para permitir el cálculo actualizado de la fuerza que las dirige. La aproximación Monte Carlo es ampliamente usada en la simulación de dispositivos semiconductores [BAY⁺03, FSVF02], aunque el gran número de partículas que deben ser simuladas, la enorme cantidad de números aleatorios y el acoplamiento con la ecuación de Poisson hacen que este método sea muy costoso computacionalmente.

1.3.4. Transporte cuántico

En la cima de la jerarquía de los métodos de simulación están las técnicas de transporte cuántico. La modelización del transporte dentro de un sólido usando para ello una aproximación cuántica completa es muy costosa computacionalmente y poco práctica para simulaciones realistas de dispositivos. Una técnica que está ganando popularidad en el campo del transporte cuántico es la aproximación de las funciones de Green fuera del equilibrio (NEGF) [Dat00, MSA⁺05]. Utiliza un método matemático conocido como Funciones de Green para obtener la solución de un Hamiltoniano independiente del tiempo. La función de Green, a una energía dada, tiene dos entradas que pueden ser relacionadas con dos posiciones del espacio real permitiendo simular áreas de un transistor. Esta función considera la influencia de una perturbación que tiene lugar en una entrada sobre la otra entrada, y tiene, en teoría, la habilidad de modelar las propiedades físicas del sistema, tales como la densidad electrónica, la densidad de corriente y la densidad de estados. Una de las principales limitaciones del uso de NEGF es su elevadísimo coste computacional.

Como acabamos de comentar, las simulaciones completamente mecano-cuánticas son prohibitivas en términos de tiempo computacional, pero es posible incluir efectos cuánticos en simulaciones clásicas mediante correcciones cuánticas [AI89] con un coste computacional mucho menor. Estas correcciones permiten considerar efectos de confinamiento cuántico y ciertos aspectos del efecto túnel. Las correcciones cuánticas desempeñan un papel muy importante conforme los dispositivos son escalados más agresivamente a dimensiones nanométricas. Los dos métodos más conocidos para incluir correcciones cuánticas en simulaciones clásicas de dispositivos son las aproximaciones *density gradient* [AW01] y potencial efectivo [FAV00].

La aproximación *density gradient* introduce un potencial cuántico que proporciona un término adicional de arrastre a la expresión de la densidad de corriente. Este potencial cuántico es proporcional a la segunda derivada de la densidad de portadores y aleja a los electrones de variaciones pronunciadas en el potencial clásico.

La técnica del potencial efectivo representa a los portadores por medio de un paquete de ondas gaussiano de dispersión mínima. El potencial efectivo está relacionado con el potencial a través de una integral de convolución. El suavizado del potencial asociado con la operación de convolución representa los efectos mecánico-cuánticos que alejan a la concentración de electrones de la interfaz y reducen picos bruscos en el potencial.

Capítulo 2

Generación de mallas tetraédricas

Uno de los aspectos más importantes en la resolución de problemas usando métodos numéricos es la etapa de subdivisión del dominio del problema. Este proceso de discretización se realiza mediante la descomposición en elementos de tal forma que la solución se calcula únicamente en sus nodos. A estas descomposiciones del dominio de simulación se las conoce como “mallas”.

En la primera sección del capítulo presentaremos una clasificación sencilla de los distintos tipos de mallas. A continuación describiremos brevemente algunos de los principales métodos de generación de mallas tetraédricas utilizados actualmente. Estos métodos son, por ejemplo, los que tratan de obtener triangulaciones de Delaunay, el método *advancing front* y métodos basados en la descomposición recursiva del espacio, especialmente mediante el uso de árboles octales. En la siguiente parte del capítulo describiremos con algo más de detalle el uso del *octree* para generación de mallas tetraédricas así como la implementación que realizamos y algunos resultados obtenidos. En la siguiente sección del capítulo describiremos una aproximación a la generación de mallas atomísticas, que se caracterizan por tener los nodos en las posiciones de los átomos del material del dispositivo que queremos simular. Finalmente, para terminar el capítulo describiremos el uso de refinamiento adaptativo basado en bisección para aumentar la resolución de las mallas y mejorar así la convergencia de las simulaciones numéricas.

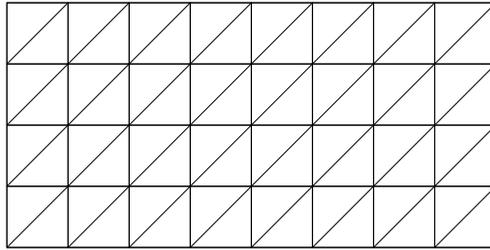


Figura 2.1: Ejemplo de malla estructurada.

2.1. Clasificación de mallas

Los diferentes métodos de generación de mallas permiten obtener discretizaciones con diversas propiedades [TMI⁺04, FG00] que a su vez permiten clasificarlas desde distintos puntos de vista. En este trabajo vamos a considerar dos criterios para la clasificación: estructuración de la malla y tipos de elementos que componen la malla.

Dependiendo de las características de la conectividad entre elementos podemos clasificar las mallas en dos grupos: estructuradas y no estructuradas. En las primeras los nodos tienen una conectividad regular. Esto supone que todos los nodos internos tienen el mismo número de vecinos. En la figura 2.1 se puede ver un ejemplo de malla estructurada en dos dimensiones en el que cada nodo tiene seis vecinos (salvo los que están sobre la frontera). En el caso de las mallas no estructuradas las conectividades son irregulares, pudiendo tener cada nodo diferente número de vecinos. Un ejemplo en dos dimensiones se puede ver en la figura 2.2. Las mallas no estructuradas presentan la ventaja de permitir una mayor densidad de nodos en una zona determinada del dominio del problema, pudiendo así obtener una solución más precisa usando un menor número de nodos. Aunque todas las mallas pueden ser englobadas en alguno de estos dos grupos, es común introducir un tercer tipo de malla, las mallas semiestructuradas. Estas mallas, pese a ser no estructuradas, tienen irregularidades bien definidas. Aunque un poco difuso, el término quedará más claro al describir métodos de generación de mallas basados en patrones.

Otra forma de clasificar las mallas es atendiendo al tipo de elementos utilizados. En el caso bidimensional tenemos fundamentalmente triángulos (figura 2.1) y cuadriláteros (figura 2.3). En tres dimensiones hay más elementos posibles: tetraedros, hexaedros, prismas, pirámides, etc. Además, también se usan a veces mallas mixtas, que mezclan varios tipos de elementos en la misma

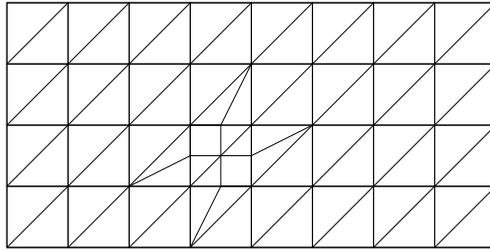


Figura 2.2: Ejemplo de malla no estructurada.

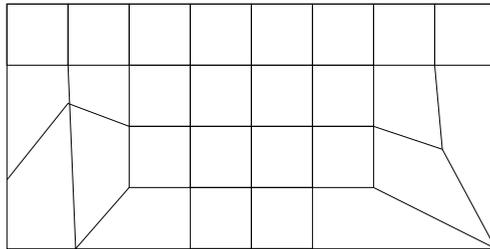


Figura 2.3: Ejemplo de malla de cuadriláteros.

malla. Mostramos un ejemplo en dos dimensiones en la figura 2.4.

2.2. Métodos de generación de mallas

En esta sección describiremos brevemente las principales aproximaciones para obtener una malla: generación de mallas de Delaunay, uso del método *advancing front* y uso de métodos basados en la descomposición recursiva del espacio, concretamente los basados en un *quadtrees* (en dos dimensiones) o un

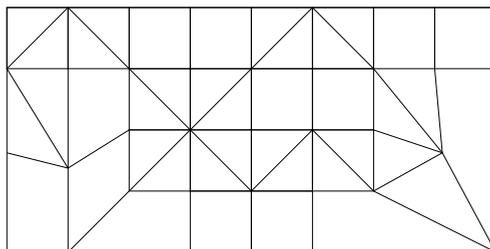


Figura 2.4: Ejemplo de malla con cuadriláteros y triángulos.

octree (en tres dimensiones).

Aunque realicemos una descripción independiente para cada uno, no se trata de métodos necesariamente independientes, sino que en algunas ocasiones se usan métodos híbridos. Comentaremos brevemente las características básicas de estos métodos en las secciones que siguen.

2.2.1. Triangulaciones de Delaunay

El origen de este método se encuentra en el trabajo de Boris Delaunay [Del34]. Se basó en trabajos de Dirichlet [Dir50] y Voronoï [Vor08], quienes habían trabajado en la construcción de diagramas de proximidad para conjuntos de puntos en dos y tres dimensiones. El diagrama de Voronoï en dos dimensiones divide el espacio de la siguiente manera:

Consideremos los puntos p_1, p_2, \dots, p_n en el plano. El polígono de Voronoï V_i es la región del plano cuyos puntos están más cerca de p_i que de cualquier otro punto.

Un vértice de un polígono de Voronoï es compartido por tres polígonos adyacentes. Uniendo los tres puntos asociados a estos tres polígonos formamos un triángulo. El conjunto de estos triángulos se llama triangulación de Delaunay. Mostramos un ejemplo de este diagrama en la figura 2.5.

Tres de las características más interesantes de las triangulaciones de Delaunay son las siguientes:

1. En general, es única. Sin embargo, algunas disposiciones de nodos pueden dar lugar a degeneración.
2. Satisface el criterio del círculo vacío. Esto es, la circunferencia circunscrita de cada triángulo no contiene ningún vértice de la triangulación aparte de los del propio triángulo (ver figuras 2.6(a) y 2.6(b)).
3. De todas las triangulaciones de un conjunto de puntos, la de Delaunay maximiza el ángulo mínimo. Esta propiedad se pierde al pasar a 3D.

Como ya comentamos, la triangulación de Delaunay está asociada a un conjunto de puntos. Podemos adoptar dos estrategias para crear estos puntos: generarlos todos antes de empezar el proceso (métodos no incrementales) o generarlos junto con la generación de los elementos (métodos incrementales). En este segundo caso se aprovecha que para insertar puntos en una triangulación de forma que siga siendo de Delaunay sólo hace falta realizar operaciones locales. Los puntos generados determinarán la densidad final de la malla.

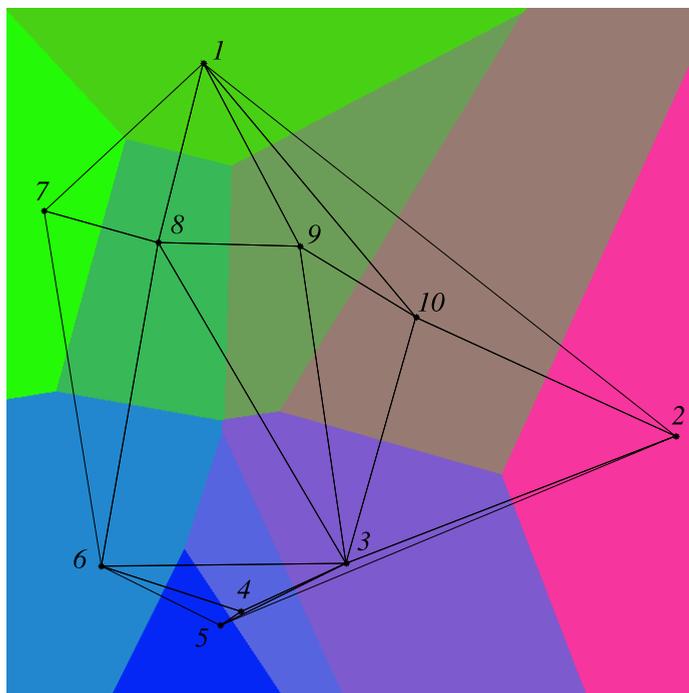


Figura 2.5: Diagrama de Voronoi y triangulación de Delaunay asociada.

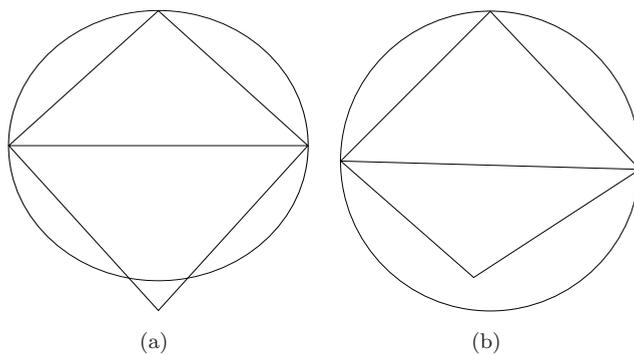


Figura 2.6: Triángulos que (a) verifican la condición del círculo vacío y que (b) no la verifican.

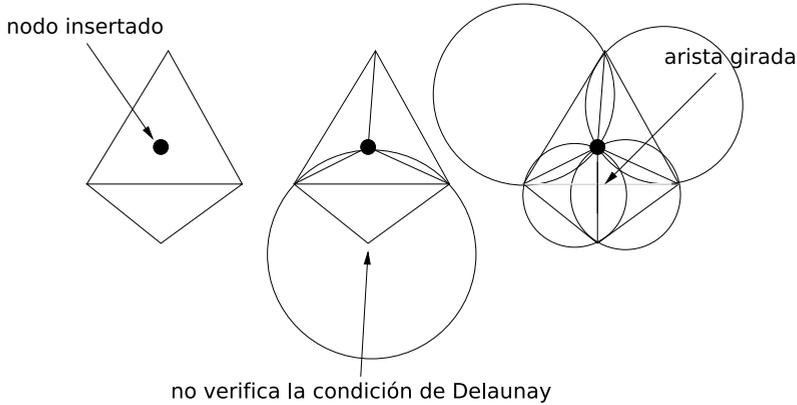


Figura 2.7: Ejemplo del algoritmo de giro de aristas.

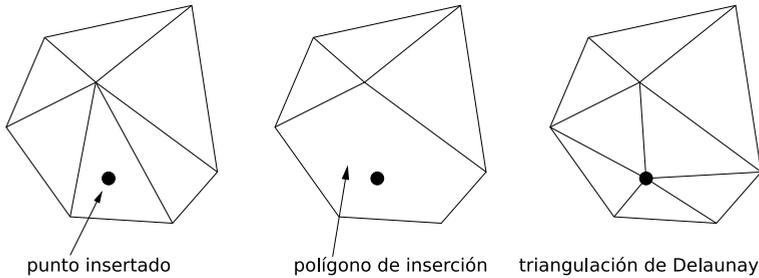


Figura 2.8: Ejemplo del método del polígono de inserción.

Si queremos añadir algún nodo, en los algoritmos no incrementales hay que rehacer la triangulación para que siga verificando el criterio de círculo vacío. Para superficies existen dos métodos principales para hacer esto: el algoritmo de giro de aristas [Law77] y el del polígono de inserción [GS78]. Mostramos un ejemplo de estos métodos en las figuras 2.7 y 2.8. En tres dimensiones existen generalizaciones de ambos métodos, aunque en este caso el del poliedro de inserción, generalización del método del polígono de inserción, es más fácil de implementar [Bow81, Wat81].

Para la aplicación del método de elementos finitos es importante que la triangulación esté ajustada a la frontera del dominio del problema. Si únicamente creamos la triangulación de Delaunay de un conjunto de puntos, no podremos asegurar que esto se verifique. Para ello se pueden seguir dos estrategias: ajustar a la frontera permitiendo que algunos elementos dejen de

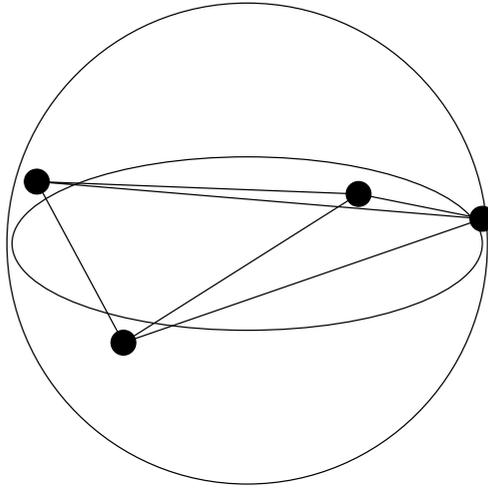


Figura 2.9: Tetraedro *sliver*.

satisfacer la condición de Delaunay o modificar la triangulación de forma que se mantengan sus propiedades. En el primer caso, en 2D, basta únicamente con intercambio de aristas para adaptarse a la frontera [Wat90]. Para conseguir la adaptación evitando que la triangulación deje de ser de Delaunay hace falta, en general, incluir nuevos puntos en la frontera. Aparte de modificar la triangulación una vez generada para ajustarse a la frontera también es posible incluir esto como condición en su creación. En este caso hablaríamos de Triangulaciones de Delaunay Ligadas (*Constrained Delaunay Triangulations, CDT*), aunque todavía están limitadas al caso bidimensional [LL86].

En tres dimensiones, la recuperación de la frontera no es tan sencilla como en el caso bidimensional, siendo en general necesaria la inserción de puntos adicionales a los de la malla original para lograr una adaptación que, en general, no va a seguir verificando el criterio de Delaunay [GHS91, WH94].

Una malla de triángulos (o tetraedros en 3D) que verifique la condición de Delaunay no tiene que estar compuesta por elementos de buena calidad. Un ejemplo son los tetraedros llamados *slivers*, caracterizados por un volumen arbitrariamente pequeño pero con las seis aristas de longitud semejante. Una forma de generarlos es la siguiente (ver figura 2.9): se sitúan cuatro vértices sobre el ecuador de una esfera y se desplazan ligeramente.

En muchos casos hace falta una etapa posterior a la generación de la malla que elimine este tipo de elementos [HVKR03]. Las operaciones a realizar sobre

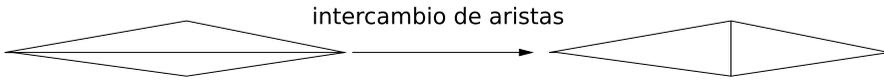


Figura 2.10: Intercambio de aristas en 2D.

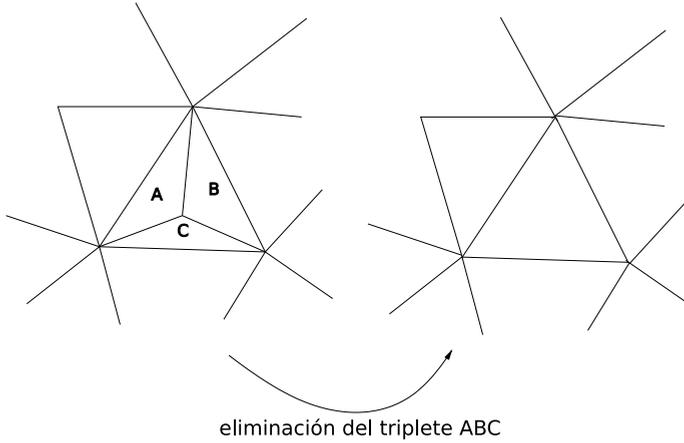


Figura 2.11: Eliminación de tripletes.

la malla pueden dividirse en dos grupos: operaciones geométricas y operaciones topológicas. Las primeras no cambian la topología de la malla, mientras que las segundas sí lo hacen, por ejemplo cambiando conectividades de elementos. Entre estas operaciones se encuentran operaciones como el cambio de aristas, caras, o, en dos dimensiones, la eliminación de tripletes. En las figuras 2.10 y 2.11 mostramos ejemplos de estas operaciones en el caso bidimensional.

Es posible generalizar estas mallas generadas con el método de Delaunay a casos en los que interesa reflejar anisotropía. Para ello hay que introducir nuevas condiciones. Los nodos que se van añadiendo deben ser colocados a distancias que reflejen la anisotropía deseada. Además, el criterio de la esfera vacía, equivalente tridimensional al criterio del círculo vacío, debe ser revisado. En este caso, como centro de la “esfera” usaríamos el punto equidistante de los cuatro vértices en la métrica dada. El criterio de Delaunay se satisface si ningún vértice, aparte de los del propio tetraedro, está a una distancia del centro menor o igual que la dada por el radio determinado por la métrica que define la anisotropía.

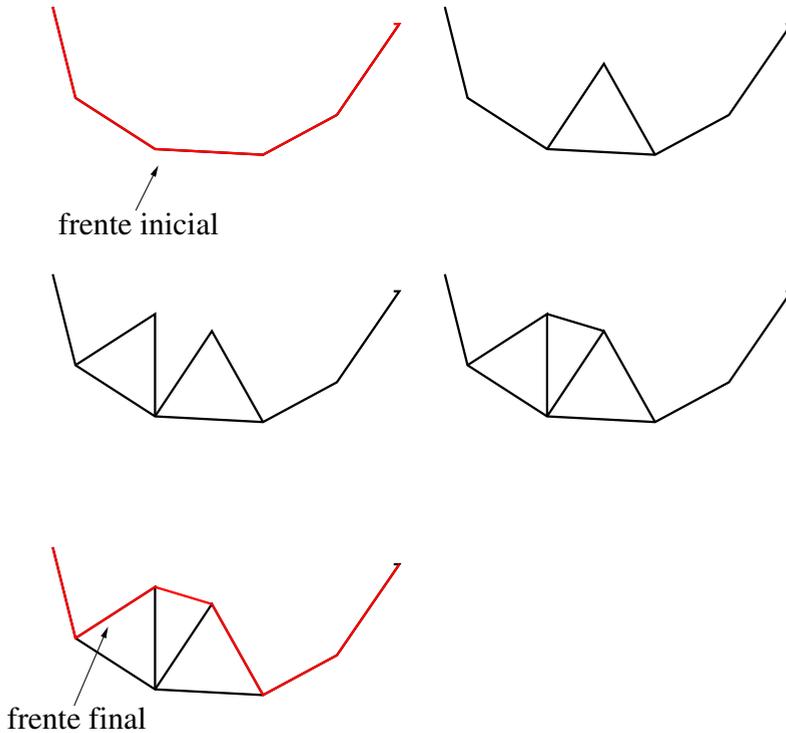


Figura 2.12: Ejemplo de inicio del proceso de triangulación usando el método *advancing front*.

2.2.2. Método *advancing front*

El primer algoritmo basado en este método fue presentado en 1985 por Lo [Lo85], aunque la mayoría de la investigación posterior se centró en un nuevo algoritmo desarrollado por Peraire *et al.* [PVMZ87]. Este último algoritmo fue modificado y adaptado para mallas de cuadriláteros, mallas en tres dimensiones, mallas de superficies no planas, mallas anisótropas, etc.

En dos dimensiones este método consiste en lo siguiente. Se crea un frente inicial con segmentos que definen la frontera. Se van añadiendo nodos que se conectan con estos segmentos de forma que se crean triángulos. Después de cada inserción de un nodo y de la subsecuente creación de un triángulo, se cambia el frente (se eliminan segmentos utilizados y se añaden segmentos nuevos) y se repite el proceso hasta que no queden segmentos en el frente. Mostramos en la figura 2.12 un ejemplo de inicio del proceso de triangulación.

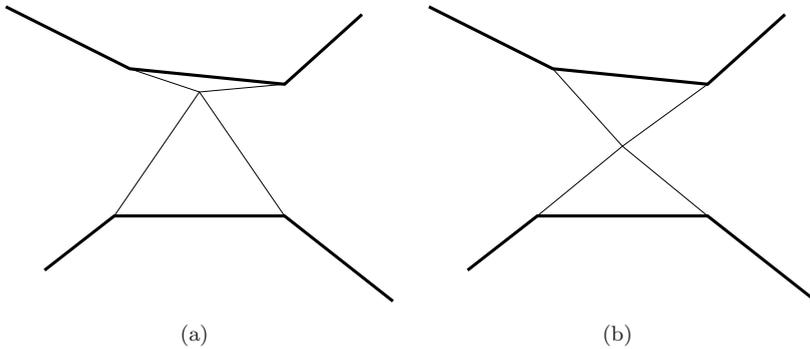


Figura 2.13: Triángulos generados (a) sin considerar la proximidad de otro segmento del frente y (b) considerando este factor.

En tres dimensiones el proceso es equivalente [LP88]. Se crea un frente inicial con una triangulación de la frontera. Se van añadiendo nodos en posiciones adecuadas con los que se forman tetraedros y se actualiza el frente. Este proceso se repite hasta terminar el frente.

A continuación comentamos algunos de los puntos más importantes en la implementación de un algoritmo que genere mallas usando el método *advancing front*. Dada una cara (segmento en 2D) hay que determinar la localización ideal del cuarto nodo (tercero en 2D), que puede ser nuevo o un nodo que ya esté en el frente. Para esto se definen factores de forma. Un ejemplo de esta medida para el caso bidimensional sería el siguiente:

$$F_i = 1 - \frac{\sum_{j=1}^3 e_j e_{((j+1) \bmod 3)}}{3} \quad (2.1)$$

donde e_j es el vector unidad de la arista j del elemento i . Esta expresión tiene un mínimo en cero cuando los tres nodos están alineados, mientras que en el caso de un triángulo equilátero $F_i = 0,5$. Aparte de esto hay que considerar otros factores a la hora de seleccionar el nodo. Uno de ellos es la proximidad a un elemento del frente. En las figuras 2.13(a) y 2.13(b) vemos un ejemplo para el caso bidimensional. Se puede ver que si no tenemos en cuenta este factor la calidad del elemento formado puede ser muy baja. Si no estamos en estas situaciones, el tetraedro con un mayor factor de forma es el generado. Otro caso que hay que tener en cuenta en 3D es la intersección de superficies. Al añadir un nodo hay que tener en cuenta que el tetraedro generado no debe intersectar el frente. Además, para determinar la situación de los nodos añadidos hay que

tener en cuenta el nivel de refinado necesario en cada zona del dominio. Para tener este factor en cuenta se pueden usar, por ejemplo, puntos de control con información del tamaño de tetraedro requerido.

Un problema de este método es que no siempre garantiza la convergencia para dominios tridimensionales complejos. Su estabilidad y eficiencia pueden mejorarse, por ejemplo, si se usa el método acoplado con triangulaciones de Delaunay.

2.2.3. Métodos basados en la descomposición recursiva del espacio

Los principales métodos de esta familia están basados en la utilización de un *quadtree* en el caso bidimensional o de un *octree* en el caso tridimensional.

Quadtree

El *quadtree*, originalmente usado para aproximar geometrías de objetos [Knu75], fue utilizado por primera vez para la generación de mallas de elementos finitos en los años 80 [YS83, She88].

Un *quadtree* es una estructura de datos que representa una subdivisión recursiva del espacio en dos dimensiones. Esta estructura en forma de árbol permite realizar de forma eficiente ciertas operaciones sobre la malla (como por ejemplo la búsqueda).

Al inicio de este método el dominio que queremos discretizar se encierra en un rectángulo. Este rectángulo, el cuadrante raíz, se subdivide recursivamente guiado por la geometría de la región de interés y algún criterio sobre el refinamiento deseado en las diferentes zonas del dominio que queremos mallar. Para adaptarse a la frontera del dominio algunos cuadrantes terminales (cuadrantes que no tienen hijos) deben ser deformados. Esto suele requerir bastantes cálculos de intersecciones, por lo que se introdujeron variantes del método para tratar con las fronteras. En la figura 2.14 mostramos la estructura de árbol del *quadtree* y la descomposición equivalente del plano.

Una vez obtenido el *quadtree* la generación de los triángulos es directa para los cuadrantes interiores (en la figura 2.15 se pueden ver ejemplos de triangulación de cuadrantes terminales) y algo más compleja para los que forman la frontera del dominio. Para mantener esta simplicidad se puede poner una restricción sobre la diferencia de profundidad en el árbol de dos cuadrantes vecinos.

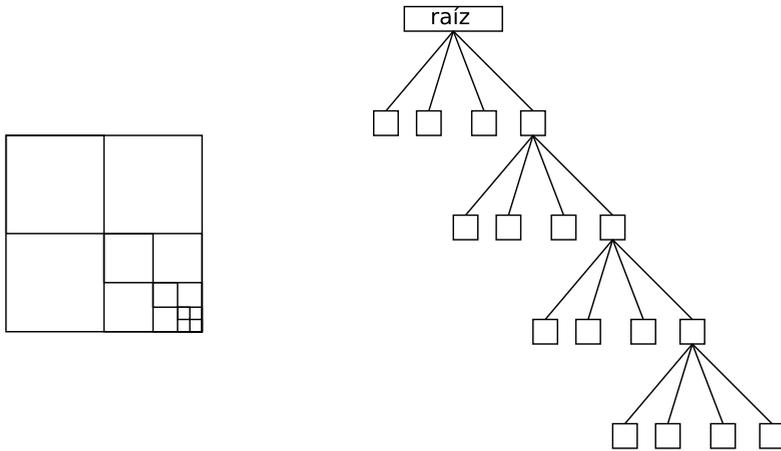


Figura 2.14: Descomposición de un cuadrado y estructura en árbol equivalente.

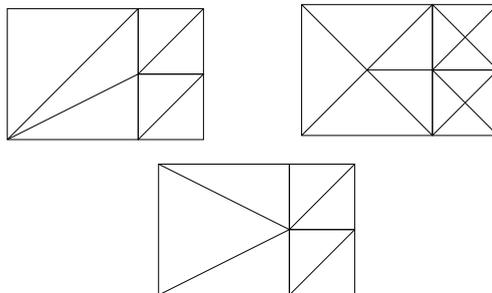


Figura 2.15: Tres formas posibles de triangular cuadrantes terminales.

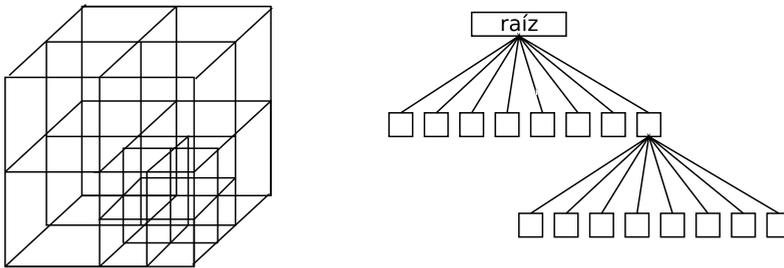


Figura 2.16: Descomposición de un cubo y árbol equivalente.

Octree

El *octree* o árbol octal [YS84, SG91], equivalente tridimensional del *quadtree*, es una estructura de datos que representa la descomposición recursiva de un espacio en tres dimensiones (ver figura 2.16).

Los algoritmos que usan un *octree* para la generación de mallas tienen una base equivalente a los del caso bidimensional. Inicialmente la región de interés se encierra por un paralelepípedo rectangular. Este paralelepípedo, el octante raíz, se subdivide recursivamente guiado por la frontera a la que queremos que se adapte y por algún criterio de refinado que especifiquemos sobre el volumen de interés. Para hacer que las transiciones entre los tamaños de los elementos sean suaves y para facilitar la generación de los tetraedros se suele limitar el cambio de nivel de profundidad en el árbol entre dos vecinos a uno. Además, en el caso tridimensional el ajuste a las fronteras es bastante más complejo que en el caso bidimensional, por lo que han surgido diferentes métodos para tratar con fronteras complejas. Algunos de estos métodos pierden en algunas zonas la estructura de *octree* para poder usar otros métodos en el ajuste a la frontera.

Una vez generado el *octree* hay varios métodos de construir la malla tetraédrica. El más sencillo está basado en el uso de patrones para generar tetraedros embebidos en los octantes terminales. Podemos ver un ejemplo de este tipo de generación para un octante terminal en la figura 2.17. Este es el método más rápido, pero es también el que tiene más problemas cuando tratamos el problema de ajuste a fronteras, por lo que no se suele usar este método “puro” cuando se trabaja con fronteras complejas. Puede usarse una aproximación de *advancing front* usando los puntos del *octree* como guía para determinar la situación óptima de los nodos a insertar. Otra posibilidad es hacer la triangulación de Delaunay de los nodos de los octantes terminales o

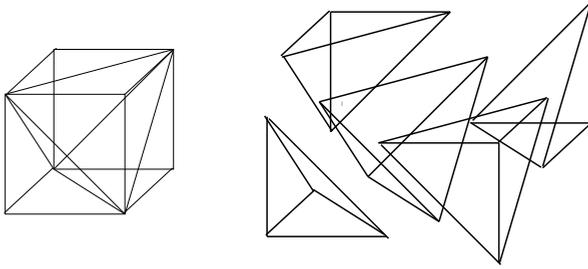


Figura 2.17: Descomposición de un cubo en cinco tetraedros.

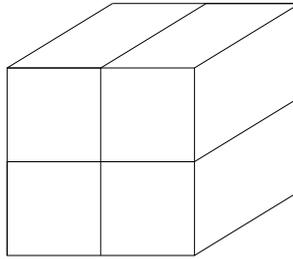


Figura 2.18: Descomposición de un octante en cuatro hijos.

mezclar algunos de estos métodos. Además, también es posible generalizar la construcción del *octree* permitiendo algunas divisiones en un número de hijos distinto de ocho. En la figura 2.18 mostramos un ejemplo de división de un octante en cuatro hijos en lugar de los ocho del caso estándar.

2.3. Generación de mallas usando un *octree*

En esta sección realizamos una descripción de las distintas partes que componen el programa de generación de mallas utilizado. En primer lugar describimos la fase de preprocesado. Posteriormente tratamos la generación del *octree* y el algoritmo utilizado para el ajuste a regiones compuestas por paralelepípedos. Por último presentamos una descripción de los diferentes algoritmos implementados para generar la malla tetraédrica de la región a partir del *octree*.

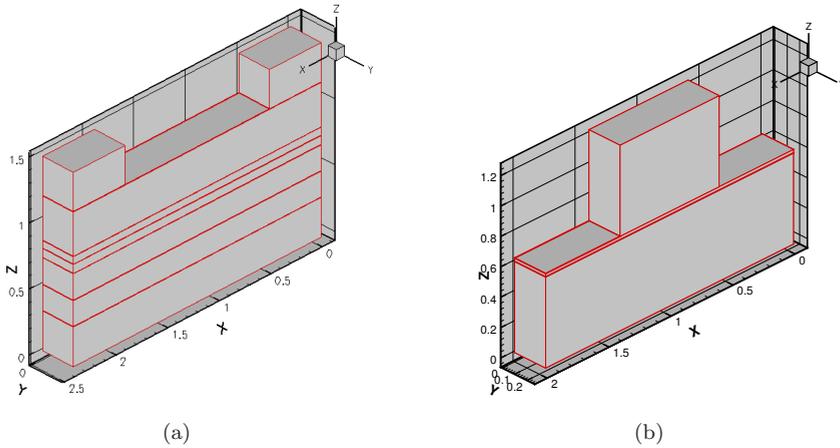


Figura 2.19: Regiones usadas para definir (a) un HEMT y (b) un MOSFET con puerta de polisilicio.

2.3.1. Construcción del *octree*

La primera etapa de la generación del *octree* se corresponde con el método genérico descrito en la sección anterior. El primer paso es la creación de un octante raíz O_r , es decir, un paralelepípedo rectangular que engloba a toda la estructura. En una segunda parte se subdivide este octante siguiendo unos criterios especificados. Si después de esto hay octantes vecinos con dos o más niveles de refinamiento es habitual forzar refinamientos donde sea necesario para que haya como máximo un solo nivel de diferencia entre cualquier par de vecinos. Esta restricción suele introducirse para evitar tetraedros de calidades bajas. Finalmente, para adaptarse a las fronteras, los octantes que intersectan con la frontera se tratan de forma especial y en general bastante compleja [Yan94, MV00].

Hemos modificado este algoritmo clásico aprovechándonos del hecho de que gran parte de las regiones de interés están formadas por unión de paralelepípedos rectangulares, regiones conocidas como tipo Manhattan. Dos ejemplos se muestran en las figuras 2.19(a) y 2.19(b).

Para la descripción de la generación del *octree* final vamos a introducir alguna notación. Llamaremos O al conjunto de todos los octantes del *octree*, tanto los internos O^* como los externos O_e a la región Ω que deseamos des-

componer en tetraedros y llamaremos o_j a los diferentes octantes del *octree*:

$$O = O^* \cup O_e,$$

con

$$O^* = \{o_j \in O / o_j \cap \Omega \neq \emptyset\},$$

y

$$O_e = O - O^*.$$

Además, dividimos el conjunto O^* en dos subconjuntos O_d y O_t tales que

$$O^* = O_d \cup O_t,$$

con

$$O_t = \{o_j \in O^* / \mathcal{D}(o_j) = \emptyset\},$$

donde $\mathcal{D}(o_j)$ es la descendencia del octante o_j . Finalmente, llamaremos \mathbf{P} al conjunto de todos los planos que forman la estructura.

Podemos dividir la generación del *octree* en tres pasos. En el primero generamos un *octree* intermedio uniformemente refinado O_i con una profundidad especificada n_D que dependerá del tamaño de las diferentes capas del dispositivo. En este punto llevamos a cabo la adaptación a los planos de la estructura, tanto a los internos como a los externos. Por último, refinamos el árbol octal hasta el nivel requerido en cada región del dispositivo, especificada como punto, línea, plano o volumen. En resumen, los pasos son los siguientes:

1. $O_r \longrightarrow O_i$ de tal forma que $\mathcal{N}(o_j) = n_D \quad \forall o_j \in O_{i,t}$, donde $\mathcal{N}(o_j)$ es el nivel en el árbol del octante o_j .
2. Deformar $o_j \in O_{i,t} / o_j \cap \mathbf{P} \neq \emptyset$ y transmitir estas deformaciones a sus ascendentes para evitar problemas posteriores en el paso 3.
3. Construir el *octree* final O^* a partir de O_i^* usando las regiones de control de refinamiento.

En la figura 2.20 se puede ver el flujo de la generación del *octree*.

En el paso 2 hay que notar que la deformación puede ser hecha en dos sentidos dependiendo de la cara del octante más próxima al plano. En el caso de que sea la cara interior la más próxima hay que eliminar el octante, ya que pasará a estar fuera de la estructura tras la deformación. En el otro caso no hace falta ninguna operación aparte de la deformación.

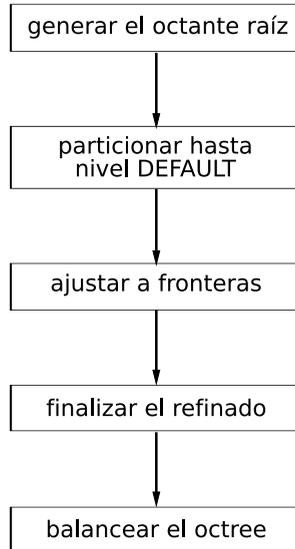


Figura 2.20: Diagrama de flujo correspondiente a la generación del *octree*.

El paso 1 se lleva a cabo de la forma estándar descrita anteriormente. El paso 2 empieza con la detección de octantes terminales que intersectan con alguna de las fronteras. Cada uno de estos octantes se deforma para ajustarse a los planos de la región y en este caso se recorre su ascendencia modificándola cuando sea necesario. Después de este paso tenemos un *octree* con nodos del mismo nivel pertenecientes a paralelepípedos rectangulares de diferente forma y volumen. La figura 2.21 muestra los octantes terminales después del paso 2 en un ejemplo bidimensional sencillo. Si no transmitiéramos la deformación de los octantes en este nivel intermedio $O_{i,t}$ a su ascendencia, los puntos de control podrían ser asignados de forma incorrecta. El origen de este problema se encuentra en la forma en la que los puntos de control se distribuyen desde el octante raíz hasta los octantes terminales de $O_{i,t}$. Un punto de control en un octante de nivel $n_j < n_D$ se transmite a uno de sus descendientes de acuerdo a las coordenadas de los hijos. En los niveles $n_D - 2$ y anteriores, si los octantes no se deforman, se podría transferir un punto de control a un octante que lo contiene pero con descendencia que no lo contiene, generando así una malla diferente a la deseada. La figura 2.22 presenta este problema de forma esquemática con un ejemplo en dos dimensiones.

Una vez que el *octree* está terminado existen varios métodos para la ge-

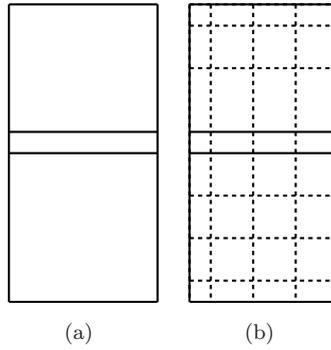


Figura 2.21: Ejemplo 2D de (a) una región y (b) los octantes terminales después del ajuste a las fronteras usando $n_D = 3$.

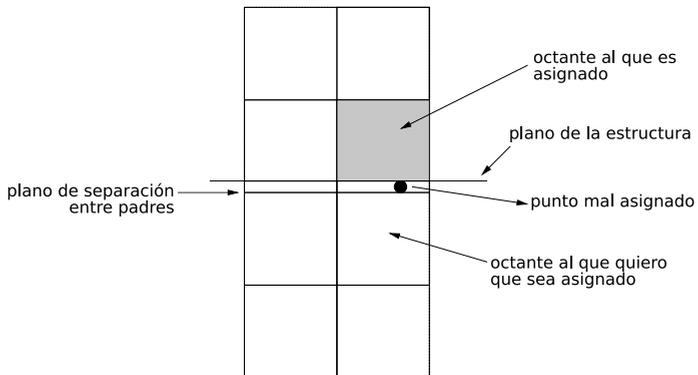


Figura 2.22: Ejemplo de asignación errónea en caso no transmitir la deformación a los padres.

neración de la malla tetraédrica final. El que hemos utilizado, el más sencillo y rápido, es el basado en patrones, aunque la generación de una malla que satisfaga el criterio de Delaunay con los nodos del *octree* también es posible. Sin embargo, este método podría requerir el uso de etapas de post-procesado para asegurar el ajuste a fronteras con la subsiguiente pérdida de la propiedad de Delaunay.

2.3.2. Descomposición en tetraedros de los octantes terminales

Una vez generado el *octree* ya podemos pasar a generar la malla de tetraedros a partir de la información que contiene. Para hacer esto hay muchas posibilidades. Hemos implementado cinco algoritmos básicos para generar las mallas tetraédricas. El primero de ellos crea, en el caso de vecinos en mismo nivel de profundidad del *octree*, seis tetraedros por octante terminal (*algoritmo A*). Otro genera en esta situación veinticuatro tetraedros por octante (*algoritmo E*). Los tres algoritmos restantes generan una cantidad intermedia de elementos en el caso de octantes sin vecinos de nivel diferente. Estos algoritmos producen doce, dieciséis y veinte tetraedros (*algoritmos B, C y D* respectivamente). Existen otros patrones similares (con cinco, catorce, dieciocho o veintidós tetraedros), pero no permiten definir los tetraedros de manera independiente en cada octante, lo que resulta útil en ciertos casos, como puede ser en una versión paralela del programa. Antes de comentar estos algoritmos describiremos la generación de nodos, base de la generación de tetraedros y esencialmente igual en los cinco casos.

Los algoritmos consisten en dos partes. En la primera se generan los nodos de la malla, mientras que en la segunda se definen las conectividades, lo que termina de generar los tetraedros. En la primera parte es necesario el conocimiento del nivel en el *octree* de cada octante y sus vecinos, pero el segundo paso es completamente independiente para cada octante, consecuencia del tipo de algoritmos de generación de tetraedros elegidos.

A continuación describimos brevemente los cinco algoritmos implementados y sus características básicas. Las propiedades de las mallas obtenidas con cada método se mostrarán más adelante con algunos ejemplos. Las figuras 2.23(a)-2.23(e) muestran las conectividades de un octante terminal para cada algoritmo en el caso de que no tenga vecinos con mayor profundidad en el árbol. La figura 2.24 representa un ejemplo de la descomposición usando el algoritmo A en el caso de que tenga un vecino con mayor nivel de refinamiento.

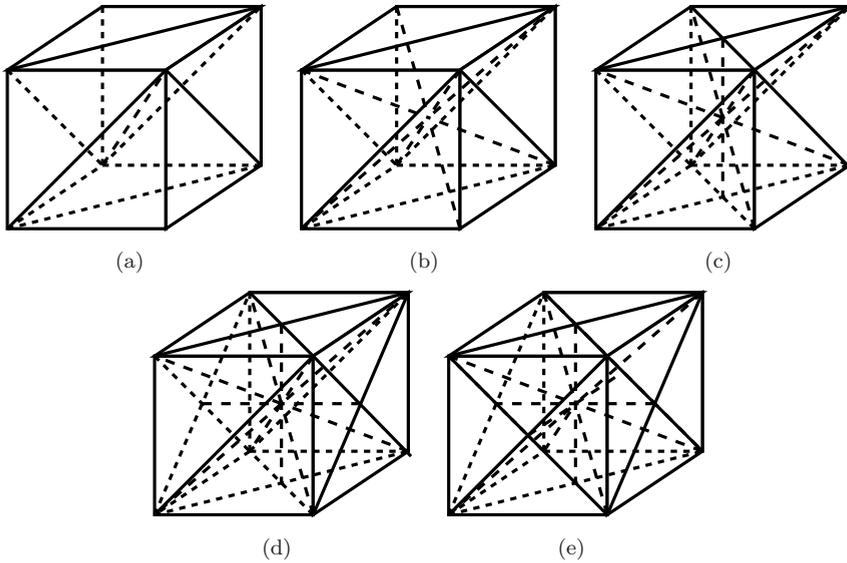


Figura 2.23: Conectividades entre nodos en un octante terminal para (a) el algoritmo A, (b) el algoritmo B, (c) el algoritmo C, (d) el algoritmo D y (e) el algoritmo E.

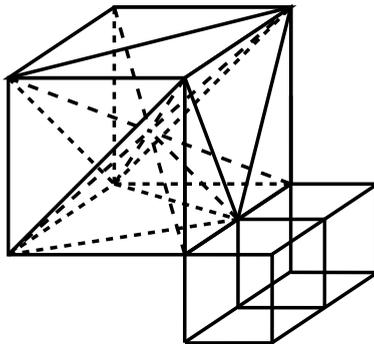


Figura 2.24: Tetraedros generados usando el algoritmo A en el caso de tener un vecino más refinado.

Algoritmo A

Este algoritmo genera el menor número de nodos, ocho por octante, y tetraedros, seis por octante, si no hay vecinos con nivel de refinamiento mayor. En el caso de que haya octantes vecinos de diferente nivel introducimos un nodo central en el octante menos refinado para evitar un número excesivo de casos a codificar, ya que no produciría beneficios importantes y el proceso sería muy propenso a errores.

Algoritmo B

Este algoritmo genera triangulaciones superficiales equivalentes al anterior, pero se introduce un nodo central en cada octante, lo que aumenta el número de nodos de las mallas. En el mejor caso se generan nueve nodos y doce tetraedros por octante terminal.

Algoritmo C

En este caso se introduce un cierto nivel de anisotropía en el mallado de un octante al cambiar la triangulación superficial en dos caras opuestas de cada octante. A partir de esta triangulación se generan tetraedros en la misma forma que en el caso anterior. El número de nodos y tetraedros resultantes en el mejor de los casos son once y dieciséis respectivamente.

Algoritmo D

Este algoritmo es la extensión lógica del anterior, cambiando la triangulación superficial de otras dos caras opuestas, lo que resulta en trece nodos y veinte tetraedros por octante terminal si no hay vecinos de diferente nivel en el árbol.

Algoritmo E

En este caso cambiamos completamente la triangulación superficial inicial (algoritmos A y B) de los octantes. Dividimos cada cara de los octantes terminales en cuatro triángulos, lo que supone la creación de quince nodos y veinticuatro tetraedros en el mejor de los casos.

La tabla 2.1 presenta el número de nodos y elementos correspondientes a cada algoritmo al generar una malla de una región utilizada para definir un

transistor HEMT (*High Electron Mobility Transistor*), mostrada en la figura 2.19(a). El *octree* resultante tiene 19536 octantes terminales. Los datos mostrados en la tabla dan una idea de la diferencia entre los algoritmos en términos del tamaño de la malla para una misma región y longitud característica de las aristas de los elementos semejante.

| Algoritmo | Nodos | Elementos |
|-----------|--------|-----------|
| A | 26726 | 144608 |
| B | 44630 | 252032 |
| C | 66758 | 327696 |
| D | 87924 | 405840 |
| E | 108532 | 497024 |

Tabla 2.1: Nodos y elementos generados por los diferentes algoritmos de descomposición en tetraedros para un mismo *octree*.

2.3.3. Ejemplos: tiempos de generación de las mallas

Generación del *octree*

Esta es la parte más costosa del algoritmo. Para analizarla vamos a considerar tres factores: generación del *octree* uniforme inicial (incluye adaptación a fronteras), balanceado e influencia del número de planos que componen la región. La figura 2.25 muestra la dependencia con el número de octantes terminales del tiempo de generación de un *octree* para un número fijo de planos de la estructura y sin considerar el tiempo de balanceo. La figura 2.26 presenta la variación de los tiempos de adaptación a las fronteras y balanceo con el número de octantes terminales. La importancia del primero es muy notable, suponiendo cerca del 40 % de la generación del *octree*. Por otro lado, el balanceo supone sólo una pequeña fracción del tiempo total, en torno al 3%. Además, podemos ver como existe una dependencia próxima a la lineal en las tres fases de generación para los tamaños de problema que estamos considerando.

Finalmente, analizamos la influencia del número de planos que definen la región. Para esto generamos *octrees* de tamaño fijo utilizando estructuras similares con un número diferente de fronteras internas. La figura 2.27 muestra los resultados obtenidos y demuestra que la influencia es muy pequeña para cambios moderados en el número de planos, ya que un incremento de este cercano al 30 % supone un incremento en el tiempo de ajuste cercano al 2%.

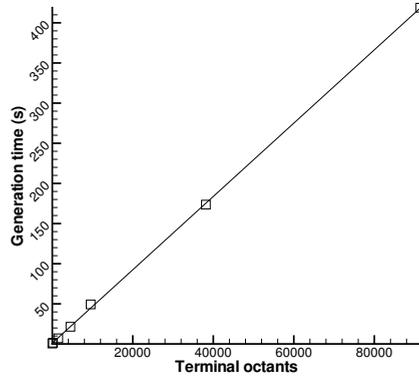


Figura 2.25: Dependencia del tiempo de generación del *octree* con su tamaño.

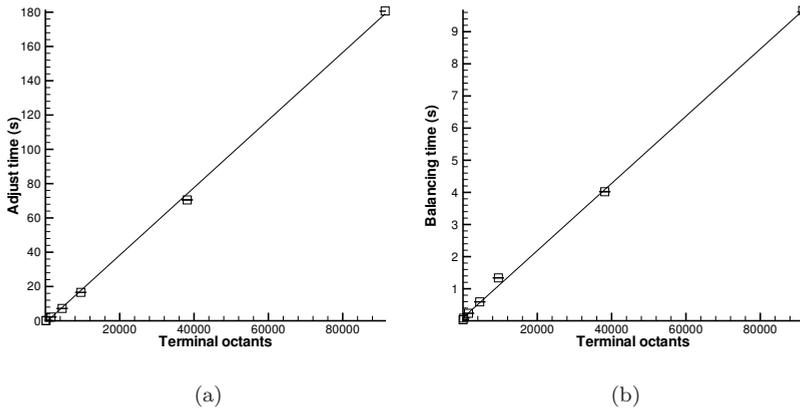


Figura 2.26: Dependencia con el tamaño del *octree* de los tiempos de (a) adaptación a las fronteras y (b) balanceo.

Generación de los tetraedros

Ahora analizamos los tiempos correspondientes a la generación de los nodos y los tetraedros con los diferentes algoritmos presentados. Para hacer esta comparación utilizaremos cinco *octrees* con cada malla. Sus tamaños son de 11360, 18528, 37056, 55584 y 92640 octantes terminales. Esto significa que no

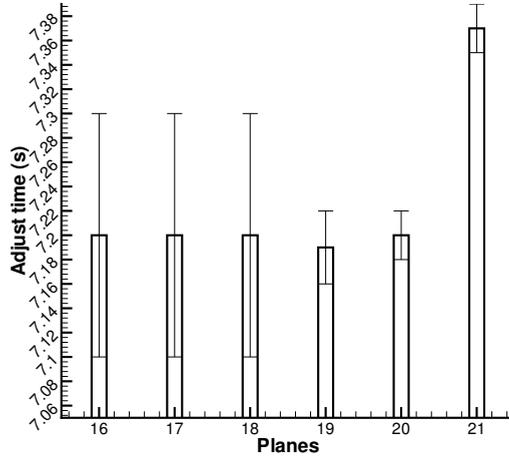


Figura 2.27: Dependencia del tiempo de ajuste con el número de planos que definen la región.

vamos a comparar mallas del mismo tamaño en el sentido de número de nodos y tetraedros, que ya hemos visto que son diferentes para un mismo *octree*. Lo que comparamos son mallas con una resolución espacial aproximada, ya que esta está dada esencialmente por el *octree*. Las figuras 2.28(a)-2.28(e) presentan los tiempos resultantes y su dependencia con el tamaño de la malla. Se puede ver la misma linealidad en todos los casos, lo que, junto con los resultados obtenidos para la generación del *octree*, nos muestra que el generador de mallas tiene una dependencia cercana a la linealidad para los tamaños de mallas considerados. Además, también podemos ver que su contribución al tiempo total de generación de la malla es bastante pequeña, inferior al 5%.

Para completar la comparación entre los diferentes algoritmos, mostramos la relación entre los tiempos para un *octree* dado en las figuras 2.29(a) y 2.29(b). La diferencia relativa es mucho más importante en el caso de la generación de tetraedros que en el de los nodos. Esto es debido a que en este último caso la generación está dominada por las operaciones de búsqueda de vecinos necesarias, las mismas para todos los algoritmos estudiados, mientras que la definición de los tetraedros es un proceso independiente para cada octante terminal.

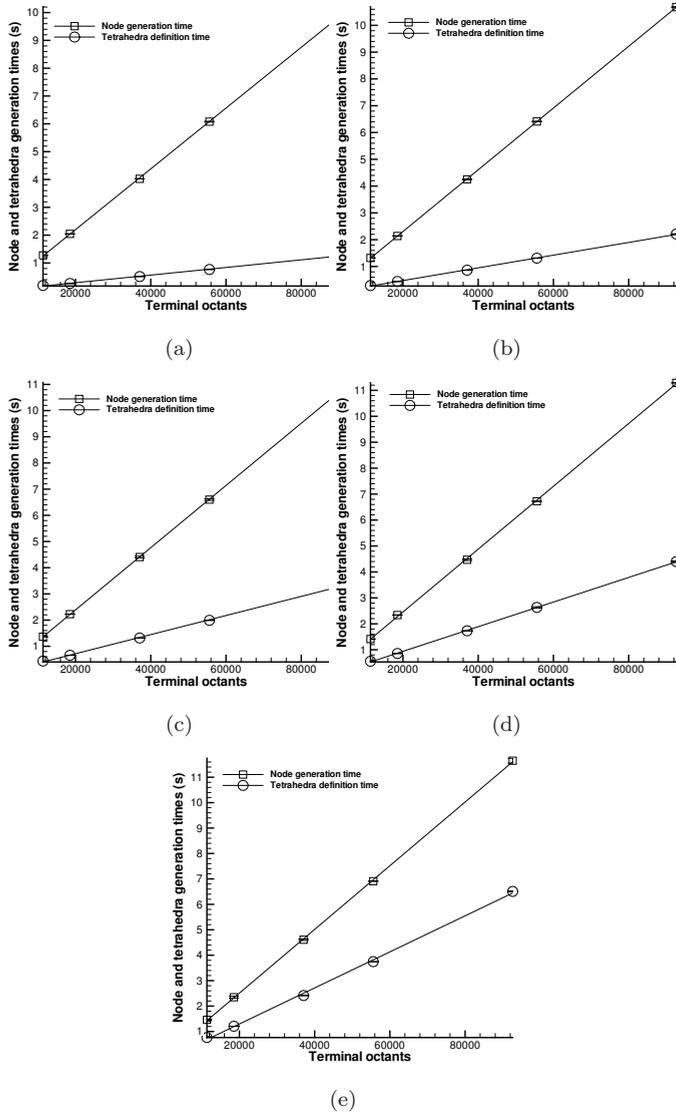


Figura 2.28: Tiempos de generación de nodos y tetraedros para diferentes tamaños del *octree* usando (a) el algoritmo A, (b) el algoritmo B, (c) el algoritmo C, (d) el algoritmo D y (e) el algoritmo E.

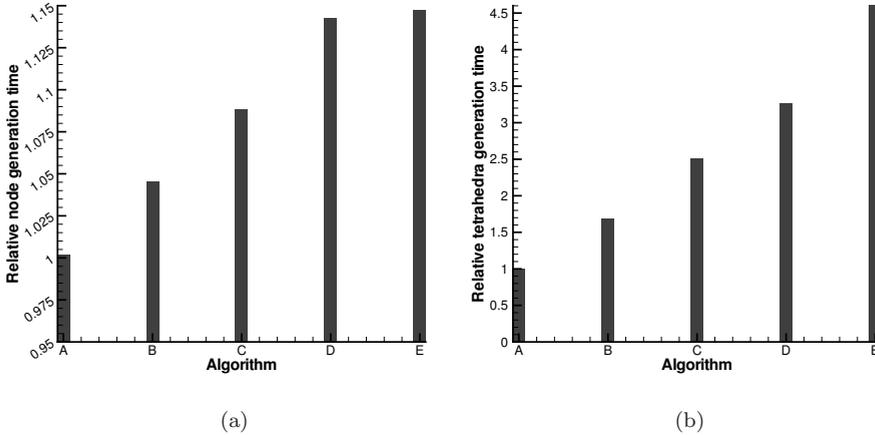


Figura 2.29: Tiempos relativos de generación de (a) nodos y (b) tetraedros.

2.3.4. Ejemplos: calidad de las mallas

En el caso de las simulaciones, ya sea usando el método de los elementos finitos u otro método, la forma de los tetraedros es muy importante, ya que determina en gran medida la convergencia de la solución. Sin embargo, la malla óptima depende del problema en consideración, por lo que no puede definirse un sistema de medida de calidades universal. Para evaluar la calidad de las diferentes mallas hemos escogido considerar únicamente la forma de los tetraedros. Esto también resulta problemático en algunas ocasiones, ya que no hay un sistema estándar de medir la calidad a través de la forma de los elementos. Esto hace que existan muchas medidas para evaluar la calidad de las mallas [DLGC98]. Hemos optado por utilizar como factor de forma el *aspect ratio*, definido de la siguiente manera:

$$\gamma = \frac{12}{\sqrt{6}} \frac{r_{in}}{\max_{i,j} l_{ij}}, \quad (2.2)$$

donde r_{in} es el radio de la esfera inscrita y l_{ij} son las longitudes de las aristas del tetraedro. El radio de la esfera inscrita se puede calcular, salvo un factor multiplicativo constante, como el cociente entre el volumen del tetraedro (v) y la suma de las áreas de cada de sus caras (s_i):

$$r_{in} = \frac{3v}{\sum_{i=0}^3 s_i}. \quad (2.3)$$

Esta medida está comprendida entre 0 y 1, obteniendo este último valor si el tetraedro es regular y el primero si es un tetraedro degenerado (por ejemplo, con los cuatro nodos coplanares).

Las figuras 2.30 y 2.31 muestran las distribuciones de calidad de las mallas para los cinco algoritmos y para dos regiones diferentes: una que describe un HEMT y otra que describe un MOSFET con puerta de polisilicio. La figura 2.32 muestra un ejemplo de la malla final para ambas regiones.

La tabla 2.2 presenta diferentes parámetros que describen estas distribuciones: valor medio del *aspect ratio* ($\langle \gamma \rangle$), desviación estándar (σ), coeficiente de asimetría (CA) y valores máximo y mínimo (γ_{max} y γ_{min} respectivamente). Se puede ver el aumento de calidad con la inclusión de más tetraedros por octante, consecuencia de la mayor proximidad de los tetraedros generados al tetraedro regular, que es el usado como referencia en esta medida. También podemos ver un incremento en la desviación estándar, posiblemente causada por la mayor cantidad de tetraedros diferentes que surgen en las transiciones entre diferentes niveles del *octree* cuando utilizamos algoritmos que dividen cada cara en cuatro triángulos. Podemos ver que el algoritmo A produce algunos tetraedros de elevada calidad. Sin embargo, su pequeño número no es suficiente para elevar el valor medio por encima del obtenido con el algoritmo E.

| Algoritmo/Dispositivo | $\langle \gamma \rangle$ | σ | CA | γ_{max} | γ_{min} |
|-----------------------|--------------------------|----------|-------|----------------|----------------|
| A/HEMT | 0.565 | 0.0901 | -42.3 | 0.963 | 0.249 |
| A/MOSFET | 0.520 | 0.132 | 214 | 0.940 | 0.137 |
| B/HEMT | 0.543 | 0.0798 | 156 | 0.963 | 0.248 |
| B/MOSFET | 0.515 | 0.114 | -2121 | 0.940 | 0.137 |
| C/HEMT | 0.577 | 0.0888 | -182 | 0.963 | 0.249 |
| C/MOSFET | 0.547 | 0.117 | -2407 | 0.924 | 0.184 |
| D/HEMT | 0.599 | 0.0950 | 299 | 0.963 | 0.281 |
| D/MOSFET | 0.561 | 0.128 | 72699 | 0.924 | 0.195 |
| E/HEMT | 0.609 | 0.103 | -4909 | 0.752 | 0.259 |
| E/MOSFET | 0.566 | 0.141 | 9266 | 0.752 | 0.156 |

Tabla 2.2: Parámetros que describen la distribución de calidades de los tetraedros para las mallas estudiadas.

Otra cuestión relevante en la utilidad de una malla para simulaciones de elementos finitos es el número de conexiones entre sus nodos (es decir, las

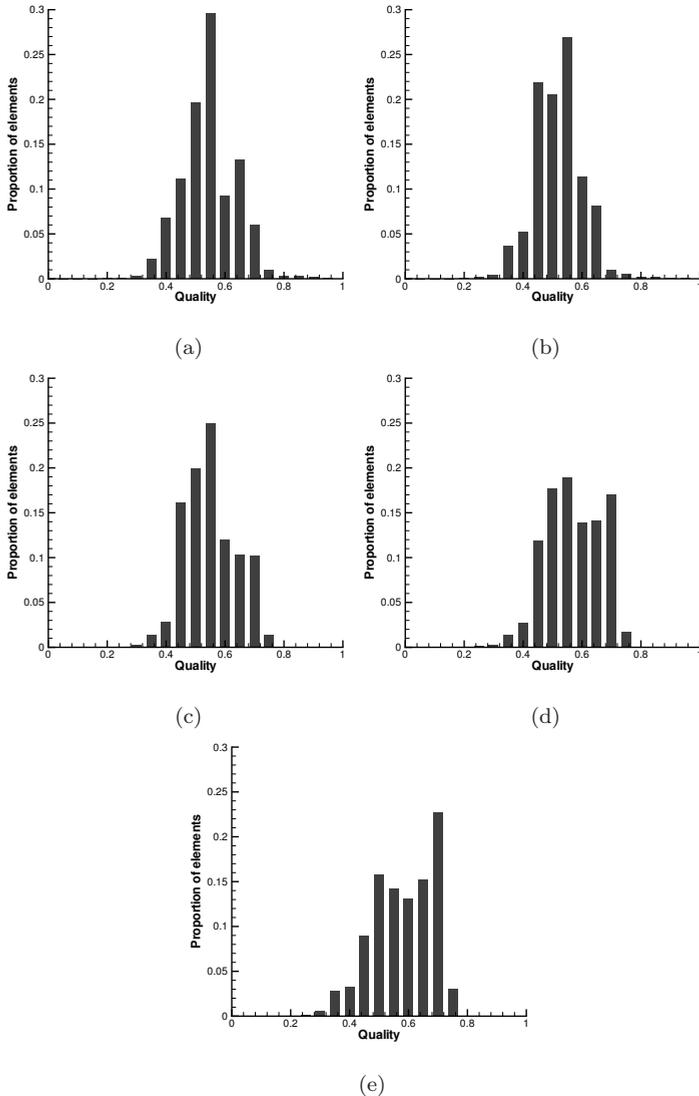


Figura 2.30: Distribución de calidades de las mallas del HEMT usando (a) el algoritmo A, (b) el algoritmo B, (c) el algoritmo C, (d) el algoritmo D and (e) el algoritmo E.

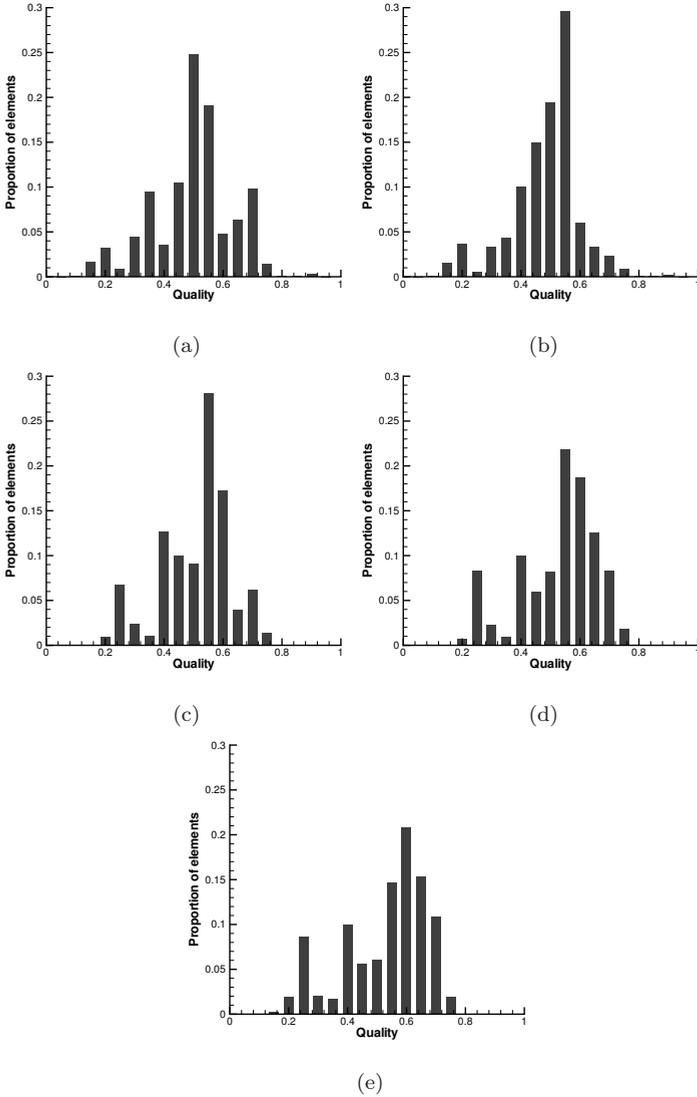


Figura 2.31: Distribución de calidades de las mallas del MOSFET usando (a) el algoritmo A, (b) el algoritmo B, (c) el algoritmo C, (d) el algoritmo D y (e) el algoritmo E.

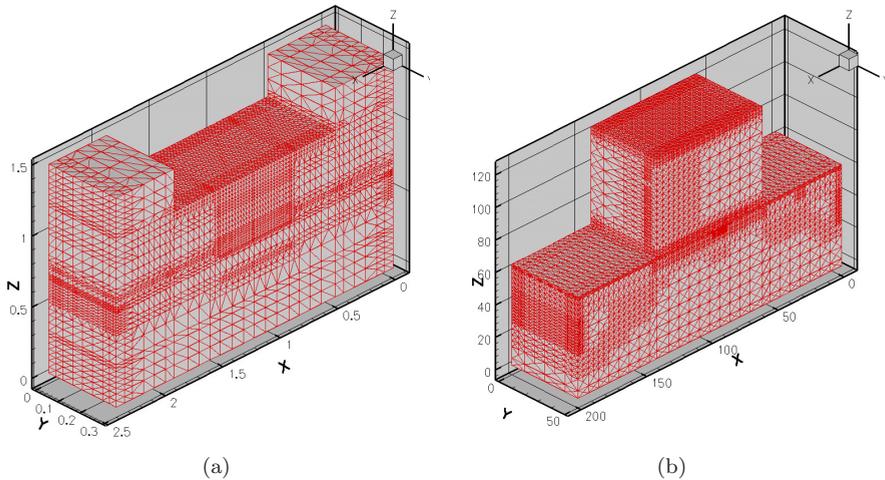


Figura 2.32: Malla final de (a) un HEMT usando el algoritmo A y (b) un MOSFET usando el algoritmo E.

aristas de la malla), ya que es un factor importante para la estabilidad del método [HVKR03]. La figura 2.33 muestra la distribución de conectividades para los nodos de la malla del HEMT. En el eje x se muestra el número de vecinos y en el eje y la proporción de nodos con estos vecinos. Se puede ver que las cinco distribuciones tienen picos en un número de conexiones bajo. Además, excepto en el algoritmo A, también aparece un pico asociado a un alto número de conectividades, lo que puede ser dañino para la convergencia del simulador. Este segundo pico está más desplazado a la derecha para algoritmos con un mayor número de nodos, lo que posiblemente les confiera peores propiedades para las simulaciones de elementos finitos subsiguientes.

Por último, otro aspecto importante es la diferencia notable en número de nodos y elementos, lo que puede hacer que los requisitos de memoria y tiempo del simulador sean los factores decisivos a la hora de elegir un algoritmo u otro.

2.4. Generación de mallas atómicas

Avances en las tecnologías de fabricación han permitido el escalado de los dispositivos semiconductores a dimensiones en las que el número de átomos es relativamente pequeño, haciendo que la posición de cada uno afecte a las

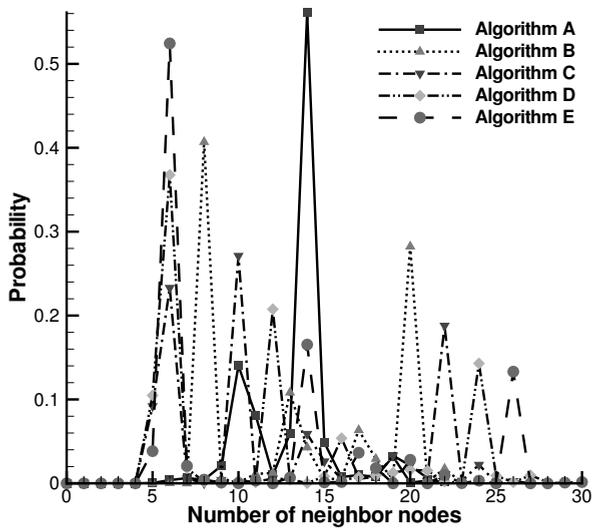


Figura 2.33: Distribución de las conexiones en una malla ejemplo para los diferentes algoritmos.

características de los dispositivos.

Las predicciones en la última edición del ITRS [roa] apuntan hacia la fabricación en masa de transistores con longitudes de puerta física de hasta 8.1 nm. En estas dimensiones, sólo habría espacio para quince celdas unidad de silicio bajo la puerta en la dirección del transporte. Por lo tanto, se hace necesaria la generación de mallas de calidad que permitan resolver de forma precisa efectos a escala atómica en la simulación de transistores MOS de dimensiones nanométricas [CWAA⁺06].

Además, en estas dimensiones, el hecho de que los átomos tomen posiciones discretas en los materiales afecta de forma importante las características de los dispositivos. Por ejemplo, el efecto de las diferentes posiciones de los dopantes en dispositivos nominalmente iguales introduce variaciones en el comportamiento de dispositivos y circuitos [RBAL⁺06]. Sin embargo, la posición exacta de los dopantes en la red cristalina no se ha tenido en cuenta en estos estudios.

En la figura 2.34 se muestran los átomos en un transistor de doble puerta de 10 nm de longitud de puerta, mostrando que la posición de cada átomo tiene que ser tomada en cuenta para resolver de forma precisa cantidades como el potencial o la densidad de portadores, dependientes de la posición precisa de los átomos dopantes. Aunque la estructura atómica de los átomos en la red cristalina tiene una simetría bien conocida, la construcción de mallas de calidad para materiales amorfos tales como α -SiO₂ (ver figura 2.35) es más complicada [GR88, PHC98]. La dificultad viene principalmente de la interfaz Si/SiO₂, difícil de construir incluso para el caso sencillo en el que se considera una estructura cristalina para el SiO₂ (cuarzo α) como se muestra en la figura 2.36.

En primer lugar haremos una breve introducción a la teoría del funcional de la densidad, método utilizados para calcular las posiciones de los átomos en la estructura en estado fundamental. A continuación presentamos una metodología para generación de mallas con los nodos en las posiciones de los átomos en el dispositivo para estructuras cristalinas. Este método lo extendemos para hacer una primera aproximación a dieléctricos amorfos. Para demostrar el uso de la metodología estudiamos dos ejemplos de sistemas que se fabrican habitualmente en transistores actuales a los que hemos aplicado la metodología: (i) un semiconductor de silicio con cuarzo α como dieléctrico de puerta y (ii) un semiconductor de silicio con HfO₂ monoclinico y capa de interfaz de SiO₂. Finalmente presentamos algunos resultados sobre la calidad de las mallas obtenidas empleando esta metodología.

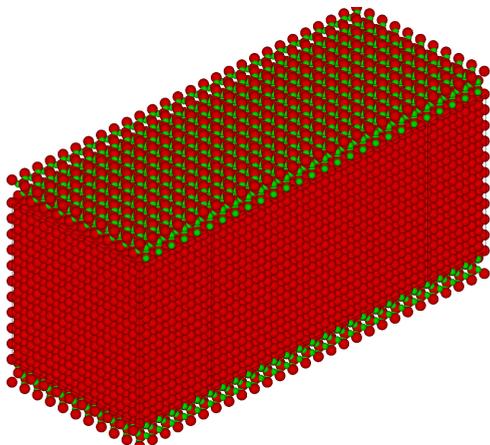


Figura 2.34: Ilustración de la estructura atómica ideal de un transistor DG-MOSFET de silicio de longitud de puerta de 10 nm. Las esferas rojas representan átomos de silicio, mientras que las verdes representan los átomos de oxígeno del óxido de puerta.

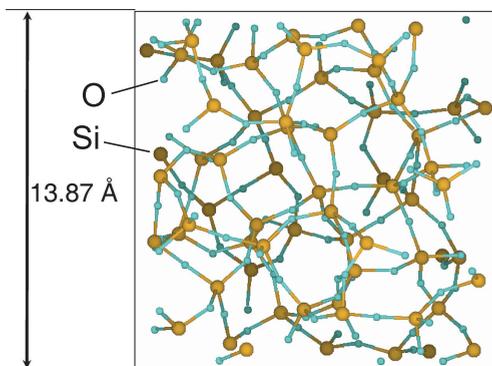


Figura 2.35: Estructura de dióxido de silicio amorfo con 192 átomos ($\rho = 2,39g/cm^{-3}$) obtenida de simulaciones de dinámica molecular clásicas [MSSS04].

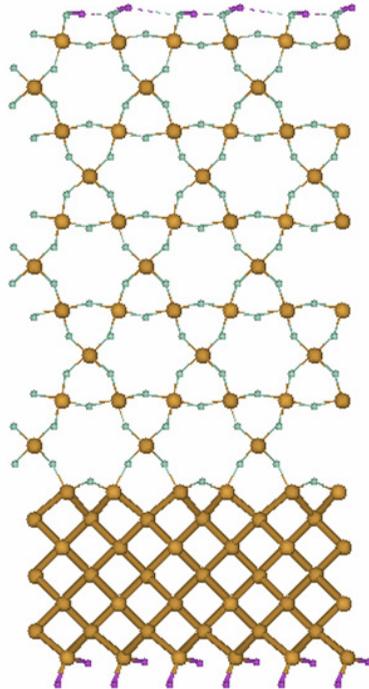


Figura 2.36: Estructura atómica de la interfaz cristalina Si/SiO₂ pasivada con hidrógeno [YKU⁺01].

2.4.1. Teoría del funcional de la densidad

La descripción microscópica de las propiedades de la materia es una tarea de una elevada complejidad. Para un sistema de P núcleos y N electrones la tarea consistiría en la resolución de la ecuación de Schrödinger

$$\widehat{H}_T \Psi_i(\mathbf{r}, \mathbf{R}) = E_i \Psi_i(\mathbf{r}, \mathbf{R}), \quad (2.4)$$

con el siguiente hamiltoniano:

$$\begin{aligned} \widehat{H}_T = & - \sum_{I=1}^P \frac{\hbar^2}{2M_I} \nabla_I^2 - \sum_{i=1}^N \frac{\hbar^2}{2m} \nabla_i^2 + \\ & + \frac{e^2}{2} \sum_{I=1}^P \sum_{J \neq I}^P \frac{Z_I Z_J}{|\mathbf{R}_I - \mathbf{R}_J|} + \frac{e^2}{2} \sum_{i=1}^N \sum_{i \neq j}^N \frac{1}{|\mathbf{r}_i - \mathbf{r}_j|} - \\ & - e^2 \sum_{I=1}^P \sum_{i=1}^N \frac{Z_I}{|\mathbf{R}_I - \mathbf{r}_i|}, \end{aligned} \quad (2.5)$$

donde $\mathbf{R} = \{\mathbf{R}_I\}$, $I = 1, \dots, P$, es el conjunto de las P coordenadas nucleares, $\mathbf{r} = \{\mathbf{r}_i\}$, $i = 1, \dots, N$, es el conjunto de las N coordenadas electrónicas y Z_I y M_I las P cargas y masas nucleares respectivamente.

En la práctica este problema es prácticamente imposible de resolver, y las soluciones se obtienen para algunos casos sencillos utilizando aproximaciones. La primera aproximación que se suele hacer es la aproximación adiabática o de Born-Oppenheimer, que permite factorizar la función de onda en dos términos, uno para los electrones y otro para los núcleos. En este caso podemos escribir el hamiltoniano del sistema electrónico como

$$\widehat{H} = \widehat{T} + \widehat{V}_{ext} + \widehat{W} = \sum_{i=1}^N \frac{(-i\hbar \nabla_i)^2}{2m} + \sum_{i=1}^N v_{ext}(\mathbf{r}_i) + \sum_{\substack{i,j=1 \\ i < j}}^N w(\mathbf{r}_i, \mathbf{r}_j), \quad (2.6)$$

donde \widehat{T} representa el operador energía cinética, \widehat{V}_{ext} la interacción con fuentes externas y \widehat{W} la interacción electrón-electrón.

La teoría del funcional de la densidad (DFT, *density functional theory*) proporciona una alternativa para resolver esta ecuación basándose en el teorema de Hohenberg-Kohn. Los principales aportaciones de este teorema son:

- El potencial externo está unívocamente determinado por la densidad electrónica salvo por una constante aditiva. Por lo tanto, la densidad

electrónica también determina de forma unívoca la función de onda en estado fundamental. Esto es, el estado fundamental es un funcional de la densidad electrónica $\Psi[n]$. Para la densidad electrónica real del sistema n_0 obtenemos la función de onda del sistema en estado fundamental $\Psi_0 = \Psi[n_0]$. Además, este funcional es universal, sólo depende de la interacción w .

- Esto permite entender cualquier observable como funcional de la densidad:

$$O[n] := \langle \Psi[n] | \hat{O} | \Psi[n] \rangle. \quad (2.7)$$

En particular, podemos escribir la energía del estado fundamental como

$$E[n] = \langle \Psi[n] | \hat{H} | \Psi[n] \rangle = F[n] + \int v_{ext}(\mathbf{r})n(\mathbf{r}) d\mathbf{r}, \quad (2.8)$$

donde $F[n]$ es la parte universal del funcional que no incluye el potencial externo.

- Podemos calcular la energía E_0 del estado fundamental a partir de un principio de mínimo como

$$E_0 = E[n_0] = \min_{n \in N} E[n], \quad (2.9)$$

donde N es el conjunto de densidades en un estado fundamental solución de la ecuación de Schrödinger.

Utilizando la teoría del funcional de la densidad es posible calcular la energía y la densidad del estado fundamental de forma exacta si se conoce el funcional $F[n] = \langle \Psi[n] | \hat{T} + \hat{W} | \Psi[n] \rangle$. Sin embargo, el teorema de Kohn-Sham no especifica como calcularlo. El principal problema es encontrar una expresión para la energía cinética. En 1965, Kohn y Sham [KS65] propusieron reemplazar la energía cinética del sistema “interactuante” por la de un sistema equivalente sin interacción, \hat{T}_R , ya que esta puede ser calculada fácilmente. De esta forma, suponiendo una interacción entre partículas de tipo coulombiano, podemos escribir el funcional energía en la siguiente forma (conocida como funcional de Kohn-Sham):

$$E_{KS}[n] = T_R[n] + \int n(\mathbf{r})v_{ext}(\mathbf{r}) d\mathbf{r} + \frac{1}{2} \iint \frac{n(\mathbf{r})n(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r}d\mathbf{r}' + E_{XC}[n], \quad (2.10)$$

donde $E_{XC}[n]$ es la energía de intercambio-correlación como funcional de la densidad y que incluye todos los efectos de interacción “de muchos cuerpos” no contenidos en los otros términos. Este sistema sin interacción se define de tal forma que la densidad en estado fundamental sea igual a la del sistema real.

El sistema no “interactuante” verifica la ecuación de Schrödinger para un potencial efectivo v_{eff} :

$$\left\{ -\frac{\nabla^2}{2} + v_{eff}(\mathbf{r}) \right\} \phi_{i,s}(\mathbf{r}) = \varepsilon_i \phi_{i,s}(\mathbf{r}). \quad (2.11)$$

Se puede demostrar que el potencial v_{eff} es

$$v_{eff} = v_{ext} + \int \frac{n(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r}' + \mu_{XC}[n](\mathbf{r}), \quad (2.12)$$

donde $\mu_{XC} = \delta E_{XC}[n]/\delta n$ es la derivada funcional de la energía de intercambio-correlación. A partir de las funciones ϕ_i se puede calcular la densidad del sistema como

$$n(\mathbf{r}) = \sum_s \sum_i^{N_s} |\phi_{i,s}|^2 \quad (2.13)$$

y la energía total como

$$E_{KS}[n] = \sum_s \sum_i^{N_s} \varepsilon_{i,s} - \frac{1}{2} \iint \frac{n(\mathbf{r})n(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r}d\mathbf{r}' + E_{XC}[n] - \int n(\mathbf{r})\mu_{XC}[n](\mathbf{r}) d\mathbf{r}, \quad (2.14)$$

donde se ha escogido el número de ocupación 1 para $i \leq N_s$ ($s = 1, 2$) y 0 para $i > N_s$.

Por último, para realizar los cálculos hace falta una expresión para el funcional de la energía de intercambio-correlación $E_{XC}[n]$. Debido a la alta complejidad de las expresiones exactas, es habitual el uso de aproximaciones. La más utilizada durante mucho tiempo fue la llamada aproximación de densidad local (*local density approximation*, LDA) [KS65], que considera el sistema inhomogéneo como localmente homogéneo y usa las expresiones obtenidas en este caso. Este método produce buenos resultados en muchos sistemas de interés, especialmente en los que la densidad electrónica es bastante uniforme. Sin embargo, hay algunas situaciones en las que esta aproximación no reproduce bien algunas características: en sistemas atómicos, donde hay grandes variaciones

de la densidad, en moléculas débilmente ligadas, en superficies metálicas, etc. Hay muchas direcciones diferentes en las que se investigan extensiones y mejoras a esta aproximación. Probablemente debido a su eficiencia computacional, las más populares se basan en introducir semilocalmente las inhomogeneidades de la densidad expandiendo $E_{XC}[n]$ como serie en términos del gradiente de la densidad. Estas aproximaciones, conocidas como aproximaciones del gradiente generalizado (*generalized gradient approximations*, GGA) [PBE96, LP94], han tenido bastante éxito mejorando algunas de las características de la aproximación LDA.

2.4.2. Regiones cristalinas

En esta sección describimos la estrategia de mallado para considerar las posiciones atómicas exactas en el dispositivo semiconductor. Recientemente, mallas de Delaunay [DW06] con nodos en la posición de los átomos se han utilizado en la simulación de transistores tipo nanohilo [LSF06, LSF07], pero sólo se consideraron los átomos de la red de Si, ignorando la posición de los átomos de la capa de óxido y las regiones de transición. Sin embargo, algoritmos que producen mallas de Delaunay pueden tener problemas cuando se aplican a regiones casi cristalinas resultantes de la inclusión de un óxido cristalino y de la interfaz. Esto se debe principalmente a la generación de tetraedros tipo *sliver*, que pueden provocar problemas de convergencia en los resolutores de elementos finitos posteriores. Esto se ve agravado si consideramos la generación de mallas CDT para mantener determinados planos del interior de la estructura que pueden ser útiles para definir propiedades de interfaz o para poner condiciones de contorno en las simulaciones subsiguientes.

Hemos escogido un método basado en un algoritmo tipo *octree* [APGL06, HK05] con celdas unidad de materiales como octantes terminales y la posibilidad de usar diferentes esquemas de descomposición en la misma malla. Como primer paso, debemos definir esquemas de descomposición para las celdas unidad extendidas de cada material. Estas celdas unidad extendidas, que denominaremos simplemente celdas unidad de ahora en adelante, incluyen átomos ficticios extra para completar un volumen que cubra el espacio por completo al ser replicado, aunque serán eliminados posteriormente en esta fase de replicación. La descomposición de las celdas unidad en tetraedros se puede llevar a cabo de diferentes formas, siendo el único requerimiento la simetría en la triangulación de caras opuestas para obtener una malla conforme en el proceso de replicación. En las figuras 2.37 y 2.38 mostramos, como ejemplo,

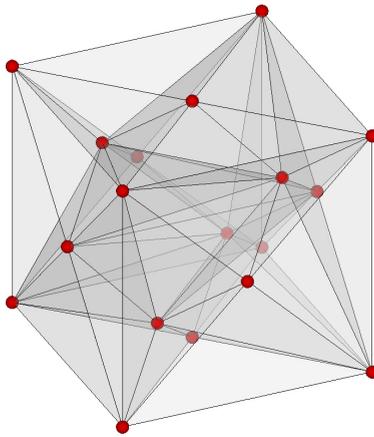


Figura 2.37: Malla del octante terminal usando los átomos en la celda unidad del silicio como nodos.

la descomposición de las celdas unidad del silicio (Si) y dióxido de silicio cristalino (α -SiO₂). En el caso del Si hemos utilizado la simetría interna de la celda unidad para llevar a cabo la descomposición, descomponiendo en tetraedros un prisma triangular y replicándolo con las transformaciones de simetría oportunas. En la figura 2.39 podemos ver la malla de una lámina de silicio de $31 \times 19 \times 12$ celdas unidad construida usando la descomposición en tetraedros descrita.

2.4.3. Regiones de interfaz

Una segunda cuestión a tratar debe ser la descomposición de la estructura periódica en la interfaz entre materiales, por ejemplo, la interfaz Si/SiO₂. Tal descomposición no se puede hacer simplemente usando las estructuras de los materiales correspondientes debido a las deformaciones de su estructura atómica local en la interfaz. Hemos adoptado la topología de esta interfaz en concreto de [YKU⁺01], mientras que la estructura atómica local de la interfaz ha sido obtenida usando la teoría del funcional de la densidad tal como se describe en [GSPS07]. Un ejemplo de descomposición de una celda de interfaz se muestra en la figura 2.40 para el caso de la estructura Si/SiO₂.

Una vez que disponemos de las unidades básicas para construir la malla, podemos proceder a la definición de la estructura que queremos mallar. Esto

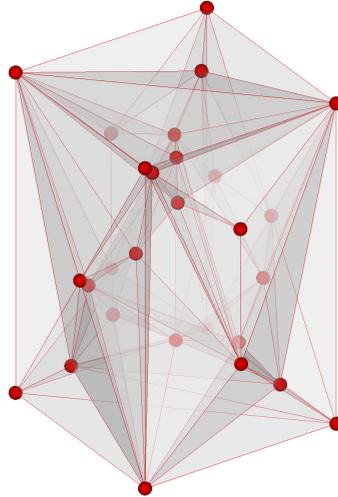


Figura 2.38: Patrón utilizado para construir la malla de SiO₂ cristalino usando nodos en las posiciones atómicas.

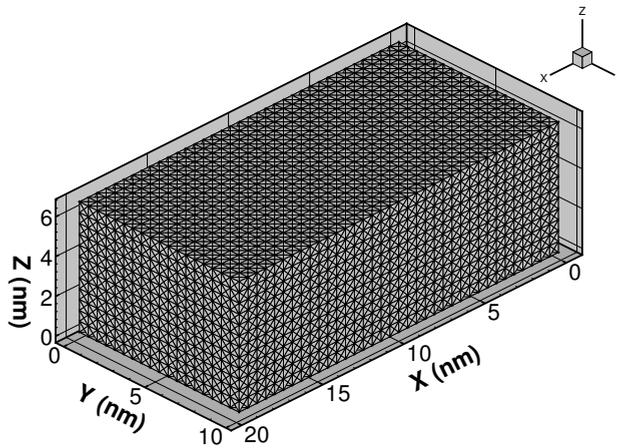


Figura 2.39: Malla de una lámina de silicio de 6.5 nm de grosor.

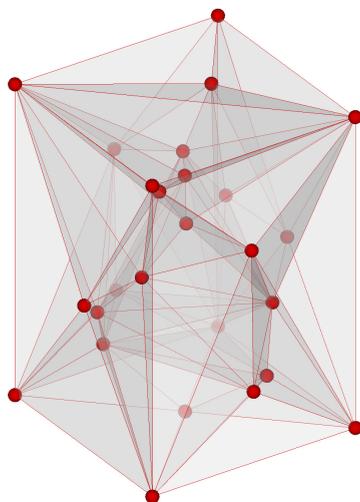


Figura 2.40: Patrón utilizado para construir la malla de la interfaz entre silicio y dióxido de silicio cristalinos.

incluye la especificación de materiales y espesor aproximado de las diferentes capas. Esta información se usa para conocer el número de celdas unidad necesarias en cada dirección. En el caso de que el espesor de una capa sea demasiado pequeño para una única celda unidad se utilizará únicamente el espesor asignado en la celda de interfaz. Su introducción se hace en un paso posterior, cuando las mallas de cada material están ya creadas. La figura 2.41 muestra un ejemplo de la malla construida aplicando esta técnica a una estructura $\text{SiO}_2/\text{Si}/\text{SiO}_2$. La estructura atómica relajada de la interfaz Si/SiO_2 se obtuvo con cálculos de primeros principios [YKU⁺01] usando la DFT.

La figura 2.42 muestra otro ejemplo de la metodología aplicada a una estructura $\text{Si}/\text{SiO}_2/\text{HfO}_2$. Nuevamente, la estructura de la interfaz $\text{SiO}_2/\text{HfO}_2$ se extrajo de la distribución atómica obtenida usando cálculos de primeros principios [GS07] mostrada en la figura 2.43. En este caso, el crecimiento de HfO_2 monoclinico sobre la estructura Si/SiO_2 no produjo una interfaz cristalina. Sin embargo, sí tenía algunos patrones que extrajimos para obtener una celda que representara la interfaz. Con esta modificación, la malla obtenida no replica de forma exacta la posiciones cristalinas iniciales. Sin embargo, en la siguiente sección presentamos una técnica para modificar estas malla cristalinas haciéndolas más cercanas a las estructuras reales.

Una dificultad final que debe ser tratada al construir mallas de resolución

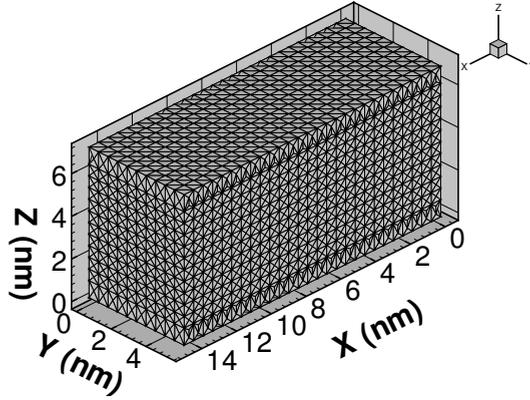


Figura 2.41: Malla de un MOSFET de doble puerta con SiO_2 cristalino como dieléctrico de puerta.

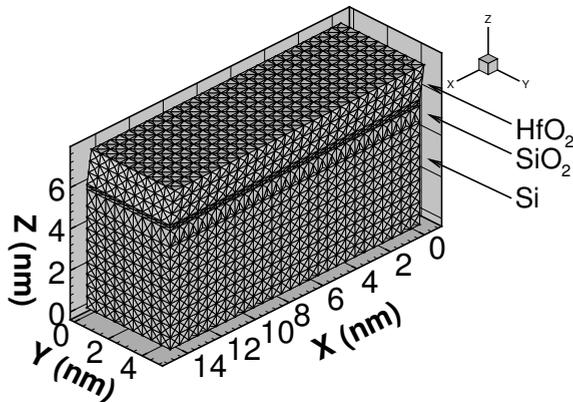


Figura 2.42: Malla de un MOSFET con un dieléctrico de puerta compuesto por una capa de SiO_2 cristalino y otra de HfO_2 (monoclínico).

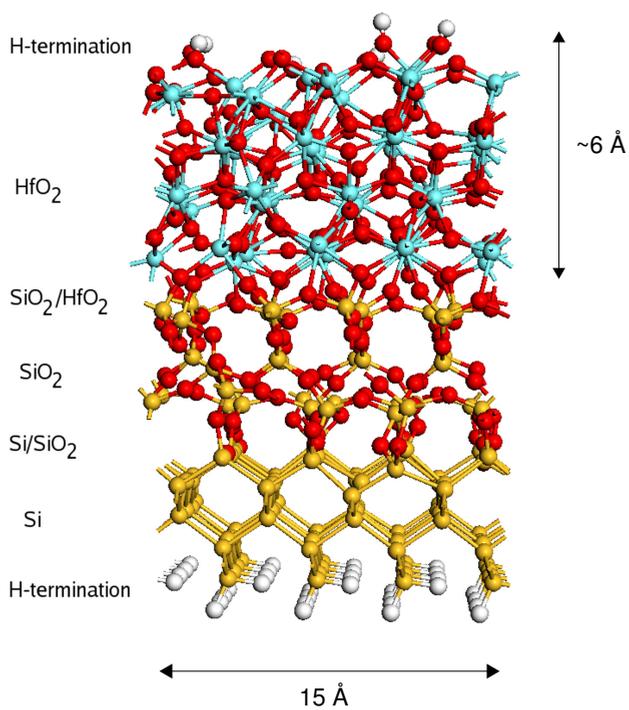


Figura 2.43: Estructura atómica calculada de una interfaz Si/SiO₂/HfO₂.

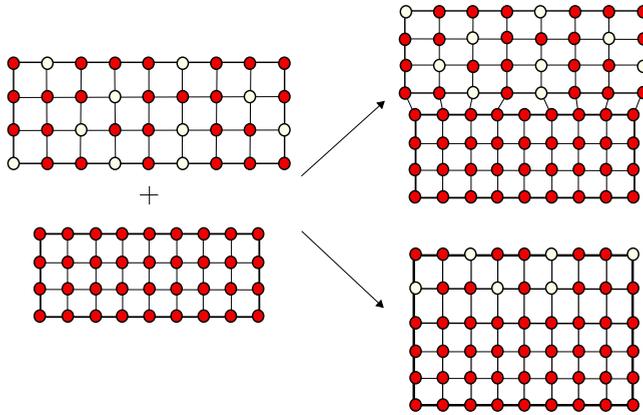


Figura 2.44: Posibles crecimientos de dos materiales cristalinos con diferente constante de red. Dependiendo del grosor de la capa, la estructura resultante puede estar tensionada (abajo) o dislocada (arriba).

atómica es la diferencia entre las constantes de red de los materiales que forman la estructura de capas. Cuando un material cristalino se crece sobre otro material cristalino con diferente constante de red, pueden darse dos situaciones [Yua99], representadas en la figura 2.44:

- Si el grosor del material que se crece está por debajo del grosor crítico, entonces su constante de red se modifica, siempre que la diferencia sea pequeña, para ajustarse a la del otro material.
- Si el grosor del material que se crece está por encima del grosor crítico, entonces la constante de red no se modifica y aparecen dislocaciones.

Como primera aproximación al problema suponemos que la capa de óxido cristalino es lo suficientemente delgada como para que su constante de red se adapte a la del silicio, de tal forma que creamos el cristal de las capas de óxido con las dimensiones del silicio en el plano paralelo a la interfaz. Aunque esta es la situación más habitual en estructuras de dispositivos semiconductores, también sería posible introducir una celda generalizada en la interfaz que nos permita estudiar materiales con diferencias mayores entre sus constantes de red.

2.4.4. Regiones amorfas

En la sección anterior mostramos como generar mallas para dispositivos compuestos por materiales cristalinos. Sin embargo, en los dispositivos reales, el dieléctrico suele ser amorfo. Esto requiere una extensión del generador de mallas para permitir el manejo de estructuras amorfas.

La descripción de la estructura de materiales amorfos y sus interfaces es una materia muy compleja que todavía no se comprende completamente, incluso para un material tan ampliamente utilizado como el SiO_2 [FCBA06], cuya interfaz con el Si ha sido la clave del éxito de la tecnología CMOS. La generación de estructuras realistas para las interfaces requiere el uso de técnicas computacionales tales como DFT o dinámica molecular [FCBA06, PHC96, BPP00, BP03].

Como primera aproximación al problema, realizamos transformaciones locales a la malla para eliminar la cristalinidad en la localización de los nodos del óxido e imitar una distribución de nodos amorfa. Esto puede ser utilizado para estudios cualitativos del impacto sobre el comportamiento de los dispositivos provocado por variaciones aleatorias de las propiedades materiales en las capas amorfas. Sin embargo, esta transformación resulta en distribuciones de átomos cuyas propiedades físicas tales como la distribución de ángulos de enlace no coinciden con las medidas experimentales.

La principal idea del algoritmo es la formación de una capa de material amorfo a partir de la cristalina obtenida con la metodología previamente descrita. También nos gustaría mantener el mismo grafo de adyacencias para la malla para poder utilizar la misma estructura de datos definida en el simulador con diferentes configuraciones, lo que aceleraría la simulación de grupos de dispositivos microscópicamente diferentes.

El algoritmo implementado realiza una redistribución aleatoria en las posiciones de los nodos con un grado ajustable. El grado de desplazamiento permitido está dado por un parámetro $\beta \in (0, 1]$. Un valor próximo a cero significa que el movimiento será muy pequeño, mientras que un valor de uno significa que el movimiento se puede realizar en todo el volumen permitido por criterios geométricos. Para evitar intersecciones entre tetraedros cuando se mueven los nodos definimos el volumen permitido en dos pasos. En el primer paso hacemos una lista de los tetraedros vecinos, que a su vez definen un volumen que rodea por completo al nodo. Si este volumen es convexo, entonces es el volumen donde el nodo se puede mover libremente sin necesidad de redefinir conectividades. Si no es convexo, entonces debemos restringir el volumen

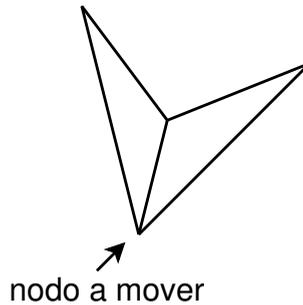


Figura 2.45: Dos triángulos delimitando una superficie no convexa.

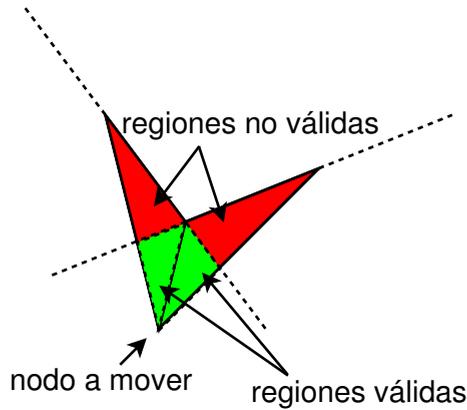


Figura 2.46: Regiones de la superficie que pueden y no pueden ser utilizadas para mover el nodo.

para evitar intersecciones entre tetraedros tras el movimiento del nodo. Las figuras 2.45 y 2.46 muestran un ejemplo de esta situación en dos dimensiones.

En tres dimensiones la situación es equivalente, solo que tenemos que restringir el movimiento usando planos en lugar de rectas. También tenemos que tener en cuenta que un nodo puede pertenecer a un número arbitrario de tetraedros, definiendo así un poliedro arbitrario. Por tanto, el volumen válido para definir el movimiento del nodo será difícil de describir en términos de los tetraedros que contienen al nodo. Por esto hemos decidido utilizar una técnica de rechazo para generar la posición final del nodo. En primer lugar generamos un punto arbitrario en el volumen completo definido por los tetraedros vecinos utilizando una distribución uniforme. Si el vector definido por las posiciones

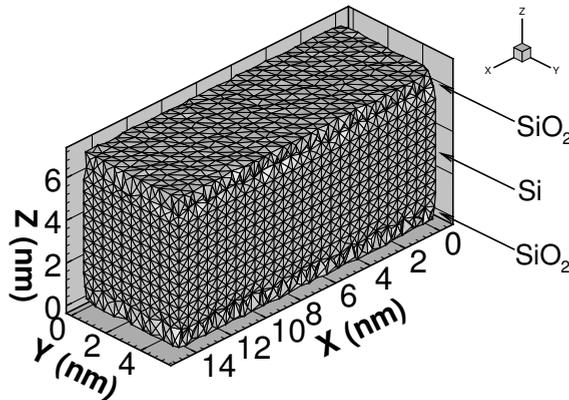


Figura 2.47: Discretización de un transistor con regiones amorfas y cristalinas obtenida de la malla de materiales cristalinos.

nueva y antigua del nodo no intersecta ninguno de los planos definidos por todas las caras opuestas a ese nodo, entonces conservamos el cambio. En caso contrario generamos una nueva posición final candidata. De esta forma obtenemos un movimiento aleatorio con distribución uniforme en el volumen válido. Ejecutando este movimiento aleatorio de los nodos varias veces, podemos obtener diferentes capas amorfas a partir de la misma estructura cristalina. Las figuras 2.47 y 2.48 muestran, respectivamente, una malla generada con este método y las posiciones de los átomos correspondientes.

2.4.5. Calidad de las mallas

Como ya dijimos antes, las características de los sistemas de ecuaciones que describen las ecuaciones diferenciales en derivadas parciales discretizadas que deben ser resueltos, dependen enormemente en las propiedades de las mallas. Al igual que antes, hemos decidido utilizar un criterio de calidad que satisfaga la definición propuesta en [DLGC98], el *aspect ratio* γ definido en (2.2). La tabla 2.3 presenta algunos de los parámetros que caracterizan la distribución de γ para las distintas celdas unidad, incluyendo el *aspect ratio* medio ($\langle \gamma \rangle$), la desviación estándar (σ), el coeficiente de asimetría (CA) y los valores máximo y mínimo del *aspect ratio* (γ_{max} y γ_{min} respectivamente).

Para evitar elementos de calidad muy pobre, el movimiento de los nodos en la generación de mallas de regiones amorfas incluye una segunda ligadura relacionada con la calidad. La nueva posición de un nodo no puede producir

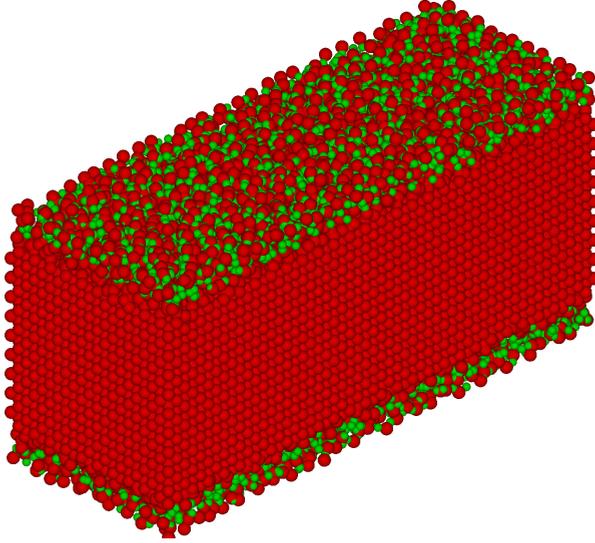


Figura 2.48: Átomos en la estructura después del proceso de hacer el óxido amorfo. Las esferas rojas representan átomos de silicio y las verdes de oxígeno. Notar las posiciones irregulares de los átomos en la capa de óxido al compararlos con sus posiciones regulares en la figura 2.34.

Tabla 2.3: Parámetros para describir la distribución de calidades de los tetraedros para las mallas de diferentes materiales. Los parámetros dados para las mallas amorfas son $\beta_1 = 0,05$ y $\beta_2 = 1,0$.

| Material | $\langle \gamma \rangle$ | σ | CA | γ_{max} | γ_{min} |
|--|--------------------------|----------|--------|----------------|----------------|
| Si | 0.51 | 0.08 | 1.031 | 0.662 | 0.458 |
| SiO ₂ | 0.31 | 0.09 | 0.399 | 0.553 | 0.151 |
| HfO ₂ | 0.46 | 0.12 | -0.089 | 0.777 | 0.200 |
| Si-SiO ₂ | 0.39 | 0.12 | 0.789 | 0.739 | 0.183 |
| SiO ₂ -HfO ₂ | 0.40 | 0.10 | -0.838 | 0.575 | 0.127 |
| α -SiO ₂ (β_1) | 0.31 | 0.09 | 0.372 | 0.595 | 0.132 |
| α -SiO ₂ (β_2) | 0.29 | 0.12 | 0.789 | 0.845 | 0.102 |

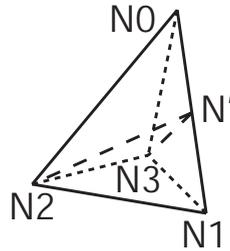


Figura 2.49: Refinamiento de un tetraedro usando bisección.

ningún tetraedro con *aspect ratio* inferior a un umbral dado. Aunque valores altos de este umbral serían ideales desde el punto de vista de la calidad, es necesaria una solución de compromiso, ya que estos valores pueden hacer la generación muy lenta o incluso imposible.

2.5. Refinamiento adaptativo de mallas tetraédricas

Las mallas no estructuradas descritas hasta ahora pueden no tener suficiente resolución en regiones con variaciones muy pronunciadas de alguna cantidad física. Nuestro objetivo es refinar la malla en regiones donde estas variables tengan cambios abruptos que, o bien no se conocían a priori, o bien no fueron más refinadas, por ejemplo en la generación del *octree*, para evitar un número de nodos excesivo. Sin embargo, una vez resuelto el problema es posible extraer información sobre el error en la solución que nos permita mejorar la malla en algunas regiones si la convergencia en ellas es pobre. Para simplificar el proceso, la estimación del error a partir de la solución debería ser local y producir una mejora en la convergencia cuando la malla se refine de acuerdo a su valor. Esta etapa de estimación del error local asigna un valor a cada tetraedro, lo que sirve como base para decidir si el elemento debería ser dividido en elementos más pequeños. Para esto hemos elegido un algoritmo de bisección [Riv84, AMP00, WCH⁺06], que incluye nuevos nodos en la malla en aristas seleccionadas (una por tetraedro). La figura 2.49 muestra un ejemplo de este tipo de refinamiento en el que el tetraedro $T_0 = \{N0, N1, N2, N3\}$ se divide en dos nuevos tetraedros, $T_1 = \{N0, N', N2, N3\}$ y $T_2 = \{N', N1, N2, N3\}$. En nuestra implementación del algoritmo, dividimos todos los tetraedros que comparten una arista en lugar de mantener la conformalidad en un paso posterior. La selección del eje que se dividirá en dos puede ser realizada utilizando

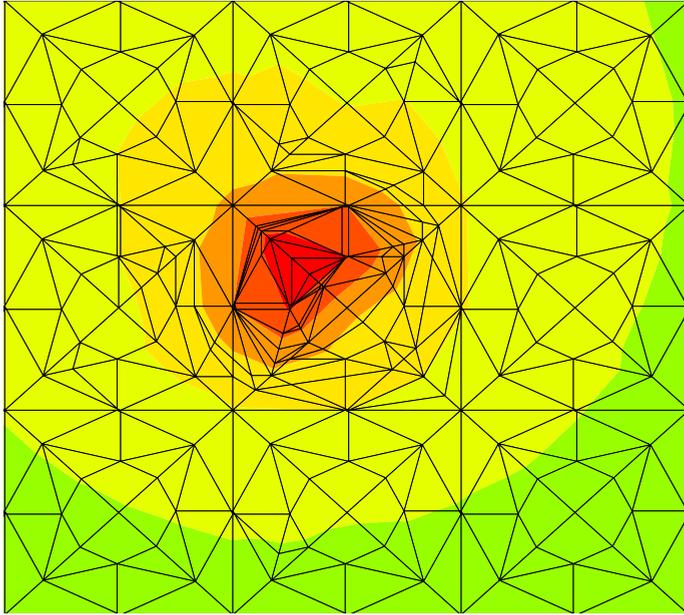


Figura 2.50: Detalle del refinamiento cerca de un átomo de impureza. Los colores se corresponden con valores del potencial.

diferentes criterios. Escogemos un umbral para una cantidad dada (física o construcción matemática como el residuo) y los tetraedros que superen este umbral se seleccionan para ser refinados. Se lleva a cabo una búsqueda en el tetraedro dado y se decide entonces cuál de las aristas tiene que ser dividida. Esta elección está basada en valores de variables sobre nodos. Además, también se visitan los tetraedros vecinos para dar preferencia a la aristas con mayor variación de esta variable de refinamiento. La figura 2.50 muestra un ejemplo de una malla refinada cerca de una impureza ionizada usando como guía el potencial. Cabe resaltar que el refinamiento de la mallas hace necesario el reordenamiento de las matrices del problema. Después de la introducción de los nuevos nodos en la malla, su ordenamiento deja de ser apropiado para las propiedades numéricas de las matrices del problema, lo que implicaría tiempos de resolución demasiado elevados.

En la figura 2.51 mostramos el efecto del refinado sobre la convergencia de las ecuaciones que debemos resolver. La figura muestra la evolución de la norma-2 del residuo de la ecuación de continuidad de electrones, $\|r\|$, con las iteraciones del resolutor para la malla original y una malla refinada. El criterio

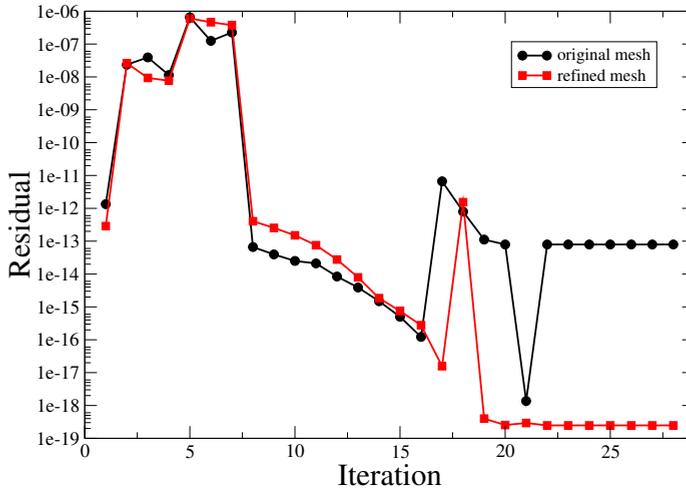


Figura 2.51: Evolución de la norma del residuo de la ecuación de continuidad con el número de iteración para la malla original y una refinada.

para refinar la malla fue la variación del potencial electrostático. En este caso podemos ver que el valor en el que se estabiliza es menor para la malla refinada.

Capítulo 3

El modelo de arrastre-difusión

Como ya se comentó en el Capítulo 1, existen diferentes modelos a la hora de simular el comportamiento de los dispositivos semiconductores, cada uno de ellos con su propio nivel de complejidad. El modelo de arrastre-difusión, pese a su validez limitada, es todavía uno de los modelos más extendidos. Esto se debe al compromiso entre coste computacional y capacidad predictiva que proporciona una vez que se introducen modelos para la movilidad. Simuladores comerciales de dispositivos semiconductores que incluyen el modelo de arrastre-difusión son, por ejemplo, el *Sentaurus Device* [sen] o el *Taurus-Medici* [med] de Synopsys, el ATLAS de Silvaco [atl] o el APSYS de Crosslight [aps].

En la primera parte de este capítulo describimos las ecuaciones principales usadas en el modelo de arrastre-difusión. En primer lugar, a partir de las ecuaciones de Maxwell obtendremos las ecuaciones de Poisson y de continuidad de corriente. Después presentaremos la aproximación utilizada en el modelo de arrastre-difusión para las densidades de corriente de portadores. A continuación describimos algunos de los modelos utilizados para aproximar los efectos cuánticos en las simulaciones, con especial énfasis en los modelos que emplearemos más adelante. Una vez presentados los modelos utilizados en las simulaciones, describimos su discretización usando el método de los elementos finitos sobre mallas tetraédricas. En este proceso el problema se transforma en el de la resolución de sistemas de ecuaciones dispersas no lineales, por lo que describiremos brevemente los métodos de resolución empleados. Finalmente,

presentamos en este capítulo los resultados obtenidos utilizando el modelo de arrastre-difusión para el estudio del impacto de algunas fuentes de fluctuaciones sobre el comportamiento de transistores MOSFET.

3.1. El modelo de arrastre-difusión

En el modelo de arrastre-difusión las ecuaciones básicas a resolver son las ecuaciones de Poisson y de continuidad de huecos y electrones. Estas ecuaciones que describen el comportamiento eléctrico del semiconductor se determinan a partir de las ecuaciones de Maxwell, que definen la evolución del campo electromagnético en un medio arbitrario [Sel84]:

$$\nabla \times \mathbf{H} = \mathbf{J} + \frac{\partial \mathbf{D}}{\partial t}, \quad (3.1)$$

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t}, \quad (3.2)$$

$$\nabla \cdot \mathbf{D} = \rho, \quad (3.3)$$

$$\nabla \cdot \mathbf{B} = 0, \quad (3.4)$$

donde \mathbf{E} y \mathbf{D} son los vectores campo y desplazamiento eléctrico, \mathbf{H} y \mathbf{B} son el campo y la inducción magnética, \mathbf{J} es el vector densidad de corriente libre, y ρ es la densidad de carga libre (no debida a la polarización). Es preciso además considerar una variable espacial en \mathbb{R}^3 $\mathbf{r} = (x_0, x_1, x_2)$ y una temporal t donde $t \in \mathbb{R}^+$. Junto con estas ecuaciones, es necesario especificar una relación entre los vectores campo y desplazamiento eléctricos. Vamos a considerar que nuestro medio es lineal, de tal forma que la relación entre \mathbf{D} y \mathbf{E} es:

$$\mathbf{D} = \varepsilon \mathbf{E}, \quad (3.5)$$

donde ε es la permitividad del medio. Además consideramos que el medio es isótropo, por lo que ε será un escalar.

3.1.1. Ecuación de Poisson

Para la deducción de la ecuación de Poisson partimos de las ecuaciones de Maxwell. Como la inducción magnética tiene divergencia cero, puede representarse como el rotacional de un potencial vector \mathbf{A} . Por lo tanto, la ecuación (3.4) puede satisfacerse si se expresa la inducción magnética como:

$$\mathbf{B} = \nabla \times \mathbf{A}, \quad (3.6)$$

donde para un \mathbf{B} dado, el vector \mathbf{A} no es único. Si sustituimos en la ecuación (3.2) el valor de \mathbf{B} por el correspondiente a la ecuación (3.6) se obtiene la siguiente expresión:

$$\nabla \times \left(\mathbf{E} + \frac{\partial \mathbf{A}}{\partial t} \right) = 0. \quad (3.7)$$

Si el rotacional de un vector es cero, se puede expresar dicho vector como el gradiente de un campo escalar. Así, se obtiene:

$$\mathbf{E} + \frac{\partial \mathbf{A}}{\partial t} = -\nabla \psi. \quad (3.8)$$

Si suponemos que las velocidades de los portadores dentro del dispositivo son suficientemente pequeñas comparadas con la velocidad de propagación de los campos (c), podemos considerar que $\left| \frac{\partial \mathbf{A}}{\partial t} \right| \ll |\nabla \psi|$. Usando esta aproximación podemos escribir:

$$\mathbf{E} = -\nabla \psi. \quad (3.9)$$

Combinando esta ecuación (3.9) con (3.3) y (3.5) obtenemos la ecuación de Poisson que define el comportamiento del potencial eléctrico:

$$\nabla(\varepsilon \nabla \psi) = -\rho. \quad (3.10)$$

En el interior de un semiconductor la densidad espacial de carga viene dada por:

$$\rho = q(p - n + C), \quad (3.11)$$

donde q es la unidad de carga elemental, p la concentración de huecos, n la concentración de electrones y C el perfil de impurezas eléctricamente activas:

$$C = N_D^+ - N_A^-, \quad (3.12)$$

donde N_D^+ y N_A^- representan, respectivamente, las concentraciones de impurezas dadoras y aceptoras ionizadas.

Por lo tanto, la ecuación de Poisson en un semiconductor genérico puede expresarse como sigue:

$$\nabla(\varepsilon \nabla \psi) = q(n - p + N_A^- - N_D^+) \quad (3.13)$$

3.1.2. Ecuaciones de continuidad

Volviendo a las ecuaciones de Maxwell, se observa que aplicando el operador divergencia a la ecuación (3.1) y usando la ecuación (3.3) se obtiene:

$$0 = \nabla \mathbf{J} + \frac{\partial \rho}{\partial t}, \quad (3.14)$$

ya que la divergencia de un rotacional es nula.

Podemos expresar la densidad de corriente \mathbf{J} en función de dos términos, uno asociado con la corriente de electrones, \mathbf{J}_n , y otro con la de huecos, \mathbf{J}_p , quedando de esta forma $\mathbf{J} = \mathbf{J}_n + \mathbf{J}_p$. Así, considerando que el perfil de impurezas es invariante en el tiempo y usando (3.11) y (3.14), obtenemos que en el interior del semiconductor se verifica la siguiente expresión:

$$-\nabla \mathbf{J}_p - q \frac{\partial p}{\partial t} = \nabla \mathbf{J}_n - q \frac{\partial n}{\partial t}. \quad (3.15)$$

Es posible obtener una ecuación para la densidad de corriente de electrones y otra equivalente para la de huecos si hacemos ambas partes de (3.15) iguales a una cantidad que denotaremos por qR :

$$\nabla \mathbf{J}_n - q \frac{\partial n}{\partial t} = qR, \quad (3.16)$$

$$\nabla \mathbf{J}_p + q \frac{\partial p}{\partial t} = -qR. \quad (3.17)$$

El término R puede interpretarse físicamente como una función que describe la generación o recombinación neta de electrones y huecos. Valores de R positivos implican que la recombinación de pares electrón-hueco prevalece sobre la generación de los mismos en el semiconductor. Valores negativos de R indican que es la generación la que predomina sobre la recombinación.

Bajo condiciones isotérmicas, si el transporte es por arrastre-difusión, las densidades de corriente de electrones y huecos son proporcionales a los gradientes de los cuasipotenciales de Fermi de electrones y huecos (ϕ_n y ϕ_p):

$$\mathbf{J}_n = -q\mu_n n \nabla(\phi_n), \quad (3.18)$$

$$\mathbf{J}_p = -q\mu_p p \nabla(\phi_p), \quad (3.19)$$

donde μ_n y μ_p son las movilidades de electrones y de huecos respectivamente. Físicamente, las movilidades están relacionadas con los tiempos medios de relajación de electrones y huecos, τ_n^r y τ_p^r , que representan el tiempo medio

entre dos procesos consecutivos de dispersión. Por ello la movilidad puede considerarse como una medida de la facilidad de movimiento de los portadores en el cristal, por lo que es evidente que las movilidades serán inversamente proporcionales a la cantidad de colisiones. En el simulador, en el caso del silicio, el valor de la movilidad de los electrones se obtiene utilizando un modelo analítico con dependencia en el dopado, la temperatura y los campos eléctricos transversal y paralelo [Sel84]. En primer lugar se parte de la expresión de la movilidad dependiente del dopado de Caughey-Thomas [CT67] modificada para incluir la dependencia con la temperatura:

$$\mu_{0,n} = \mu_{n,min} + \frac{\mu_{n,max} * \left(\frac{T}{300,0}\right)^{\nu_n} - \mu_{n,min}}{1 + \left(\frac{T}{300,0}\right)^{\xi_n} * \left(\frac{N_D+N_A}{n_{ref,n}}\right)^{\alpha_n}}, \quad (3.20)$$

donde los parámetros se obtienen a partir de ajustes a datos experimentales. Para obtener la expresión final de la movilidad para campos paralelos al transporte bajos, se incluye la dependencia con el campo perpendicular para tener en cuenta procesos en las interfaces,

$$\mu_{T,n} = \frac{\mu_{0,n}}{\sqrt{1 + \frac{E_{\perp,n}}{\mathbf{E}_{cn}}}}, \quad (3.21)$$

donde el campo crítico normal \mathbf{E}_{cn} se obtiene mediante ajuste a datos experimentales. Finalmente, se corrige la expresión (3.21) para incluir efectos de campos elevados en la dirección del transporte:

$$\mu_n = \frac{\mu_{T,n}}{\left[1 + \left(\frac{\mu_{T,n} E_{\parallel,n}}{v_{sat,n}}\right)^{\beta_n}\right]^{\frac{1}{\beta_n}}}, \quad (3.22)$$

donde $v_{sat,n}$ es la velocidad de saturación de los electrones en el material y $\beta_n = 2$ en el caso de los electrones. Para los huecos las expresiones son equivalentes, solo que los valores de los parámetros ($v_{sat,p}$, β_p , μ_{ec} , α , etc.) cambian.

3.1.3. Concentración de portadores

En este apartado obtenemos las expresiones para las concentraciones de portadores n y p en función del potencial electrostático ψ y de los cuasipotenciales de Fermi de electrones y de huecos ϕ_n y ϕ_p que utilizaremos posteriormente.

En un semiconductor con diferentes bandas (o valles) no equivalentes que intervengan en el transporte, la concentración de electrones en cada una de esas bandas, $n_j(E)$, depende del número de estados por unidad de volumen del cristal con esa energía $g_j(E)$, que es la función de densidad de estados, y de la probabilidad de que esos estados estén ocupados por un electrón $f_e(E)$, que viene dada por la función de Fermi-Dirac. Podemos escribir la concentración de electrones en la banda j como

$$n_j = \int_{E_{cj}}^{E_{supj}} f_e(E)g_j(E)dE, \quad (3.23)$$

donde E_{cj} es el mínimo de energía de la banda j y E_{supj} el nivel máximo de energía de la banda de conducción. Teniendo en cuenta que el valor del integrando anterior disminuye muy rápidamente a medida que la energía aumenta, y que es esencialmente cero para energías apenas pocos kT por encima de E_c , podemos extender el límite superior de la ecuación anterior a infinito y seguir obteniendo un valor similar de la integral. Por lo tanto, dicha integral queda como

$$n_j = \int_{E_{cj}}^{\infty} f_e(E)g_j(E)dE. \quad (3.24)$$

La función de Fermi-Dirac para los electrones es

$$f_e(E) = \frac{1}{1 + \exp \frac{E-E_{Fn}}{k_B T}}, \quad (3.25)$$

donde E_{Fn} es el cuasinivel de Fermi para los electrones.

Si tenemos en cuenta la no parabolicidad de la banda j mediante un factor de no parabolicidad B_j , podemos escribir la expresión de la densidad de estados $g_j(E)$ como

$$g_j = \frac{\sqrt{2}m_{dj}^{3/2}}{\pi^2\hbar^3} \left[(E - E_{cj})^{1/2} + B_j(E - E_{cj})^{3/2} \right], \quad (3.26)$$

siendo $\hbar = \frac{h}{2\pi}$ la constante reducida de Planck y m_{dj} la masa efectiva para la densidad de estados en la banda (o en el valle).

Substituyendo las fórmulas anteriores en (3.24), obtenemos la siguiente expresión para la concentración de portadores en la banda j :

$$n_j = N_{cj} \left[F_{1/2}(\eta_{cj}) + \frac{3}{2}k_B T B_j F_{3/2}(\eta_{cj}) \right], \quad (3.27)$$

donde $F_{1/2}(\eta_{cj})$ y $F_{3/2}(\eta_{cj})$ son las integrales de Fermi-Dirac de orden 1/2 y 3/2, respectivamente, η_{cj} viene dado por

$$\eta_{cj} = \frac{E_{Fn} - E_{cj}}{k_B T} \quad (3.28)$$

y N_{cj} representa la densidad efectiva de estados. Este parámetro se puede relacionar con la masa efectiva para la densidad de estados por:

$$N_{cj} = 2 \left(\frac{2m_{dj}k_B T \pi}{h^2} \right)^{3/2}. \quad (3.29)$$

Obtenemos la concentración total de electrones como la suma de las concentraciones para cada una de las bandas o valles:

$$n = \sum_j N_{cj} \left[F_{1/2}(\eta_{cj}) + \frac{3}{2} k_B T B_j F_{3/2}(\eta_{cj}) \right]. \quad (3.30)$$

En el modelo físico que hemos utilizado medimos el valor de la energía de los distintos niveles respecto a un nivel de energía constante que tomamos como referencia. Este nivel de energía, igual a cero, coincide con el nivel de vacío de un material tomado a su vez como referencia. Utilizamos el subíndice o para las constantes físicas y los niveles de energía constantes del material tomado como referencia, que no tiene porque corresponder con ninguno de los materiales de las regiones presentes en el dispositivo, pero que deben ser coherentes entre sí.

En la figura 3.1 mostramos los niveles de energía del material de referencia E_{oo} , E_{co} , E_{io} y E_{vo} junto con el nivel de vacío, E_o , el nivel mínimo de la banda de conducción j , E_{cj} , y el máximo de una banda de valencia l , E_{vl} , en el interior de cualquier región del dispositivo.

En el diagrama de energías de esta figura, la energía del mínimo de la banda de conducción del material de referencia, E_{co} , y del máximo de la banda de valencia, E_{vo} , se pueden expresar en función de su afinidad electrónica, χ_o , y de la anchura de su banda prohibida, E_{go} . Por lo tanto, $E_{co} = -\chi_o$ y $E_{vo} = -(\chi_o + E_{go})$.

La concentración intrínseca del material de referencia está relacionada con las densidades efectivas de estados de las bandas de valencia y conducción y con la anchura de la banda prohibida mediante la ecuación:

$$n_{io} = \sqrt{N_{co}N_{vo}} \exp\left(-\frac{E_{go}}{2k_B T}\right), \quad (3.31)$$

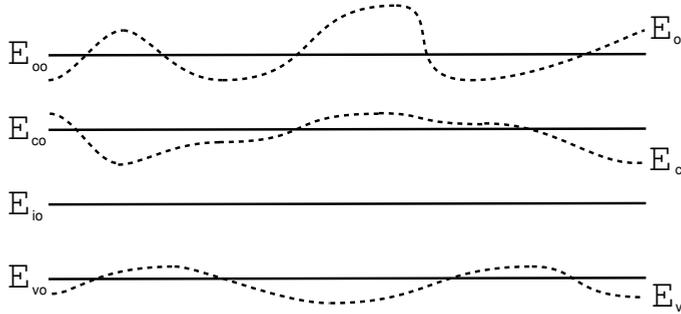


Figura 3.1: Niveles de energía del material de referencia junto con los niveles de energía de una región del semiconductor.

donde N_{co} es la densidad efectiva de estados en la banda de conducción y N_{vo} la densidad efectiva de estados en la banda de valencia del material de referencia.

Por otro lado, el nivel intrínseco del material de referencia, E_{io} , depende de la densidad efectiva de estados y de la concentración intrínseca. Se puede escribir como:

$$E_{io} = E_{co} + k_B T \ln \left(\frac{n_{io}}{N_{co}} \right). \quad (3.32)$$

Para obtener las expresiones de las concentraciones de portadores n y p , en primer lugar se relacionan los niveles de energía de las bandas en un semiconductor, esto es, nivel de vacío, E_o , mínimo de la banda de conducción j , E_{cj} y máximo de la banda de valencia l , E_{vl} y los cuasiniveles de Fermi para los electrones y los huecos, E_{Fn} y E_{Fp} , con el potencial electrostático, ψ , los cuasipotenciales de Fermi, ϕ_n y ϕ_p y los niveles de energía del material de referencia.

El nivel de vacío del semiconductor se obtiene a partir del potencial electrostático por:

$$E_o = -q\psi. \quad (3.33)$$

El nivel de energía mínimo de la banda de conducción j es:

$$E_{cj} = -q\psi - \chi_j, \quad (3.34)$$

y el máximo de la banda de valencia l :

$$E_{vl} = -q\psi - \chi_l - E_{gl}. \quad (3.35)$$

El cuasinivel de Fermi para los electrones es:

$$E_{Fn} = E_{io} - q\phi_n \quad (3.36)$$

y el cuasinivel de Fermi para los huecos:

$$E_{Fp} = E_{io} - q\phi_p. \quad (3.37)$$

Además, podemos expresar la ecuación (3.32):

$$\frac{n_{io}}{N_{co} \exp\left(\frac{E_{io} - E_{co}}{k_B T}\right)} = 1, \quad (3.38)$$

de tal manera que si multiplicamos la parte derecha de la ecuación (3.30) por el factor anterior obtenemos la siguiente expresión:

$$n = n_{io} \exp\left(\frac{E_{co} - E_{io}}{k_B T}\right) \sum_j \frac{N_{cj}}{N_{co}} \left[F_{1/2}(\eta_{cj}) + \frac{3}{2} k_B T B_j F_{3/2}(\eta_{cj}) \right]. \quad (3.39)$$

Teniendo en cuenta la ecuación (3.28), que $E_{co} = \psi_o$, $E_{cj} = -q\psi - \chi_j$ y expresando el cuasinivel de Fermi de los electrones en función de su cuasi-potencial de Fermi, $E_{Fn} = E_{io} - q\phi_n$, obtenemos la siguiente expresión para n :

$$n = \left\{ n_{io} \sum_j \frac{N_{cj}}{N_{co}} \frac{[F_{1/2}(\eta_{cj}) + \frac{3}{2} k_B T B_j F_{3/2}(\eta_{cj})]}{\exp(\eta_{cj})} \right. \\ \left. \exp\left(\frac{\chi_j - \chi_o}{k_B T}\right) \right\} \exp\left(\frac{q\psi - q\phi_n}{k_B T}\right). \quad (3.40)$$

Del mismo modo, podemos obtener una expresión similar para la concentración de huecos p :

$$p = \left\{ n_{io} \sum_l \frac{N_{vl}}{N_{vo}} \frac{[F_{1/2}(\eta_{vl}) + \frac{3}{2} k_B T B_j F_{3/2}(\eta_{vl})]}{\exp(\eta_{vl})} \right. \\ \left. \exp\left(-\frac{\chi_j - \chi_o + E_{gl} - E_{go}}{k_B T}\right) \right\} \exp\left(\frac{q\phi_p - q\psi}{k_B T}\right), \quad (3.41)$$

donde el subíndice j o l se refiere a las diferentes bandas que pueden intervenir en la población de portadores. El cuasipotencial de Fermi para los huecos se escribe como $q\phi_p = E_{io} - E_{Fp}$, y el parámetro $\eta_{vl} = \frac{E_{vl} - E_{Fp}}{k_B T}$.

En las ecuaciones anteriores, correspondientes a las concentraciones de electrones (3.40) y huecos (3.41), hemos indicado entre llaves las expresiones de las concentraciones intrínsecas efectivas de los electrones y huecos, n_{ien} y n_{iep} . Estas expresiones son:

$$n_{ien} = n_{io} \sum_j \frac{N_{cj}}{N_{co}} \frac{[F_{1/2}(\eta_{cj}) + \frac{3}{2}kTB_j F_{3/2}(\eta_{cj})]}{\exp(\eta_{cj})} \exp\left(\frac{\chi_j - \chi_o}{k_B T}\right), \quad (3.42)$$

$$n_{iep} = n_{io} \sum_l \frac{N_{vl}}{N_{vo}} \frac{[F_{1/2}(\eta_{vl}) + \frac{3}{2}kTB_j F_{3/2}(\eta_{vl})]}{\exp(\eta_{vl})} \exp\left(-\frac{\chi_j - \chi_o + E_{gl} - E_{go}}{k_B T}\right). \quad (3.43)$$

Los valores de las concentraciones intrínsecas efectivas de electrones y huecos, n_{ien} y n_{iep} , permiten tener en cuenta los efectos de degeneración del semiconductor, de la variación de los parámetros con la composición, e incluso de la existencia de varias bandas o valles, incluyendo efectos de no parabolicidad de esas bandas. Estas concentraciones son función de la posición, dependen de las características de los materiales en cada punto, N_c , N_v , χ y E_g , y además variarán a medida que se modifique la polarización del dispositivo puesto que el potencial y los cuasipotenciales de Fermi de los portadores cambian al variar la polarización. Por tanto, n_{ien} y n_{iep} , tienen un papel primordial en el modelo presentado.

Las ecuaciones (3.42) y (3.43) permiten escribir las siguientes expresiones para las concentraciones de electrones y de huecos de forma compacta como:

$$n = n_{ien} \exp\left(\frac{q\psi - q\phi_n}{k_B T}\right), \quad (3.44)$$

$$p = n_{iep} \exp\left(\frac{q\phi_p - q\psi}{k_B T}\right). \quad (3.45)$$

Las ecuaciones (3.42) y (3.43), junto con las ecuaciones (3.44) y (3.45), constituyen el modelo utilizado para expresar la concentración de portadores en un punto cualquiera de un semiconductor degenerado o no degenerado, en el que pueden intervenir varias bandas o valles en la población de electrones y huecos, permitiendo además estudiar bandas no parabólicas.

Podemos simplificar las ecuaciones anteriores para las concentraciones intrínsecas de electrones y huecos si cada tipo de portador se sitúa en una sola

banda de energía y si se prescinde de los factores de no parabolicidad de las bandas. En este caso,

$$n_{ien} = n_{io} \frac{N_c}{N_{co}} \exp\left(\frac{\chi - \chi_o}{k_B T}\right) \frac{F_{1/2}\left(\frac{\chi - \chi_o}{k_B T} - \ln\left(\frac{N_{co}}{n_{io}}\right) + \frac{q\psi - q\phi_n}{k_B T}\right)}{\exp\left(\frac{\chi - \chi_o}{k_B T} - \ln\left(\frac{N_{co}}{n_{io}}\right) + \frac{q\psi - q\phi_n}{k_B T}\right)}, \quad (3.46)$$

$$n_{iep} = n_{io} \frac{N_v}{N_{vo}} \exp\left(-\frac{\chi - \chi_o + E_g - E_{go}}{k_B T}\right) \frac{F_{1/2}\left(-\frac{\chi_j - \chi_o}{k_B T} - \frac{E_g - E_{go}}{k_B T} - \ln\left(\frac{N_{vo}}{n_{io}}\right) - \frac{q\phi_p - q\psi}{k_B T}\right)}{\exp\left(-\frac{\chi_j - \chi_o}{k_B T} - \frac{E_g - E_{go}}{k_B T} - \ln\left(\frac{N_{vo}}{n_{io}}\right) - \frac{q\phi_p - q\psi}{k_B T}\right)}. \quad (3.47)$$

3.1.4. Factor de generación-recombinación

En la derivación de las ecuaciones de continuidad (sección 3.1.2) introdujimos el parámetro R que representa la diferencia entre las tasas de recombinación y generación de pares electrón-hueco. La recombinación ocurre cuando un electrón en la banda de conducción salta a la banda de valencia neutralizando un hueco, mientras que la generación sucede cuando un electrón de valencia pasa a la banda de conducción produciendo un hueco. La generación necesita energía mientras que la recombinación la libera.

Cuando un semiconductor extrínseco está en equilibrio térmico existe un equilibrio dinámico entre los procesos de generación y recombinación, con lo que $R = 0$ (principio del balance detallado). Si el equilibrio se rompe, por ejemplo al aplicar un potencial externo al semiconductor, las concentraciones de portadores se modifican respecto a sus valores en el equilibrio. Empiezan a producirse entonces varios procesos de generación-recombinación que tienden a restaurar el equilibrio de forma que el exceso o déficit de portadores es estabilizado si la perturbación se mantiene, o eliminado si la causa perturbadora desaparece. Así, si la perturbación produce un exceso de portadores, la recombinación domina sobre la generación (es decir, $R > 0$) para eliminar dicho exceso; en caso contrario, la generación domina y $R < 0$.

La generación-recombinación es provocada por diversos tipos de procesos. Para cada uno de ellos es necesario obtener una expresión que determine el valor de R en función de las variables características del dispositivo. Entre los principales procesos responsables de la generación-recombinación destacan:

- Recombinación intrínseca también llamada directa, banda-a-banda o de transición de fotones.
- Recombinación extrínseca o de transición de dos partículas, de transición de fonones o Shockley-Read-Hall.
- Recombinación Auger o de transición de tres partículas.
- Recombinación superficial.

El primer mecanismo de generación-recombinación implica un proceso directo en el cual un electrón pasa de la banda de conducción a la de valencia (o viceversa) liberando (o absorbiendo) un fotón. Este proceso es importante para semiconductores de banda estrecha y semiconductores, como el GaAs, cuya estructura de bandas permite transiciones directas. La tasa de recombinación banda-a-banda, R_{BB} , es proporcional a la diferencia entre el producto de las concentraciones de portadores en ese momento y el que habría en equilibrio. A la constante de proporcionalidad se le conoce como coeficiente banda-a-banda, C_{BB} . Teniendo en cuenta que n_o y p_o representan las concentraciones de electrones y de huecos en equilibrio, esta recombinación se puede expresar como:

$$R_{BB} = C_{BB}(np - n_o p_o). \quad (3.48)$$

La forma más importante de generación-recombinación es la extrínseca. Este proceso tiene lugar a través de centros de recombinación (impurezas, imperfecciones de la red, etc). Así, la recombinación ocurre en dos pasos, en el primero un electrón de la banda de conducción es atrapado por el centro, y en el segundo dicho electrón pasa a la banda de valencia neutralizando un hueco. Este proceso es típicamente no-radiactivo. En él se libera (o absorbe en caso de generación) energía térmica o, equivalentemente, vibraciones de la red (fonones). El proceso de recombinación extrínseca se describe a través del denominado término de Shockley-Read-Hall:

$$R_{SRH} = \frac{np - n_o p_o}{\tau_p(n + n_T) + \tau_n(p + p_T)}, \quad (3.49)$$

que relaciona la recombinación global con las concentraciones de portadores, siendo τ_n y τ_p los tiempos de vida media de electrones y huecos, respectivamente. El término n_T (p_T) representaría la concentración de electrones (de huecos) que habría en la banda de conducción (en la banda de valencia), si el

nivel de Fermi en equilibrio coincidiese con el nivel de energía del centro de recombinación. Su valor es:

$$n_T = n_o \exp \frac{E_T - E_F}{k_B T}, \quad (3.50)$$

$$p_T = p_o \exp \frac{E_F - E_T}{k_B T}. \quad (3.51)$$

La recombinación Auger es un proceso no-radiactivo en el que la recombinación directa o vía centros ocurre simultáneamente a la colisión entre dos portadores del mismo tipo. Existen dos tipos de procesos Auger. En el primero la energía que se obtiene de la destrucción de un par electrón-hueco se emplea en aumentar la energía de un electrón de la banda de conducción mientras que, en el segundo esta energía se emplea en aumentar la de un hueco de la banda de valencia. Como en el primer proceso intervienen dos electrones la recombinación es proporcional a $n^2 p$ y en el segundo, como intervienen dos huecos ésta es proporcional a $n p^2$ por lo tanto:

$$R_A = C_{An}(n^2 p - n_o^2 p_o) + C_{Ap}(n p^2 - n_o p_o^2), \quad (3.52)$$

donde C_{An} es el coeficiente Auger para los electrones y C_{Ap} es el coeficiente Auger para los huecos.

3.2. Correcciones cuánticas

Con el escalado de los dispositivos, las dimensiones características se aproximan a la longitud de onda de de Broglie, lo que hace que las características eléctricas de los dispositivos se vean modificadas con respecto a las predicciones puramente clásicas. Como ya mencionamos previamente, existen técnicas de simulación que resuelven las ecuaciones del transporte cuántico. Sin embargo, estas técnicas son muy costosas computacionalmente, por lo que la aproximación más extendida actualmente es la inclusión de correcciones a los modelos semiclásicos para reproducir los efectos cuánticos.

Uno de los modelos más extendidos es el llamado *density gradient*, que añade una ecuación a mayores dependiente en el gradiente de la densidad de portadores. Un modelo estrechamente relacionado con este es el ECBE (*effective conduction band edge*). Además, se han propuesto otros modelos para las correcciones cuánticas, como el modelo “arrastre-difusión cuántico”, que resuelve las ecuaciones del modelo de arrastre-difusión para las distintas subbandas o los modelos de potencial efectivo, que incluyen en el cálculo de la

densidad de carga una convolución con una función para imitar la extensión espacial del paquete de ondas que representa al electrón. A continuación describiremos estos modelos, con especial énfasis en el *density gradient* y el ECBE ya que constituyen la base de nuestra implementación en el simulador.

3.2.1. El modelo *density gradient*

El modelo *density gradient* fue introducido por M. Ancona [AT87, AI89] con la intención de extender la validez del modelo continuo de arrastre-difusión a la zona de inversión fuerte en uniones MOS. Para ello se extendieron las ecuaciones de estado del gas de electrones con una dependencia en el gradiente de la densidad de portadores. Esto llevó a la obtención de un sistema de ecuaciones análogo al del modelo arrastre-difusión clásico en el que el potencial era sustituido por un potencial efectivo. A continuación describimos la formulación para electrones, aunque en el caso de los huecos sería totalmente equivalente. En este caso el potencial efectivo se escribe como:

$$\psi_{ef} = \psi + \psi_{qn}, \quad (3.53)$$

donde el llamado potencial cuántico ψ_{qn} tiene la forma:

$$\psi_{qn} = 2b_n \frac{\nabla^2 \sqrt{n}}{\sqrt{n}}, \quad (3.54)$$

donde la constante b_n tiene que ser obtenida a partir de derivaciones microscópicas de esta aproximación. Una aproximación sencilla es la siguiente. Partimos de la ecuación de Schrödinger para una partícula en un potencial externo $\psi(\mathbf{r})$ en la aproximación de la masa efectiva:

$$-\frac{\hbar^2}{2m^*} \nabla^2 \xi - q\psi(\mathbf{r})\xi = i\hbar \frac{\partial \xi}{\partial t}, \quad (3.55)$$

donde ξ es la función de onda de la partícula. Escribiendo la función de onda en función de su amplitud y su fase, $\xi = R \exp iS/\hbar$, sustituyendo en (3.55) y separando las partes real y compleja obtenemos las siguientes ecuaciones:

$$-\frac{\partial S}{\partial t} = \frac{(\nabla S)^2}{2m^*} - q\psi - \frac{\hbar^2}{2m^*} \frac{\nabla^2 R}{R}, \quad (3.56)$$

$$-\frac{\partial R^2}{\partial t} = \nabla \left(\frac{R^2 \nabla S}{m^*} \right). \quad (3.57)$$

Considerando un estado estacionario y condiciones de corriente nula, $S = -Et + cte$, donde E es la autoenergía. Usando esta expresión para la fase, la ecuación (3.56) queda:

$$E = -q\psi - \frac{\hbar^2}{2m^*} \frac{\nabla^2 \sqrt{P}}{\sqrt{P}}, \quad (3.58)$$

donde $P = R^2$ es la densidad de probabilidad, que, para un estado puro, es proporcional a la densidad de partículas. En el caso de los electrones, aproximando $P \approx n$, obtenemos:

$$E \approx -q\psi - \frac{\hbar^2}{2m^*} \frac{\nabla^2 \sqrt{n}}{\sqrt{n}} = -q\psi - \frac{\hbar^2}{4m^*} \left[\nabla^2 \ln n + \frac{1}{2} (\ln n)^2 \right]. \quad (3.59)$$

Comparando las ecuaciones (3.54) y (3.59) podemos obtener un valor para b_n :

$$b_n = \frac{\hbar^2}{4m^*q}. \quad (3.60)$$

En esta derivación asumimos una sola subbanda ocupada, lo que puede corresponderse con situaciones de baja temperatura o fuerte confinamiento. Si suponemos que hay más de una subbanda ocupada (alta temperatura o débil confinamiento), entonces esta constante es:

$$b_n = \frac{\hbar^2}{12m^*q}. \quad (3.61)$$

Una solución habitual es utilizar la expresión

$$b_n = \frac{\hbar^2}{4m^*qr}, \quad (3.62)$$

donde $1 \leq r \leq 3$ y su valor se selecciona en función de la situación a simular. Además, m^* también se suele utilizar como parámetro para calibrar los resultados con los dados por simulaciones autoconsistentes del sistema Poisson-Schrödinger.

Para especificar por completo la ecuación (3.54) hace falta definir unas condiciones de contorno. Las condiciones originales establecían un valor de la densidad de portadores arbitrariamente pequeño en las uniones semiconductor-aislante. Sin embargo, estas condiciones de contorno sobreestiman el confinamiento al suponer una barrera de potencial infinita. Más recientemente

se propusieron otras condiciones de contorno que fijan la derivada sobre la interfaz [JPM04], teniendo en cuenta la posible penetración de la función de onda en el óxido. En este caso se asume que la penetración en el óxido de la densidad de portadores es de la forma

$$n(z) = n(0)e^{-2z/x_p}, \quad (3.63)$$

donde se asume que z es la dirección normal a la interfaz, que se encuentra en $z = 0$, y que

$$x_p = \frac{\hbar}{\sqrt{2m_{ox}^* \Phi_B}}. \quad (3.64)$$

Este valor es la profundidad de penetración característica obtenida a partir de la aproximación Wentzel-Kramers-Brillouin (WKB). Los parámetros de los que depende esta expresión son la masa efectiva en el óxido, m_{ox}^* , y la altura de la barrera semiconductor-óxido, Φ_B . A partir de (3.63) se obtiene el valor de la derivada normal sobre la interfaz,

$$\hat{n} b_n \nabla \sqrt{n} = -\frac{b_{ox}}{x_p} \sqrt{n}, \quad (3.65)$$

donde

$$b_{ox} = \frac{\hbar^2}{12m_{ox}^* q}. \quad (3.66)$$

El parámetro m_{ox}^* se escoge de tal forma que coincida con del simulaciones Poisson-Schrödinger autoconsistentes unidimensionales de condensadores MOS.

3.2.2. El modelo *effective conduction band edge*

El modelo *density gradient* presentado en la sección anterior tiene como principal problema la necesidad de calcular derivadas segundas sobre funciones de la densidad de portadores. Esto puede ser de importancia secundaria en simulaciones arrastre-difusión, pero será fundamental para utilizar la corrección cuántica en simulaciones Monte Carlo, donde la densidad de portadores es una cantidad muy ruidosa.

El modelo *effective conduction band edge* (ECBE) [wTW04] trata de transformar la dependencia en la densidad de portadores en una dependencia en los potenciales. Para ello se asume que la densidad de portadores depende de

forma exponencial del potencial. En el caso de los electrones esta dependencia es de la forma

$$n \propto e^{-\psi_{ef}}. \quad (3.67)$$

Sustituyendo esta dependencia en la ecuación (3.59), obtenemos la ecuación que debemos resolver en el modelo ECBE:

$$\psi_{ef} = \psi + \frac{q\hbar^2}{4rm^*k_B T} \left[\nabla^2 \psi_{ef} + \frac{q}{2k_B T} (\nabla \psi_{ef})^2 \right]. \quad (3.68)$$

Esta ecuación tiene el problema de que la masa utilizada en las direcciones de confinamiento y transporte es la misma. Sin embargo, el efecto de las correcciones cuánticas puede ser diferente y podemos estar interesados en calibrar su efecto en estas dos direcciones de forma independiente. Esto es necesario, por ejemplo, para reproducir el efecto túnel de fuente a drenador, sobreestimado si usamos un valor demasiado bajo de la masa longitudinal [BAW02]. En este caso hemos de partir de la ecuación (3.55) modificada para incluir el tensor de masa efectiva $\frac{1}{m^*}$:

$$-\frac{\hbar^2}{2} \nabla \left[\left(\frac{1}{m^*} \right) \nabla \xi \right] + q\psi(\mathbf{r})\xi = i\hbar \frac{\partial \xi}{\partial t}. \quad (3.69)$$

Siguendo los mismos pasos que antes llegamos a la siguiente ecuación para el potencial efectivo:

$$\psi_{ef} = \psi + \frac{q\hbar^2}{4rk_B T} \left\{ \nabla \left[\left(\frac{1}{m^*} \right) \nabla \psi_{ef} \right] + \frac{q}{2k_B T} \left[\nabla \psi_{ef} \left(\frac{1}{m^*} \right) \nabla \psi_{ef} \right] \right\}. \quad (3.70)$$

3.2.3. Otras aproximaciones

Aparte de los dos modelos presentados hasta ahora, existen otros métodos para aproximar los efectos cuánticos. En el caso de simulaciones arrastre-difusión, una de las opciones es utilizar el modelo “arrastre-difusión cuántico” [BGG⁺08]. En este caso en primer lugar se obtienen las subbandas a partir del sistema Poisson-Schrödinger con ciertas simplificaciones y luego se utiliza el modelo de arrastre-difusión para cada subbanda. Finalmente, para obtener la corriente total se suma a todas las subbandas:

$$\mathbf{J}_n = \sum_i (\mu_n^i n^i \nabla E_{sb}^i + qD_n^i \nabla n_i), \quad (3.71)$$

donde E_{sb} está compuesta por dos términos, el potencial eléctrico y un potencial de corrección cuántica que depende de la subbanda y que, en estado estacionario, se corresponde con la energía cinética asociada al movimiento vibracional normal a la interfaz semiconductor-óxido.

Otro modelo es el del potencial efectivo [FAV00], muy utilizado en simulaciones Monte Carlo. En este caso el potencial efectivo en el que se mueven las partículas se obtiene a partir de la convolución del potencial electrostático y una función que representa la extensión espacial del paquete de ondas asociado al electrón:

$$\psi_{ef} = \int d\mathbf{r}' \psi(\mathbf{r}') F(\mathbf{r} - \mathbf{r}'), \quad (3.72)$$

donde F es la función que se usa para representar el tamaño finito de las partículas. La formulación original del método [FAV00] utilizaba una gaussiana para esta representación. Sin embargo, aunque este sigue siendo el método más común, también se ha propuesto el uso de distribuciones de Pearson para representar a las partículas [JBSM⁺08]. Este último método permite solucionar el principal problema de que presentaba esta aproximación, la obtención de una distribución de portadores bastante diferente a la obtenida al resolver el sistema Poisson-Schrödinger.

3.3. Escalado de las variables

En las secciones anteriores hemos descrito las ecuaciones de los modelos usando su forma original. Sin embargo, para mejorar las propiedades numéricas de estas ecuaciones en la simulación, se suele introducir un escalado de las variables de tal forma que se obtengan cantidades adimensionales y que se aislen los parámetros dimensionales relevantes de los que depende el modelo.

Si $\tilde{\omega}$ es el factor de escalado de la variable ω , el valor de la variable escalada y adimensional ω^* vendrá dado por

$$\omega^* = \frac{\omega}{\tilde{\omega}}. \quad (3.73)$$

Existen diferentes formas de escalar las variables, algunas de ellas recogidas en [DeM68a, DeM68b, Sel84, Mar86]. En nuestro caso el escalado utilizado está recogido en [Mar86] y parte de las siguientes premisas:

- Las longitudes están referidas a la longitud característica del dispositivo \tilde{l} elegida de tal forma que l sea del mismo orden de magnitud que el

diámetro del dominio semiconductor:

$$\tilde{l} = O(\text{diam}(\Omega)). \quad (3.74)$$

- Los potenciales están escalados por el potencial térmico $V_T = \frac{k_B T}{q}$.
- Las concentraciones de portadores se escalan por una concentración característica \tilde{C} que se elige de tal forma que resulte del mismo orden de magnitud que la máxima concentración de impurezas:

$$\tilde{C} = O\left(\max_{\Omega} |C(x)|\right). \quad (3.75)$$

De esta forma, las concentraciones escaladas máximas serán del orden de la unidad.

- Las movilidades se escalan por una movilidad de referencia $\tilde{\mu}$ elegida de tal forma que $\mu_n/\tilde{\mu}$ sea como máximo del orden de la unidad.

Los valores de los factores de escala resultantes para los distintos parámetros pueden encontrarse en [Mar86]. En la tabla 3.1 se indican algunos de los principales factores de escalado utilizados y su valor típico.

Es posible reescribir, en función de las variables escaladas, las ecuaciones que modelan el comportamiento de los dispositivos en la aproximación de arrastre-difusión. Denotando las nuevas variables escaladas con los mismos símbolos que las antiguas, las ecuaciones de Poisson y continuidad se expresan como:

$$\lambda^2 \nabla^2 \psi = (n - p - C), \quad (3.76)$$

$$\nabla \mathbf{J}_n - \frac{\partial n}{\partial t} = R, \quad (3.77)$$

$$\nabla \mathbf{J}_p + \frac{\partial p}{\partial t} = -R, \quad (3.78)$$

$$\psi_{qn} = 2b_n \frac{\nabla^2 \sqrt{n}}{\sqrt{n}}, \quad (3.79)$$

$$\psi_{qp} = -2b_p \frac{\nabla^2 \sqrt{p}}{\sqrt{p}}. \quad (3.80)$$

| Símbolo | Significado | Factor escala | Valor típico |
|----------------|--------------------------|---|---|
| x | posición | \tilde{l} | $5 \times 10^{-3} \text{ cm}$ |
| t | tiempo | $\frac{\tilde{l}^2}{\tilde{\mu}V_T}$ | $9,7 \times 10^{-7} \text{ s}$ |
| ϕ | potencial | V_T | $0,0259V$ |
| n | concent. de e^- | \tilde{C} | 10^{19} cm^{-3} |
| p | concent. de h^+ | \tilde{C} | 10^{17} cm^{-3} |
| \mathbf{J}_n | dens. corriente de e^- | $\frac{qV_T\tilde{C}\tilde{\mu}}{\tilde{l}}$ | $83A\text{cm}^{-2}$ |
| \mathbf{J}_p | dens. corriente de h^+ | $\frac{qV_T\tilde{C}\tilde{\mu}}{\tilde{l}}$ | $83A\text{cm}^{-2}$ |
| μ_n | movilidad de e^- | $\tilde{\mu}$ | $1000\text{cm}^2V^{-1}s^{-1}$ |
| μ_p | movilidad de h^+ | $\tilde{\mu}$ | $1000\text{cm}^2V^{-1}s^{-1}$ |
| C | concent. de dopantes | \tilde{C} | 10^{17} cm^{-3} |
| R | recombinación | $\frac{V_T\tilde{\mu}\tilde{C}}{\tilde{l}^2}$ | $1,1 \times 10^{23} \text{ cm}^{-3} \text{ s}^{-1}$ |
| τ_n | tiempo vida media e^- | $\frac{\tilde{l}^2}{\tilde{\mu}V_T}$ | $9,7 \times 10^{-7} \text{ s}$ |
| τ_p | tiempo vida media h^+ | $\frac{\tilde{l}^2}{\tilde{\mu}V_T}$ | $9,7 \times 10^{-7} \text{ s}$ |

Tabla 3.1: Factores de escalado

El parámetro λ , cuyo cuadrado multiplica al laplaciano del potencial en la ecuación de Poisson escalada 3.76, viene dado por:

$$\lambda = \frac{\lambda_D}{\tilde{l}}, \quad \lambda_D = \sqrt{\frac{\varepsilon V_T}{q\tilde{C}_o}}, \quad (3.81)$$

siendo λ_D la longitud de Debye característica del dispositivo. En la mayoría de las situaciones, λ es un parámetro muy pequeño, típicamente del orden de 10^{-3} a 10^{-5} .

3.4. Discretización de las ecuaciones de arrastre-difusión y *density gradient*

Los modelos descritos en la sección anterior están formados por las ecuaciones diferenciales que modelan el comportamiento de los dispositivos objeto de estudio. La resolución de este sistema de ecuaciones diferenciales no se puede hacer analíticamente, por lo que hemos de buscar soluciones utilizando métodos numéricos. Para ello discretizaremos las ecuaciones utilizando el

método de los elementos finitos (MEF). Esta es una técnica numérica que resuelve, de forma aproximada, problemas descritos por ecuaciones diferenciales en derivadas parciales o que pueden ser descritos como minimización de un funcional. En lugar de aproximar la ecuación directamente, como ocurre en otros métodos, se reformula el problema de forma variacional, utilizando una integral sobre el dominio del problema que involucra la ecuación diferencial. Después, mediante el método de Ritz-Galerkin [BCO81], se busca solución en un subespacio finito del espacio infinito de funciones base. Normalmente este subespacio consiste en polinomios de orden bajo definidos sobre una partición del dominio, en nuestro caso compuesta por tetraedros. El resultado de esta aproximación es un sistema algebraico disperso de dimensión finita.

A continuación se muestra una breve introducción al método de los elementos finitos, para posteriormente aplicarlo a las ecuaciones que resolveremos en las simulaciones de dispositivos: la ecuación de Poisson, las ecuaciones de continuidad y las ecuaciones del modelo *density gradient*.

3.4.1. El método de los elementos finitos

Empezamos considerando el siguiente problema en el dominio Ω para introducir el método de los elementos finitos:

$$-\nabla^2 u(x) = f \quad \forall x \in \Omega, \quad (3.82)$$

$$u(x) = 0 \quad \forall x \in \partial\Omega. \quad (3.83)$$

Esta formulación del problema es la llamada *formulación fuerte*. En este caso buscamos una función u que satisfaga la ecuación y las condiciones de contorno en todo punto del dominio Ω . Esto suele ser una exigencia demasiado estricta para poder encontrar la solución, por lo que usaremos una formulación que imponga condiciones más suaves a la solución y a sus derivadas, la *formulación débil* o *variacional*. Esta formulación se adapta bien tanto a soluciones irregulares (con derivada discontinua, por ejemplo) como a soluciones “bien comportadas”.

El objetivo es reformular el problema para buscar funciones u que satisfagan la ecuación diferencial y las condiciones de contorno en el sentido de “medias ponderadas”. Con esto se quiere decir que se satisfaga:

$$-\int_{\Omega} \nabla^2 uv \, dx = \int_{\Omega} f v \, dx, \quad (3.84)$$

para todos los miembros v de un tipo de funciones. Estas funciones v , llamadas *funciones de ponderación* o *funciones de test*, pueden ser cualquier función de x tal que la integral tenga sentido. Al conjunto de funciones de este tipo, que además verifiquen $v(x) = 0$ en $\partial\Omega$, lo llamaremos H . La formulación del problema queda de la siguiente forma: encontrar una función u tal que

$$\int_{\Omega} (\nabla^2 u + f)v \, dx = 0 \quad \forall v \in H, \quad (3.85)$$

$$u = 0 \quad \forall x \in \partial\Omega. \quad (3.86)$$

Al término $r(x) = \nabla^2 u + f$ se le llama residuo. La solución de (3.82) también lo es de (3.85) y además es su única solución.

La formulación débil presentada tiene un problema, el conjunto de funciones al que pertenece la función u , \tilde{H} , no tiene que ser el mismo que el conjunto H . Para evitar esta falta de simetría podemos reformular el problema tomando, sin pérdida de generalidad, el mismo conjunto de funciones H_0^1 para la función solución y para la función de test, verificando además las condiciones de frontera.

Además, considerando la siguiente relación,

$$\nabla(g\nabla h) = \nabla g\nabla h + g\nabla^2 h \quad (3.87)$$

y el teorema de Green-Gauss, obtenemos:

$$\int_{\Omega} \nabla g\nabla h \, dx = - \int_{\Omega} g\nabla^2 h \, dx + \int_{\partial\Omega} g \frac{\partial h}{\partial n} \, dS, \quad (3.88)$$

en el que $\frac{\partial h}{\partial n}$ es la derivada normal a la superficie S . Para nuestro ejemplo obtenemos la siguiente formulación variacional del problema:

$$\int_{\Omega} \nabla u\nabla v \, dx = \int_{\Omega} f v \, dx, \quad \forall v \in H_0^1. \quad (3.89)$$

El conjunto H_0^1 es la *clase de funciones admisibles para el problema* (3.89). El término más irregular de la expresión es $\nabla u\nabla v$. Como v puede ser cualquier función del conjunto de funciones admisibles y puede suceder que $v = u$, $(\nabla v)^2$ debe ser integrable. Si verifica esto se dice que tiene primera derivada cuadrado integrable. Por tanto, definiremos H_0^1 como el conjunto de funciones que se anulen en la frontera del dominio y que tengan primera derivada cuadrado integrable:

$$\left. \begin{array}{l} \int_{\Omega} (\nabla w)^2 \, dx < \infty \\ w = 1 \quad \forall x \in \partial\Omega \end{array} \right\}. \quad (3.90)$$

El problema que queremos resolver a continuación es el de buscar soluciones aproximadas a (3.89). H_0^1 es un espacio lineal de funciones de dimensión infinita. Esto implica que la función u puede expresarse de la siguiente forma usando una base $\{\phi_i\}_{i=1}^\infty$:

$$u(x) = \sum_{i=1}^{\infty} \alpha_i \phi_i(x). \quad (3.91)$$

El método de Galerkin consiste en buscar una aproximación de la solución, u_N , en el subespacio N -dimensional de H_0^1 definido por un grupo de N funciones $\{\phi_1, \phi_2, \dots, \phi_N\}$ al que llamaremos $H_0^{(N)}$:

$$u_N(x) = \sum_{i=1}^N \alpha_i \phi_i(x). \quad (3.92)$$

Con esta aproximación, el problema queda ahora de la siguiente forma: encontrar $u_N \in H_0^{(N)}$ tal que

$$\int_{\Omega} (\nabla u_N \nabla v_N) dx = \int_{\Omega} f v_N dx \quad \forall v_N \in H_0^{(N)}. \quad (3.93)$$

Dado que los ϕ_i son conocidos, la solución aproximada quedará determinada por los N coeficientes α_i . Estos coeficientes se conocen como grados de libertad de la aproximación. Para obtener sus valores sustituimos las aproximaciones de u y v en (3.93). Llegamos a una expresión de la siguiente forma:

$$\sum_{i=1}^N \beta_i \left(\sum_{j=1}^N K_{ij} \alpha_j - F_i \right) = 0, \quad (3.94)$$

donde

$$K_{ij} = \int_{\Omega} \nabla \phi_i \nabla \phi_j dx \quad (3.95)$$

y

$$F_i = \int_{\Omega} f \phi_i dx, \quad (3.96)$$

con $i = 1, 2, \dots, N$.

La matriz $\mathbf{K} = [K_{ij}]$, de tamaño $N \times N$, es habitualmente llamada *matriz de rigidez*, y $\mathbf{F} = \{F_i\}$ *vector de carga*. Como los coeficientes del desarrollo

de la función de test (β_i) son arbitrarios, (3.94) representa un sistema de N ecuaciones que deben satisfacer los α_j :

$$\sum_{j=1}^N K_{ij} \alpha_j - F_i = 0, \quad i = 1, 2, \dots, N. \quad (3.97)$$

Las funciones ϕ_i se eligieron de forma que fueran independientes, por lo que la matriz \mathbf{K} será invertible. Por tanto, la solución que estamos buscando, el vector $\{\alpha_i\}$, se puede obtener resolviendo el sistema (3.94):

$$\alpha_j = \sum_{i=1}^N (K^{-1})_{ji} F_i. \quad (3.98)$$

Las propiedades numéricas de la matriz \mathbf{K} están totalmente determinadas por la elección que hagamos de las funciones ϕ_i y por la geometría del dominio Ω .

El método de los elementos finitos proporciona una forma general y sistemática para generar las funciones base del espacio de dimensión finita $H_0^{(N)}$. Este método da lugar a unas funciones básicas que se definen a trozos sobre subregiones del dominio llamadas elementos finitos que en el caso tridimensional pueden ser tetraedros, cubos, prismas, pirámides, etc.

Con el fin de simplificar la construcción de las funciones base ϕ_i se definen un conjunto de funciones elementales φ_l^e , conocidas como funciones de forma, sobre cada elemento finito. El índice $e = 1, \dots, N$ referencia los elementos de la partición y el índice l el vértice P del elemento que se está tratando, con $l = 1, 2, 3, 4$ si se usan elementos tetraédricos. Las funciones φ_l^e se pueden definir como:

$$\varphi_l^e(P_i) = \begin{cases} 1 & i = l \\ 0 & i \neq l \end{cases}$$

De esta forma, cualquier función base ϕ_i se puede definir sumando las contribuciones de todas las funciones de forma definidas sobre el vértice i -ésimo,

$$\phi_i = \sum_{e \ni i} \varphi_i^e. \quad (3.99)$$

Además de esto vamos a introducir un elemento de referencia normalizado, el elemento maestro o patrón, sobre el que se realizarán todos los cálculos para simplificar las operaciones. En el caso de elementos tetraédricos las funciones

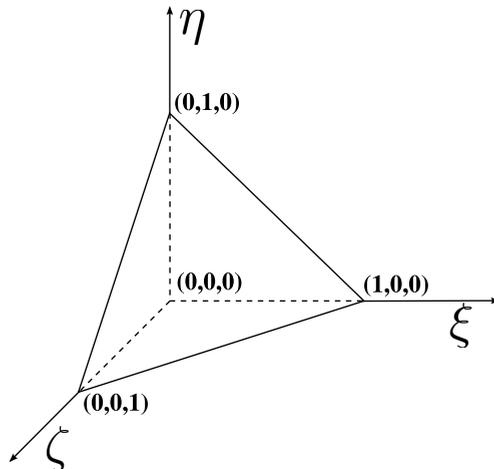


Figura 3.2: Elemento maestro tetraédrico de cuatro nodos.

de forma para este tetraedro, representado en la figura 3.2, son,

$$\hat{\varphi}_0 = 1 - \xi - \eta - \zeta, \quad (3.100a)$$

$$\hat{\varphi}_1 = \xi, \quad (3.100b)$$

$$\hat{\varphi}_2 = \eta, \quad (3.100c)$$

$$\hat{\varphi}_3 = \zeta. \quad (3.100d)$$

Como se puede ver en la figura 3.2, $\xi, \eta, \zeta \in [0, 1]$. La introducción de este elemento maestro simplifica los cálculos de la matriz de rigidez y del vector de carga. El cálculo de las integrales sobre los elementos originales sería una labor más complicada, ya que tendríamos que considerar los límites de integración diferentes para cada elemento. Al introducir una transformación entre un elemento de referencia $\hat{\Omega}$ y un elemento arbitrario Ω_e podemos realizar las operaciones sobre Ω_e en $\hat{\Omega}$. Esta transformación se hace realizando un mapeo T_e de $\hat{\Omega}$ en Ω_e definido por el cambio de coordenadas

$$T_e : \left. \begin{array}{l} x = x(\xi, \eta, \zeta) \\ y = y(\xi, \eta, \zeta) \\ z = z(\xi, \eta, \zeta) \end{array} \right\} \quad (3.101)$$

Al introducir esta transformación podemos ver la generación de la malla de E elementos como una secuencia de transformaciones $\{T_1, T_2, \dots, T_E\}$ en la que cada elemento Ω_e es la imagen del elemento maestro $\hat{\Omega}$ bajo el mapeo T_e .

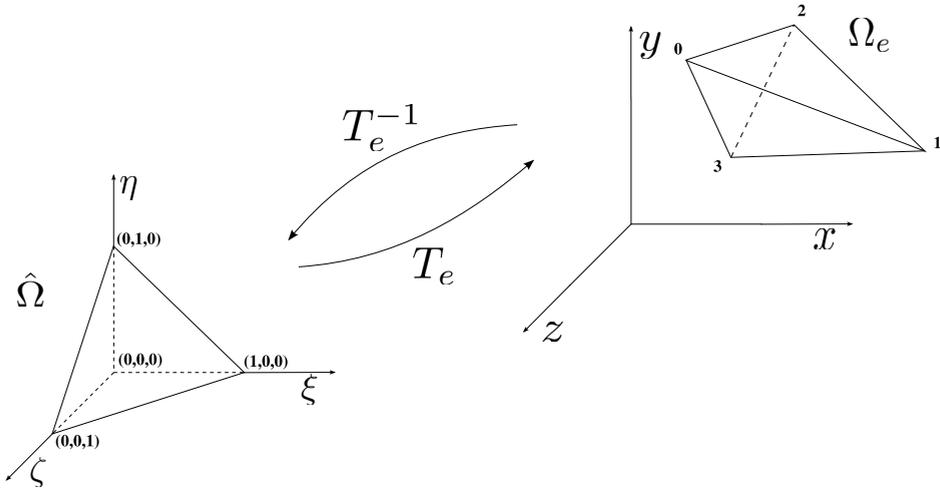


Figura 3.3: Transformación entre el tetraedro maestro y un tetraedro arbitrario.

Una ilustración de este proceso de mapeo se muestra en la figura 3.3. Podemos construir estas transformaciones a utilizando las funciones de forma:

$$x = \sum_{j=0}^3 x_j \hat{\varphi}_j(\xi, \eta, \zeta), \quad (3.102a)$$

$$y = \sum_{j=0}^3 y_j \hat{\varphi}_j(\xi, \eta, \zeta), \quad (3.102b)$$

$$z = \sum_{j=0}^3 z_j \hat{\varphi}_j(\xi, \eta, \zeta), \quad (3.102c)$$

donde $P_i = (x_i, y_i, z_i)$ con $i = 0, 1, 2, 3$ son los cuatro vértices del tetraedro Ω_e . En primer lugar podemos ver que las funciones x, y, z son continuamente diferenciables con respecto a ξ, η, ζ . En este caso los diferenciales $d\xi, d\eta, d\zeta$ se transforman en dx, dy, dz de la siguiente forma:

$$\begin{pmatrix} dx \\ dy \\ dz \end{pmatrix} = \begin{pmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} & \frac{\partial x}{\partial \zeta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} & \frac{\partial y}{\partial \zeta} \\ \frac{\partial z}{\partial \xi} & \frac{\partial z}{\partial \eta} & \frac{\partial z}{\partial \zeta} \end{pmatrix} \begin{pmatrix} d\xi \\ d\eta \\ d\zeta \end{pmatrix}. \quad (3.103)$$

La matriz de derivadas parciales, que se denota por J_e , es la matriz jacobiana de la transformación T_e . Podemos escribir la matriz de forma explícita si insertamos las expresiones (3.100) y (3.102) en (3.103),

$$J_e = \begin{pmatrix} x_1 - x_0 & x_2 - x_0 & x_3 - x_0 \\ y_1 - y_0 & y_2 - y_0 & y_3 - y_0 \\ z_1 - z_0 & z_2 - z_0 & z_3 - z_0 \end{pmatrix}. \quad (3.104)$$

La matriz jacobiana de la transformación nos permite el cálculo directo de integrales y gradientes utilizando el elemento maestro:

$$\int_{\Omega_e} g(x, y, z) dx dy dz = \int_{\hat{\Omega}} \hat{g}(\xi, \eta, \zeta) |J_e| d\xi d\eta d\zeta, \quad (3.105)$$

$$\nabla g(x, y, z) = J_e^{-t} \hat{\nabla} \hat{g}(\xi, \eta, \zeta), \quad (3.106)$$

siendo $\hat{g}(\xi, \eta, \zeta) = g(x(\xi, \eta, \zeta), y(\xi, \eta, \zeta), z(\xi, \eta, \zeta))$, $|J_e|$ el determinante de la matriz jacobiana y J_e^{-t} la inversa de la traspuesta de esta misma matriz.

Además de estas operaciones también deberemos, en general, tratar integrales de superficie. Para esto introducimos un conjunto de transformaciones $\{w_s\}$ que proyectan un elemento patrón de un espacio bidimensional sobre cada cara $\partial\hat{\Omega}$. Finalmente, con la definición de la transformación t_e que proyecta esta cara $\partial\hat{\Omega}$ en la cara $\partial\Omega_e$ del elemento original, podemos obtener la integral de superficie

$$\int_{\partial\Omega_e} h dS_e = \int_{\hat{S}} (\hat{h} \circ w_s) |j| dS, \quad (3.107)$$

donde $\hat{h} = h \circ t_e$ y $|j|$ es el determinante de la matriz jacobiana de la transformación.

3.4.2. Ecuación de Poisson

En este primer caso mostramos el proceso de discretización de la ecuación de Poisson no escalada (3.10) en lugar de (3.76). Así, la ecuación a discretizar es:

$$\nabla(\varepsilon \nabla \psi) = q(n - p + N_A^- - N_D^+) = q(n - p - C) \quad x \in \Omega, \quad (3.108)$$

sujeta a las siguientes condiciones de contorno:

$$\left. \frac{\partial \psi}{\partial n} \right|_{\partial\Omega_N} = 0, \quad (3.109)$$

$$\psi \Big|_{\partial\Omega_D} = \psi_D \Big|_{\partial\Omega_D}, \quad (3.110)$$

donde $\partial\Omega_N$ son las fronteras tipo Neumann y $\partial\Omega_D$ las fronteras tipo Dirichlet, que en nuestro caso son los contactos de drenador, puerta y fuente. Si además se tiene en cuenta la presencia de una carga interfacial Q_I entre dos materiales a y b , entonces es necesario considerar otra condición de contorno,

$$\varepsilon_a \frac{\partial\psi}{\partial n} \Big|_{\partial\Omega_{I,a}} - \varepsilon_b \frac{\partial\psi}{\partial n} \Big|_{\partial\Omega_{I,b}} = Q_I. \quad (3.111)$$

Utilizando la misma metodología que en la sección anterior podemos discretizar la ecuación de Poisson (3.108) sujeta a las condiciones de contorno (3.109), (3.110) y (3.111).

Sea $H_0^1(\Omega)$ el espacio de funciones con derivada cuadrado integrable y que satisfacen las condiciones de contorno especificadas. La formulación débil del problema anterior se obtiene multiplicando la ecuación 3.108 por una función de test arbitraria ξ ,

$$\nabla(\varepsilon\nabla\psi)\xi = q(n-p-C)\xi, \quad \forall \xi \in H_0^1(\Omega). \quad (3.112)$$

Integrando en Ω y aplicando el teorema de Green-Gauss al primer término de la ecuación se obtiene

$$-\int_{\Omega} \varepsilon(\nabla\psi)^t \nabla\xi \, d\Omega + \int_{\partial\Omega} \varepsilon \frac{\partial\psi}{\partial n} \xi \, dS = \int_{\Omega} q(n-p-C)\xi \, d\Omega, \quad \forall \xi \in H_0^1 \quad (3.113)$$

Usando el método de Galerkin buscamos una solución aproximada de esta ecuación en un subespacio de dimensión finita H_0^K del espacio H_0^1 . Sea $\{\theta_i\}_{i=1}^K$ base de $H_0^K \subset H_0^1$. Expresando las funciones de test en esta base obtenemos K ecuaciones,

$$-\int_{\Omega} \varepsilon(\nabla\psi)^t \nabla\theta_i \, d\Omega + \int_{\partial\Omega} \varepsilon \frac{\partial\psi}{\partial n} \theta_i \, dS = \int_{\Omega} q(n-p-C)\theta_i \, d\Omega, \quad \forall i = 1, \dots, K. \quad (3.114)$$

Si además expresamos las aproximaciones de las funciones ψ , n , p y C en esta misma base y aplicamos las condiciones de contorno sobre una interfaz $\partial\Omega_I$

con carga interfacial Q_I , obtenemos el siguiente sistema de ecuaciones,

$$\begin{aligned} & \sum_{j=1}^K \psi_j \int_{\Omega} \varepsilon (\nabla \theta_j)^t \nabla \theta_i \, d\Omega + \int_{\partial\Omega_I} Q_I \theta_i \, dS + \\ & + \sum_{j=1}^K \int_{\Omega} q(n_j - p_j - C_j) \theta_j \theta_i \, d\Omega = 0, \\ & \forall i = 1, \dots, K. \end{aligned} \quad (3.115)$$

Como dijimos antes, con el objetivo de facilitar los cálculos, introducimos las funciones base y calculamos las integrales sobre el elemento patrón. De esta forma obtenemos la expresión final para la resolución de la ecuación de Poisson:

$$\begin{aligned} & \sum_{j=1}^K \psi_j \sum_{e \ni i, j} \int_{\hat{\Omega}} \varepsilon (\hat{\nabla} \hat{\varphi}_j)^t J_e^{-1} J_e^{-t} \hat{\nabla} \hat{\varphi}_i |J_e| \, d\hat{\Omega} + \sum_{s \ni i} \int_{\partial\hat{\Omega}_I} Q_I \hat{\varphi}_i |j_s| \, d\hat{S} + \\ & + \sum_{j=1}^K q(n_j - p_j - C_j) \sum_{e \ni i, j} \int_{\hat{\Omega}} \hat{\varphi}_j \hat{\varphi}_i |J_e| \, d\hat{\Omega} = \\ & = 0, \quad \forall i = 1, \dots, K. \end{aligned} \quad (3.116)$$

Aplicando las reglas de escalado a este sistema de ecuaciones obtendríamos el sistema final que se usaría en la simulación.

3.4.3. Ecuaciones de continuidad

En la discretización de las ecuaciones de continuidad de electrones y huecos en estado estacionario se toman como punto de partida las siguientes expresiones escaladas ($\mathbf{J}_n \rightarrow q\mathbf{J}_n$):

$$\nabla \mathbf{J}_n = R, \quad (3.117)$$

$$\nabla \mathbf{J}_p = -R, \quad (3.118)$$

$$\mathbf{J}_n = -\mu_n n \nabla(\phi_n), \quad (3.119)$$

$$\mathbf{J}_p = -\mu_p p \nabla(\phi_p), \quad (3.120)$$

$$R = R_{BB} + R_A + R_{SRH}, \quad (3.121)$$

$$\forall x \in \Omega,$$

junto con las condiciones contorno,

$$n|_{\partial\Omega_D} = n_{eq}, \quad p|_{\partial\Omega_D} = p_{eq}, \quad (3.122)$$

$$\left. \frac{\partial n}{\partial \nu} \right|_{\partial\Omega_N} = 0, \quad \left. \frac{\partial p}{\partial \nu} \right|_{\partial\Omega_N} = 0. \quad (3.123)$$

Una primera aproximación en la búsqueda de la solución numérica de estas ecuaciones consistiría en proceder como en el caso general explicado en la sección 3.4.1, de tal forma que, tomando como ejemplo la ecuación de continuidad de electrones, se realizaría una formulación variacional del problema y se aplicaría el teorema de Green-Gauss:

$$\int_{\Omega} \mu_n n \nabla(\phi_n) \nabla \theta_i \, d\Omega = \int_{\Omega} R \theta_i \, d\Omega \quad (3.124)$$

A partir de esta ecuación se realizaría una formulación en términos de las funciones de forma y se integraría sobre el elemento patrón. Pero es necesario tener en cuenta que la discretización de las ecuaciones de continuidad requiere un cuidado especial, debido al comportamiento de capa que presenta el potencial y las concentraciones de portadores. Se puede demostrar [Mar86] que, para un dispositivo unidimensional, la discretización estándar de las ecuaciones sólo conduce a soluciones aceptables si se verifica que:

$$\max_i |\psi_{i+1} - \psi_i| \ll 1 \quad (3.125)$$

donde ψ_i representa el potencial en el vértice i -ésimo de la discretización. Esta es una condición muy restrictiva, pues requiere que las zonas de transición, en las que el potencial varía de forma brusca, sean discretizadas con un tamaño de malla muy fino. Así, si un número pequeño de puntos de la malla cae dentro de esta capa, $|\psi_{i+1} - \psi_i|$ puede tomar valores muy grandes y el esquema estándar no obtiene resultados satisfactorios.

En un caso multidimensional la situación es la misma por lo que es necesario que el potencial varíe lentamente en cada elemento finito. Esto puede requerir mallas excesivamente finas con un número de vértices muy elevado, lo que hace muy costosa, en términos de memoria y tiempo de computación, y a veces imposible su realización práctica. Por tanto, es necesario utilizar otros métodos de discretización que eviten este problema.

Uno de los métodos más importantes utilizado para evitar este problema fue propuesto inicialmente por Scharfetter y Gummel [SG69] para un dispositivo unidimensional usando una aproximación en diferencias finitas, aunque

posteriormente fue aplicado a varias dimensiones y tipos de elementos no sólo usando diferencias finitas sino también elementos finitos.

Con esta aproximación es posible obtener discretizaciones apropiadas para las ecuaciones de continuidad en sistemas unidimensionales teniendo en cuenta que las densidades de corriente \mathbf{J}_n y \mathbf{J}_p son unas funciones que varían lentamente, es decir, no presentan comportamiento de capa como el potencial. De esta forma, es posible aproximar \mathbf{J}_n y \mathbf{J}_p mediante funciones con valor constante sobre cada elemento. Calculando estas funciones, y aproximando la densidad de corriente como combinación lineal de las mismas, se pueden obtener nuevas formulaciones del problema que conducen a mejores resultados. Es preciso tener en cuenta que en casos multidimensionales las densidades de corriente pueden exhibir también comportamiento de capa, pero sin embargo, su variación va a ser mucho menor que la de ψ , n o p .

El procedimiento de este método se detalla a continuación sólo para la ecuación de continuidad de electrones, puesto que la formulación sería análoga para la ecuación de continuidad de huecos.

Sea G_e el centro de gravedad de un elemento Ω_e y $\psi_\Delta(x)$ la función del potencial eléctrico definido sobre el elemento finito. Es posible obtener una aproximación de la densidad de corriente sobre el elemento Ω_e aproximando μ_n por su valor μ_{n,G_e} en el centro, y ϕ_n por funciones definidas a trozos sobre el dominio. Haciendo estas consideraciones se expresa \mathbf{J}_n sobre el elemento finito como el vector

$$\mathbf{J}_{n,G_e} = -\mu_{n,G_e} n_{ien,G_e} e^{[\psi_\Delta(x) - \phi_{n,\Delta}(x)]} \nabla \phi_{n,\Delta}(x) \quad \forall x \in \Omega_e. \quad (3.126)$$

De esta forma,

$$\mathbf{J}_n = \sum_e \mathbf{J}_{n,G_e}. \quad (3.127)$$

Pasando a integrar sobre el elemento patrón y usando $\nabla g = J_e^{-t} \widehat{\nabla} \widehat{g}$ se transforma la ecuación anterior en,

$$\mathbf{J}_{n,G_e} = -\mu_{n,G_e} n_{ien,G_e} e^{[\psi_\Delta(\widehat{x}) - \phi_{n,\Delta}(\widehat{x})]} J_e^{-t} \widehat{\nabla} \phi_{n,\Delta}(\widehat{x}) \quad \forall \widehat{x} \in \widehat{\Omega}. \quad (3.128)$$

Si se multiplican ambos términos de la ecuación anterior por $e^{-\psi_\Delta(\widehat{x})} J_e^t$, integrando la j -ésima componente con respecto a \widehat{x}_j en el intervalo $[0, 1]$ y teniendo en cuenta que $\frac{\partial x}{\partial \widehat{x}}$ es constante sobre el elemento patrón, tras invertir el cambio de variables se obtiene,

$$\mathbf{J}_{n,G_e} = \mu_{n,G_e} n_{ien,G_e} e^{\psi_\Delta(P_0)} J_e^{-t} B_T(\psi_\Delta) J_e^t \nabla e^{-\phi_{n,\Delta}(x)} \quad (3.129)$$

siendo $\{P_0, \dots, P_d\}$ los vértices del elemento Ω_e , $\psi(P_i)$ el valor del potencial en el vértice i -ésimo y $B_e(\psi)$ la matriz diagonal

$$B_T(\psi) = \begin{pmatrix} B(\psi(P_0) - \psi(P_1)) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & B(\psi(P_0) - \psi(P_d)) \end{pmatrix}, \quad (3.130)$$

donde B es la función de Bernoulli definida como [Sel84]

$$B(z) = \begin{cases} \frac{z}{(e^z - 1)} & \text{si } |z| \geq 10^{-4} \\ 1 - \frac{z}{2} \left(1 + \frac{z}{6}\right) \left(1 - \frac{z^2}{60}\right) & \text{si } |z| < 10^{-4} \end{cases} \quad (3.131)$$

Expresando la ecuación de continuidad de electrones en función del valor de la densidad de corriente en cada elemento se obtiene,

$$\sum_{\Omega_e} \nabla \mathbf{J}_{n,Ge} = R. \quad (3.132)$$

Teniendo en cuenta las condiciones de contorno, la formulación variacional de esta ecuación se puede expresar como:

$$\sum_e \int_{\Omega_e} J_{n,Ge} \nabla \xi d\Omega_e = - \sum_e \int_{\Omega_e} R \xi d\Omega_e, \quad \forall \xi \in H_0^1(\Omega \cup \partial\Omega_N). \quad (3.133)$$

Sustituyendo en la expresión anterior el valor de la densidad de corriente expresado en la fórmula (3.129) y tomando una base finita, $\{\theta_i\}_{i=1}^K$, del espacio $H_0^1(\Omega \cup \partial\Omega_N)$ se obtiene:

$$\begin{aligned} \sum_e \int_{\Omega_e} \mu_{n,Ge} n_{ien,Ge} e^{\psi_\Delta(P_0)} (\nabla e^{-\phi_{n,\Delta}(x)})^t J_e B_T^t(\psi_\Delta) J_e^{-1} \nabla \theta_i d\Omega_e = \\ = - \sum_e \int_{\Omega_e} R \theta_i d\Omega_e, \quad \forall i = 1, \dots, K. \end{aligned} \quad (3.134)$$

A continuación, se expresa el término de recombinación R como combinación lineal de las funciones de la base, sabiendo que su valor en el nodo i -ésimo es:

$$\begin{aligned} R_i = C_{BBi}(n_i p_i - n_{oi} p_{oi}) + \frac{n_i p_i - n_{oi} p_{oi}}{\tau_{p_i}(n_i + n_{Ti}) + \tau_{n_i}(p_i + p_{Ti})} + \\ + C_{Ani}(n_i^2 p_i - n_{oi}^2 p_{oi}) + C_{Ap_i}(n_i p_i^2 - n_{oi} p_{oi}^2). \end{aligned} \quad (3.135)$$

Por último, utilizando la expresión anterior junto con la relación

$$\nabla e^{-\phi_{n,\Delta}(x)} = \nabla \left(\sum_{j=1}^K e^{-\phi_{n,j}} \theta_j \right) = \sum_{j=1}^K e^{-\phi_{n,j}} \nabla \theta_j,$$

reemplazando las funciones elementales por las funciones de forma e integrando sobre el elemento patrón, es posible expresar la expresión (3.134) como:

$$\begin{aligned} \sum_e \sum_{j \in e} \mu_{n,G_e} n_{ien,G_e} e^{\psi_{\Delta}(P_0)} e^{-\phi_{n,j}} \int_{\hat{\Omega}} (\hat{\nabla} \hat{\varphi}_j)^t B_T^t(\psi_{\Delta}) J_e^{-1} J_e^{-t} \hat{\nabla} \hat{\varphi}_i |J_e| d\hat{\Omega} = \\ = - \sum_e \sum_{j \in e} R_j \int_{\hat{\Omega}} \hat{\varphi}_j \hat{\varphi}_i d\hat{\Omega}, \quad \forall i = 1, \dots, K. \end{aligned} \quad (3.136)$$

En el caso de la ecuación de continuidad de huecos se realizaría el mismo procedimiento y la expresión final obtenida sería:

$$\begin{aligned} \sum_e \sum_{j \in e} \mu_{p,G_e} n_{iep,G_e} e^{-\psi_{\Delta}(P_0)} e^{\phi_{p,j}} \int_{\hat{\Omega}} (\hat{\nabla} \hat{\varphi}_j)^t B_T^t(-\psi_{\Delta}) J_e^{-1} J_e^{-t} \hat{\nabla} \hat{\varphi}_i |J_e| d\hat{\Omega} = \\ = - \sum_e \sum_{j \in e} R_j \int_{\hat{\Omega}} \hat{\varphi}_j \hat{\varphi}_i d\hat{\Omega}, \quad \forall i = 1, \dots, K. \end{aligned} \quad (3.137)$$

3.4.4. Ecuaciones del modelo *density gradient*

Al igual que para el caso de las ecuaciones de continuidad, sólo detallamos la discretización de la ecuación correspondiente a los electrones, ya que en el caso de los huecos sería análoga. Partiendo de las ecuaciones escaladas usando los factores indicados anteriormente, empezamos mostrando el proceso en el caso de considerar una masa efectiva isotrópica. Para esto usamos la ecuación (3.54) para electrones,

$$\psi_{qn} = 2b_n \frac{\nabla^2 \sqrt{n}}{\sqrt{n}}, \quad (3.138)$$

sujeta a las condiciones de contorno

$$b_n \nabla \sqrt{n} \cdot \mathbf{n} \Big|_{\partial\Omega_N} = 0, \quad (3.139)$$

$$b_n \nabla \sqrt{n} \cdot \mathbf{n} \Big|_{\partial\Omega_R} = b_{ox} \frac{\sqrt{n}}{x_p}, \quad (3.140)$$

$$\psi_{qn} \Big|_{\partial\Omega_D} = 0, \quad (3.141)$$

donde la frontera $\partial\Omega$ del dominio de simulación Ω se divide en una parte tipo Neumann $\partial\Omega_N$, una parte tipo Dirichlet $\partial\Omega_D$ y una parte tipo Robin $\partial\Omega_R$.

Teniendo en cuenta que $\sqrt{\frac{n}{n_i}} = \sqrt{e^{\psi - \phi_n + \psi_{qn}}}$, y usando una nueva variable $u = \frac{1}{2}(\psi - \phi_n + \psi_{qn})$, podemos obtener la siguiente expresión:

$$2b_n \frac{\nabla^2 e^u}{e^u} = 2u - \psi + \phi_n. \quad (3.142)$$

Podemos aplicar el método de elemento finitos para discretizar esta ecuación. Usando el teorema de Green-Gauss, y asumiendo que v es la función de test, podemos obtener:

$$\begin{aligned} & \int_{\partial\Omega} 2b_n \frac{v}{e^u} \nabla e^u \mathbf{n} dS - \int_{\Omega} 2b_n \nabla \left(\frac{v}{e^u} \right) \nabla e^u d\Omega = \\ & = \int_{\Omega} (2u - \psi + \phi_n) v d\Omega, \quad \forall i = 1, \dots, K. \end{aligned} \quad (3.143)$$

Si desarrollamos la expresión y sustituimos las variables por sus aproximaciones en el subespacio H_0^K obtenemos

$$\begin{aligned} & \sum_{j=1}^K u_j \int_{\partial\Omega} 2b_n \theta_i \nabla \theta_j \mathbf{n} dS - \sum_{j=1}^K u_j \int_{\Omega} 2b_n (\nabla \theta_i)^t \nabla \theta_j d\Omega + \\ & + \sum_{j=1}^K \sum_{l=1}^K u_j u_l \int_{\Omega} 2b_n \theta_i (\nabla \theta_j)^t \nabla \theta_l d\Omega = \\ & = \sum_{j=1}^K (2u_j - \psi_j + \phi_{n,j}) \int_{\Omega} \theta_i \theta_j d\Omega, \\ & \quad \forall i = 1, \dots, K. \end{aligned} \quad (3.144)$$

Introduciendo las funciones base y calculando las integrales sobre el elemento patrón, obtenemos la expresión final utilizada para la resolución de la ecuación del modelo *density gradient* con masa isótropa:

$$\begin{aligned}
 & \sum_{j=1}^K u_j \sum_{s \ni i, j} \int_{\partial \hat{\Omega}} 2b_n \hat{\varphi}_i \nabla \hat{\varphi}_j \mathbf{n} |J_s| d\hat{S} - \\
 & - \sum_{j=1}^K u_j \sum_{e \ni i, j} \int_{\hat{\Omega}} 2b_n (\hat{\nabla} \hat{\varphi}_i)^t J_e^{-1} J_e^{-t} \hat{\nabla} \hat{\varphi}_j |J_e| d\hat{\Omega} + \\
 & + \sum_{j=1}^K \sum_{l=1}^K u_j u_l \sum_{e \ni i, j, l} \int_{\hat{\Omega}} 2b_n \hat{\varphi}_i (\hat{\nabla} \hat{\varphi}_j)^t J_e^{-1} J_e^{-t} \hat{\nabla} \hat{\varphi}_l |J_e| d\hat{\Omega} = \\
 & = \sum_{j=1}^K (2u_j - \psi_j + \phi_{n,j}) \sum_{e \ni i, j} \int_{\hat{\Omega}} \hat{\varphi}_i \hat{\varphi}_j |J_e| d\hat{\Omega}, \quad \forall i = 1, \dots, K. \quad (3.145)
 \end{aligned}$$

Si en lugar de considerar la masa efectiva como un escalar consideramos su carácter tensorial en el cristal, la ecuación que debemos resolver para obtener ψ_{qn} es:

$$\psi_{qn} = 2 \frac{\nabla([b_n] \nabla(\sqrt{n}))}{\sqrt{n}}, \quad (3.146)$$

donde

$$[b_n] = \frac{\hbar^2}{4qr} \begin{pmatrix} 1/m_{xx} & 0 & 0 \\ 0 & 1/m_{yy} & 0 \\ 0 & 0 & 1/m_{zz} \end{pmatrix}.$$

Si usamos la variable u obtenemos la siguiente ecuación para discretizar:

$$2 \frac{\nabla([b_n] \nabla e^u)}{e^u} = 2u - \psi + \phi_n. \quad (3.147)$$

Una vez simplificada, la formulación variacional de esta ecuación es:

$$\begin{aligned}
 & \int_{\partial \Omega} 2v(\nabla u)^t [b_n]^t \mathbf{n} dS - \int_{\Omega} 2(\nabla v)^t [b_n] \nabla u d\Omega + \\
 & + \int_{\Omega} 2v(\nabla u)^t [b_n] \nabla u d\Omega = \int_{\Omega} (2u - \psi + \phi_n)v d\Omega, \quad (3.148)
 \end{aligned}$$

donde v es la función de test y \mathbf{n} un vector normal a la superficie $\partial \Omega$.

Empleando las funciones de forma sobre los elementos e integrando posteriormente sobre el elemento patrón podemos reescribir las ecuaciones en su forma final:

$$\begin{aligned}
& 2 \sum_{j=1}^K u_j \sum_{s \ni i, j} \int_{\partial \hat{\Omega}} \hat{\varphi}_i (\nabla \hat{\varphi}_j)^t [b_n]^t \mathbf{n} |j_s| d\hat{S} - \\
& - 2 \sum_{j=1}^K u_j \sum_{e \ni i, j} \int_{\hat{\Omega}} (\hat{\nabla} \hat{\varphi}_i)^t J_e^{-1} [b_n] J_e^{-t} \hat{\nabla} \hat{\varphi}_j |J_e| d\hat{\Omega} + \\
& + 2 \sum_{j=1}^K \sum_{l=1}^K u_j u_l \sum_{e \ni i, j, l} \int_{\hat{\Omega}} \hat{\varphi}_i (\hat{\nabla} \hat{\varphi}_j)^t J_e^{-1} [b_n] J_e^{-t} \hat{\nabla} \hat{\varphi}_l |J_e| d\hat{\Omega} = \\
& = \sum_{j=1}^K (2u_j - \psi_j + \phi_{n,j}) \sum_{e \ni i, j} \int_{\hat{\Omega}} \hat{\varphi}_i \hat{\varphi}_j |J_e| d\hat{\Omega}, \quad \forall i = 1, \dots, K. \quad (3.149)
\end{aligned}$$

Un proceso similar con la ecuación del *density gradient* para huecos nos llevaría a la siguiente forma final de la ecuación discretizada:

$$\begin{aligned}
& 2 \sum_{j=1}^K w_j \sum_{s \ni i, j} \int_{\partial \hat{\Omega}} \hat{\varphi}_i (\nabla \hat{\varphi}_j)^t [b_p]^t \mathbf{n} |j_s| d\hat{S} - \\
& - 2 \sum_{j=1}^K w_j \sum_{e \ni i, j} \int_{\hat{\Omega}} (\hat{\nabla} \hat{\varphi}_i)^t J_e^{-1} [b_p] J_e^{-t} \hat{\nabla} \hat{\varphi}_j |J_e| d\hat{\Omega} + \\
& + 2 \sum_{j=1}^K \sum_{l=1}^K w_j w_l \sum_{e \ni i, j, l} \int_{\hat{\Omega}} \hat{\varphi}_i (\hat{\nabla} \hat{\varphi}_j)^t J_e^{-1} [b_p] J_e^{-t} \hat{\nabla} \hat{\varphi}_l |J_e| d\hat{\Omega} = \\
& = \sum_{j=1}^K (2w_j + \psi_j - \phi_{p,j}) \sum_{e \ni i, j} \int_{\hat{\Omega}} \hat{\varphi}_i \hat{\varphi}_j |J_e| d\hat{\Omega}, \quad \forall i = 1, \dots, K, \quad (3.150)
\end{aligned}$$

donde hemos usado la definición $w = \frac{1}{2}(\phi_p + \psi_{qp} - \psi)$.

3.5. Resolución paralela de sistemas de ecuaciones

Las ecuaciones discretizadas que hemos obtenido están acopladas y son no lineales. Esto hace que su resolución directa sea un problema muy complejo, especialmente al tratarse de simulaciones tridimensionales que requieren

un número de nodos muy elevado. En esta sección describiremos brevementes los pasos necesarios para la resolución de este sistema de ecuaciones, que finalmente se reducirá a la resolución de tres sistemas lineales. Pese a esta simplificación, este es un problema costoso y el uso de métodos de resolución paralelos resulta de gran utilidad y en ocasiones necesario para poder satisfacer los requerimientos computacionales del mismo, especialmente al abordar problemas como el estudio de fluctuaciones de parámetros intrínsecos en dispositivos semiconductores.

3.5.1. Ecuaciones del sistema discretizado

En los dispositivos MOSFET de canal n que estudiaremos a continuación, los portadores mayoritarios son electrones, por lo que lejos de la zona de ruptura es posible ignorar, tal y como se hará a partir de este momento, la contribución de la ecuación de continuidad de huecos y los términos de generación y recombinación. De esta forma, en el modelo de arrastre-difusión sólo será necesario resolver las ecuaciones de Poisson, de continuidad de electrones y *density gradient* para electrones, acelerando así el proceso de simulación.

Por lo tanto, especificando para el caso concreto de estos dispositivos, las ecuaciones del modelo de arrastre-difusión a resolver son (3.116), (3.149) y (3.136), considerando en esta última ecuación el término de recombinación nulo. Estas ecuaciones constituyen un sistema acoplado de ecuaciones diferenciales no lineales en derivadas parciales que definen el comportamiento del transistor. Si el dispositivo está discretizado en K nodos, es necesario resolver un sistema de $3K$ ecuaciones con $3K$ incógnitas $\Psi = (\psi, \psi_{qn}, \phi_n)$ de la forma:

$$F(\Psi) = \begin{pmatrix} F_{\psi}(\psi, \psi_{qn}, \phi_n) \\ F_{\psi_{qn}}(\psi, \psi_{qn}, \phi_n) \\ F_{\phi_n}(\psi, \psi_{qn}, \phi_n) \end{pmatrix} = 0 \quad (3.151)$$

donde $F_{\psi}(\psi, \psi_{qn}, \phi_n)$, $F_{\psi_{qn}}(\psi, \psi_{qn}, \phi_n)$ y $F_{\phi_n}(\psi, \psi_{qn}, \phi_n)$ representan las ecuaciones de Poisson, de *density gradient* para electrones y de continuidad de electrones respectivamente.

Para la resolución del sistema (3.151) se pueden utilizar métodos acoplados o desacoplados. Los primeros son métodos tipo Newton [RPD85], en los que se resuelve el sistema completo formado por las $3K$ ecuaciones directamente, actualizando a la vez los valores de ψ , ψ_{qn} y ϕ_n . Sin embargo, si el número K de variables es muy elevado (como ocurre en la simulación 3D), esto puede

ser muy costoso tanto en tiempo de computación como en consumo de memoria. Los métodos desacoplados se basan en el método de Gummel [Gum64] y resuelven las tres ecuaciones secuencialmente, siendo preciso resolver tres sistemas de ecuaciones de dimensión K . De esta forma se obtienen de forma consecutiva actualizaciones para ψ , ψ_{qn} y ϕ_n . Este método se describe en mayor detalle en la siguiente sección. En cualquier caso, usemos métodos acoplados o desacoplados, al final es necesaria la resolución de sistemas no lineales. Probablemente el método más extendido es el de Newton-Raphson, un método iterativo que resuelve el sistema no lineal mediante la resolución de una secuencia de sistemas lineales. Esto permite la utilización de los métodos de resolución altamente eficientes disponibles hoy en día para resolución de sistemas lineales. Describiremos este método en detalle en la sección 3.5.3.

3.5.2. Método de Gummel

El método de Gummel [Gum64] ha sido aplicado ampliamente a la resolución de las ecuaciones básicas de los semiconductores. Matemáticamente, corresponde a un algoritmo iterativo por bloques no lineal tipo Gauss-Seidel [BBC⁺94]. En el caso particular de los dispositivos n-MOSFET, usando este método se desacoplan las ecuaciones de Poisson, *density gradient* para electrones y continuidad de electrones y se resuelve cada una de ellas por separado.

El método de Gummel en este caso realizaría dos iteraciones, como se muestra en el algoritmo 3.1. Utilizaremos los índices k, l para referirnos a ellas. El índice l hace referencia a las iteraciones del lazo Gummel más interno, que resuelve el sistema Poisson-*density gradient* con el pseudopotencial de electrones ϕ_n fijo. Las iteraciones del lazo más externo, que incluyen la resolución de la ecuación de continuidad, se indican con el índice k . Partimos de la aproximación inicial dada por $(\psi^{k,l}, \psi_{qn}^{k,l}, \phi_n^k)$ para $k, l = 0$. En primer lugar hacemos una iteración sobre l para obtener los valores ψ^{k+1} y ψ_{qn}^{k+1} que se necesitan para resolver la ecuación de continuidad. $\psi^{k,l+1}$ se computa resolviendo la ecuación de Poisson, con $\psi_{qn}^{k,l}$ y ϕ_n^{k-1} constantes. El valor de $\psi_{qn}^{k,l+1}$ se obtiene a continuación, resolviendo la ecuación de *density gradient* para electrones con $\psi^{k,l+1}$ y ϕ_n^{k-1} constantes. Esta iteración en l continúa hasta que en la iteración i -ésima se alcanza la convergencia, $\|\psi^{k,i} - \psi^{k,i-1}\| < \varepsilon_\psi$ y $\|\psi_{qn}^{k,i} - \psi_{qn}^{k,i-1}\| < \varepsilon_{\psi_{qn}}$, o se sobrepase un número dado de iteraciones máximo M_l . En este momento tendremos los valores de $\psi^k = \psi^{k,i}$ y $\psi_{qn}^k = \psi_{qn}^{k,i}$. El valor de ϕ_n^k se obtiene a continuación, resolviendo la ecuación de continuidad de electrones con ψ^k y ψ_{qn}^k constantes.

Algoritmo 3.1: Iteración Gummel con *density gradient*.

Entrada: Aproximaciones iniciales a los potenciales $(\psi^0, \psi_{qn}^0, \phi_n^0)$
Salida: Valor de los potenciales $(\psi, \psi_{qn}, \phi_n)$
Dados: $\varepsilon_\psi, \varepsilon_{\psi_{qn}}, \varepsilon_{\phi_n}, M_k, M_l$

- 1 $k = 1$
- 2 **mientras** $\|\Delta_k(\psi^k, \psi_{qn}^k, \phi_n^k)\| > (\varepsilon_\psi, \varepsilon_{\psi_{qn}}, \varepsilon_{\phi_n})$ y $k < M_k$ **hacer**
- 3 $l = 1$
- 4 $\psi^{k,0} := \psi^{k-1}; \psi_{qn}^{k,0} := \psi_{qn}^{k-1}$
- 5 **mientras** $\|\Delta_l(\psi^{k,l}, \psi_{qn}^{k,l})\| > (\varepsilon_\psi, \varepsilon_{\psi_{qn}})$ y $l < M_l$ **hacer**
- 6 Resolver Poisson para $\psi^{k,l}$
- 7 Resolver *density gradient* para $\psi_{qn}^{k,l}$
- 8 Actualizar n
- 9 $l = l + 1$
- 10 **fin**
- 11 $\psi^k := \psi^{k,l-1}; \psi_{qn}^k := \psi_{qn}^{k,l-1}$
- 12 Resolver continuidad de electrones para ϕ_n^k
- 13 Actualizar n
- 14 $k = k + 1;$
- 15 **fin**

Este método se ha mostrado muy útil en la práctica. La convergencia puede alcanzarse incluso empezando con malas condiciones iniciales, y puede ser muy rápida en muchas ocasiones. Por otro lado, en algunas aplicaciones, como por ejemplo en situaciones de muy alto nivel de inyección en el dispositivo semiconductor y en los casos en los que el factor de generación-recombinación tiene un peso elevado, el método podría tener problemas de convergencia. Sin embargo, en situaciones cercanas al equilibrio térmico y con $R = 0$, numerosos autores avalan la utilidad de este método, ya que la ecuación de continuidad de electrones puede considerarse como una pequeña perturbación de la ecuación del potencial.

Existen diferentes trabajos destinados a mejorar la convergencia del método de Gummel, pero evitando tener que resolver el sistema acoplado. En algunas ocasiones el método de Gummel converge rápidamente en las primeras iteraciones, pero luego la convergencia se hace más lenta. En este caso se pueden combinar los métodos de Gummel y el método acoplado de Newton. De

esta forma primero se aproxima la solución haciendo uso del método de Gummel y luego se cambia al método de Newton para aprovechar su propiedad de convergencia cuadrática cerca de la solución. También existen otras aproximaciones distintas basadas en combinar los métodos de Gummel y Newton usando otras estrategias [RS88, WLWD89].

3.5.3. Método de Newton-Raphson

El método de Newton es un método iterativo que se aplica a la resolución de sistemas no lineales.

Dada una función $F : \mathbb{R}^N \rightarrow \mathbb{R}^N$ se busca un vector $x^* \in \mathbb{R}^N$ tal que $F(x^*) = 0$. Para resolver el sistema se parte de un vector solución inicial $x^0 \in \mathbb{R}^N$ y se busca la serie x^k , tal que $\lim_{k \rightarrow \infty} x^k = x^*$. Los elementos de dicha serie se obtienen mediante la siguiente iteración:

$$F'(x^k)\Delta x^k = -F(x^k) \quad (3.152)$$

$$x^{k+1} = x^k + \Delta x^k, \quad (3.153)$$

la cual se repite hasta que se cumple alguno de los criterios de parada, siendo $F'(x)$ la matriz jacobiana del sistema, que se obtiene como:

$$F'_{ij}(x) = \frac{\partial F_i}{\partial x_j} \quad (3.154)$$

La importancia del método de Newton se debe a que, con ciertas condiciones naturales de F , la iteración presenta una convergencia cuadrática

$$\|x^{k+1} - x^*\| \leq c \cdot \|x^k - x^*\|^2 \quad (3.155)$$

siempre que el vector inicial x^0 esté suficientemente próximo a la solución buscada. Esto indica que el $(k+1)$ -ésimo error cometido es proporcional al cuadrado del k -ésimo error, por lo que la convergencia es muy rápida, siempre que los errores sean pequeños.

La aplicación del método de Newton a las ecuaciones de los semiconductores implica resolver el siguiente sistema en cada iteración:

$$F'(\Psi^k)\Delta\Psi^k = -F(\Psi^k) \quad (3.156)$$

y actualizar las variables haciendo $\Psi^{k+1} = \Psi^k + \Delta\Psi^k$.

Suele ser necesario mejorar la convergencia del método de Newton, para lo que se utiliza una versión modificada que emplea un parámetro de amortiguamiento o sobrerrelajación, t_k , en la actualización de Ψ , es decir:

$$\Psi^{k+1} = \Psi^k + t_k \cdot \Delta \Psi^k \quad (3.157)$$

siendo $0 < t_k \leq 1$. El parámetro de amortiguamiento t_k se suele ajustar a partir de resultados experimentales, aunque también existen métodos teóricos para obtenerlo [BR81, BR80]. En el caso de que la matriz de coeficientes esté muy mal condicionada se puede utilizar otra versión que suma al jacobiano el término $s_k I$, donde $s_k \in \mathbb{R}^+$ e I es la matriz identidad, de modo que se mejoran las propiedades del sistema.

3.5.4. Resolución de sistemas lineales

Podemos agrupar los métodos utilizados en la resolución de los sistemas lineales en dos categorías: métodos directos, si están basados en la factorización de la matriz dispersa, y métodos iterativos, si tratan de encontrar la solución al sistema por medio de aproximaciones sucesivas tomando como partida una solución inicial. En nuestro caso hemos optado por el uso de métodos iterativos, concretamente técnicas basadas en descomposición de dominios [Saa96]. Estas técnicas son altamente eficientes en computación paralela, lo que las hace idóneas para su uso en el simulador.

Los sistemas de ecuaciones lineales se pueden expresar como $Ax = b$, donde dada una matriz A de orden $n \times n$ y un vector b n -dimensional, se trata de determinar el vector solución x de dimensión n . Además, en nuestro caso, la matriz A es dispersa, por lo que podemos aprovechar las ventajas de métodos que utilizan esta propiedad.

En particular, los sistemas de ecuaciones lineales que obtenemos de las ecuaciones de Poisson, *density gradient* y continuidad son resueltas usando una técnica de descomposición de dominios [BBC⁺94]. De esta manera, el dominio del problema Ω se particiona en p subdominios Ω_i de tal forma que

$$\Omega = \bigcup_{i=1}^p \Omega_i. \quad (3.158)$$

Los métodos de descomposición de dominios tratan de resolver el problema en el dominio Ω mediante la resolución concurrente en cada subdominio Ω_i .

Cada nodo perteneciente a un subdominio es una incógnita del problema. Es importante distinguir entre tres tipos de nodos o incógnitas:

- **Nodos internos:** nodos locales que sólo tienen relación con otros nodos interiores al subdominio y por tanto sin ningún contacto con nodos pertenecientes a otros subdominios.
- **Nodos frontera internos:** nodos locales pero acoplados con nodos frontera pertenecientes a otros subdominios.
- **Nodos frontera externos:** nodos pertenecientes a otros subdominios pero acoplados con nodos frontera internos de este subdominio.

Etiquetamos los nodos de acuerdo a su categoría, primero los nodos internos y después los nodos frontera internos y externos. Como resultado, el sistema lineal asociado con el problema tiene la siguiente estructura:

$$\begin{pmatrix} B_1 & & & E_1 \\ & B_2 & & E_2 \\ & & \ddots & \\ & & & B_p & E_p \\ F_1 & F_2 & & F_p & C \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_s \\ y \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_s \\ g \end{pmatrix}, \quad (3.159)$$

donde para el subdominio Ω_i , el subvector y indica todas las incógnitas pertenecientes a la interfaz entre los subdominios, las submatrices B_i indican el acoplamiento entre los nodos internos, las E_i el acoplamiento de las ecuaciones de cada subdominio con los nodos frontera, las F_i el acoplamiento de los nodos frontera de cada subdominio con los nodos frontera indicados por y y, por último, la matriz C indica el acoplamiento de los nodos frontera entre sí.

Para particionar la malla en subdominios utilizamos el programa METIS [KK97], que se utiliza también para reordenar los nodos de cada subdominio para obtener matrices con mejores propiedades numéricas. Para resolver los sistemas lineales locales utilizamos la librería PSPARSLIB [SLK97] modificada para aprovechar mejor el reordenamiento de las matrices [SGLA08]. Entre las distintas técnicas de descomposición de dominios soportadas por la librería los mejores resultados se obtuvieron utilizando el método de Schwarz aditivo [SGL06]. Este método es similar a una iteración de Jacobi por bloques y consiste en actualizar todas las nuevas componentes a partir del mismo residuo. La iteración básica de este método es la siguiente:

- 1.- Obtener $y_{i,ext}$
- 2.- Calcular el residuo local $r_i = (b - Ax)_i$
- 3.- Resolver $A_i \delta_i = r_i$

4.– Actualizar la solución $x_i = x_i + \delta_i$

El sistema lineal $A_i \delta_i = r_i$ se resuelve utilizando el método FGMRES pre-condicionado por una factorización LU incompleta con umbral (ILUT) [Saa96].

3.6. Simulación de fluctuaciones en dispositivos MOS-FET de silicio

Con la reducción del tamaño de los transistores a escalas del orden de los nanómetros, el efecto de las desigualdades microscópicas empieza a jugar un papel determinante en su funcionamiento. Por ejemplo, la diferencia en las posiciones y número de dopantes en los transistores es actualmente considerada como una de las fuentes más importantes de desapareamiento entre transistores [RBAL⁺06]. Además del dopado, otros parámetros considerados habitualmente continuos o uniformes dejan de poder ser descritos de esta manera si se quiere una imagen realista del dispositivo. Algunos de los más importantes incluyen la interfaz del canal con el óxido de puerta, que es presenta una rugosidad que varía entre transistores, o los bordes de la puerta, que no son rectos y paralelos por limitaciones en las técnicas litográficas [RBAL⁺06]. También las puertas de polisilicio y la introducción de dieléctricos de alta permitividad (high- κ) en los MOSFETs actuales son otras fuentes de fluctuaciones de parámetros intrínsecos que se suman a los dopantes aleatorios en canal, fuente y drenador, la rugosidad en la interfaz y en la litografía de la puerta [AS00, TMSS97, SVSP96].

3.6.1. Impacto de la granularidad del polisilicio

La estructura granular del polisilicio es una de las principales fuentes de fluctuaciones que llevan al desapareamiento del voltaje umbral en MOSFETs de canal n escalados a dimensiones por debajo de los 100 nm [ACC⁺08]. El aumento de la difusión y la segregación a lo largo de las fronteras entre cristales causa un dopado no uniforme en el polisilicio y una penetración local de dopantes en la región del canal. Además, la existencia de estados en la interfaz de tipo aceptor produce una acumulación de carga en la interfaz y un pico en el nivel de Fermi a su alrededor [YHH99]. Todos estos factores dependen de las condiciones concretas de fabricación.

Para reflejar estos estados en la interfaz introducimos una carga superficial en las fronteras entre cristales del polisilicio. En primer lugar vamos a analizar

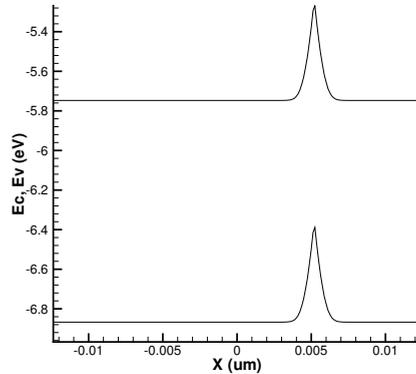


Figura 3.4: Bandas de conducción y valencia del polisilicio en una línea que corta perpendicularmente el plano de la frontera entre cristales.

el impacto de la posición de una única frontera en la puerta. El número de fronteras en un dispositivo real dependerá del tamaño característico de cada cristal de polisilicio, lo que depende del proceso tecnológico concreto.

Para este estudio simulamos dos transistores n-MOSFET de 80 y 25 nm de longitud de puerta física respectivamente. En ambos casos, introducimos una carga en la frontera de tal forma que las bandas de conducción y valencia tengan un pico de magnitud cercana a la mitad del gap. En la figura 3.4 mostramos el diagrama de bandas a lo largo de una línea en la puerta que cruza perpendicularmente la frontera de polisilicio.

En la figura 3.5 mostramos, para el dispositivo de 80 nm de longitud de puerta, el desplazamiento de la tensión umbral asociado con el movimiento de una frontera perpendicular a la dirección del flujo de corriente desde el extremo de fuente al de drenador. El desplazamiento se calcula con respecto al dispositivo sin ninguna frontera en la puerta. Podemos ver que la frontera tiene el mayor impacto cuando está alineada con el máximo de la barrera de potencial entre fuente y drenador. Cuando el voltaje de drenador es bajo el máximo está próximo al centro del canal, pero se mueve hacia la fuente al aumentar el voltaje de drenador.

En el caso de la simulación del transistor de 25 nm de longitud de puerta, como se muestra en la figura 3.6, podemos ver un comportamiento semejante. Sin embargo, en este caso la magnitud es diferente. Los valores máximos del desplazamiento de la tensión umbral a voltaje de drenador bajo son mayores en

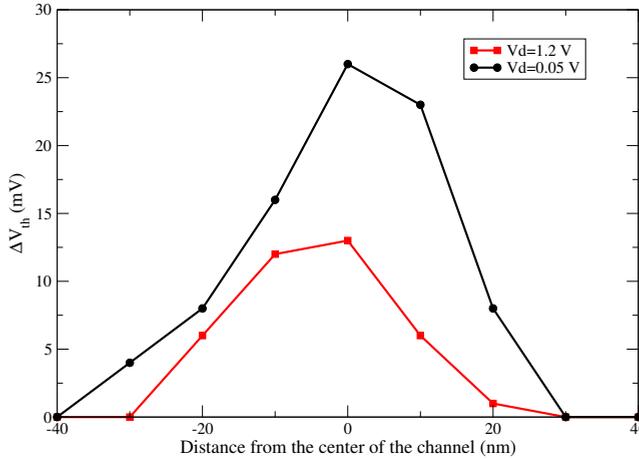


Figura 3.5: Desplazamiento del voltaje umbral en el dispositivo de 80 nm de longitud de puerta producido por una frontera moviéndose desde el extremo de fuente (valores negativos) al de drenador (valores positivos).

un 75 % aproximadamente y cerca de un 400 % en el caso en que polarizamos con una tensión de drenador alta. Como era de esperar, cuando la frontera en el polisilicio está cerca de la fuente o del drenador el desplazamiento desaparece en ambos dispositivos. En la figura 3.7 se muestra la distribución de potencial electrostático en el MOSFET de 25 nm de longitud de puerta justo debajo de la interfaz óxido-semiconductor con una polarización de $V_D = 0,05$ V y $V_G=1,1$ V y la frontera en la posición $x = -2,5$ nm. La disminución del potencial y por tanto aumento de la barrera de energía se puede ver claramente.

3.6.2. Impacto de algunas no idealidades de óxidos de alta permitividad

Uno de los problemas de los óxidos de alta permitividad es que en el proceso de fabricación puede obtenerse una fase policristalina. En este caso, cristales con diferentes orientaciones pueden presentar diferentes constantes dieléctricas. En esta sección presentamos simulaciones del impacto de la presencia de dos cristales con diferente constante dieléctrica en el óxido de puerta para el MOSFET de 25 nm. En primer lugar analizamos la influencia de la posición de la frontera suponiendo que es perpendicular a la dirección fuente-drenador. Después, estudiamos el impacto de la orientación del cristal.

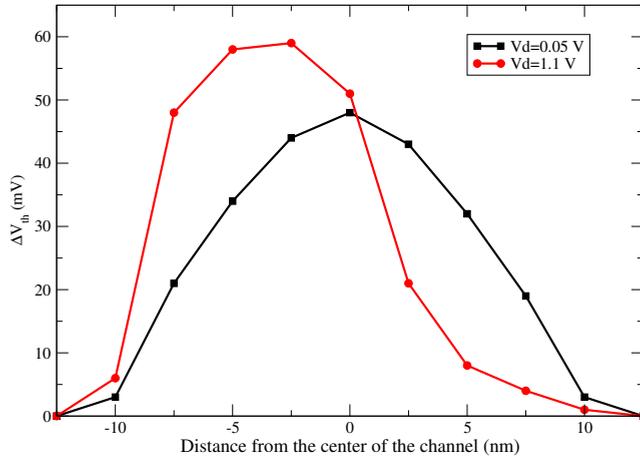


Figura 3.6: Desplazamiento del voltaje umbral en el dispositivo de 25 nm de longitud de puerta producido por una frontera moviéndose desde el extremo de fuente (valores negativos) al de drenador (valores positivos).

Para realizar el estudio consideraremos que uno de los cristales tiene constante dieléctrica 7 y el otro constante dieléctrica 5. La figura 3.8 muestra el desplazamiento como función de la posición de la frontera, perpendicular a la dirección del flujo de corriente. Podemos ver que el impacto es mayor a baja polarización de drenador. Vemos que la región en la que afecta esta discontinuidad en la constante dieléctrica (90 % de la variación) es de aproximadamente 18 nm. Al aumentar la polarización de drenador vemos que esta región disminuye notablemente, pasando a tener un ancho de unos 13 nm.

Finalmente, en la figura 3.9 mostramos la dependencia del desplazamiento del voltaje umbral con el ángulo que forma la frontera con la dirección del flujo de corriente. Cuando el voltaje de drenador es bajo, encontramos los mayores valores del desplazamiento para fronteras perpendiculares al flujo de corriente. Estos valores disminuyen cuando la frontera se aproxima a ángulos de 0 y 180 grados, aunque el mínimo cercano al ángulo cero se alcanza aproximadamente en 15 grados. En el caso de polarizar el transistor con un voltaje de drenador elevado el comportamiento es muy diferente. En este caso el parámetro más importante es la constante dieléctrica en la mitad de fuente de la puerta, de tal forma que el desplazamiento es casi nulo para un ángulo de unos 90 grados. De esta forma, la oscilación resultante tiene un período doble que en el caso de la polarización de drenador baja.

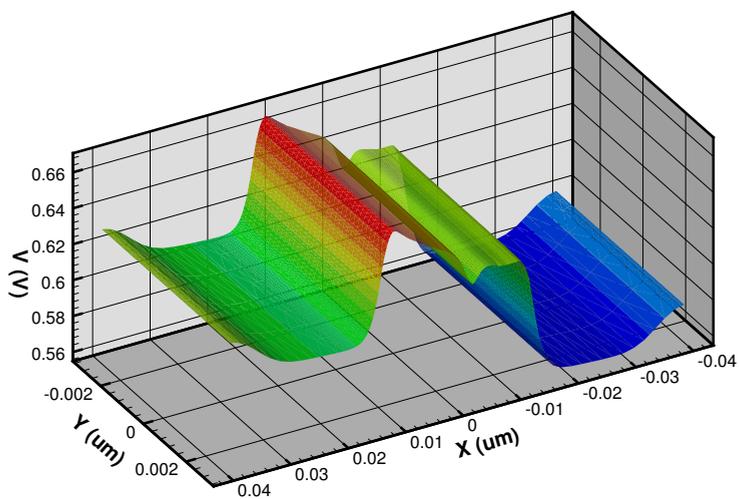


Figura 3.7: Potencial electrostático bajo la capa de dieléctrico cuando hay una frontera entre cristales de polisilicio en el plano $x = -2,5$ nm.

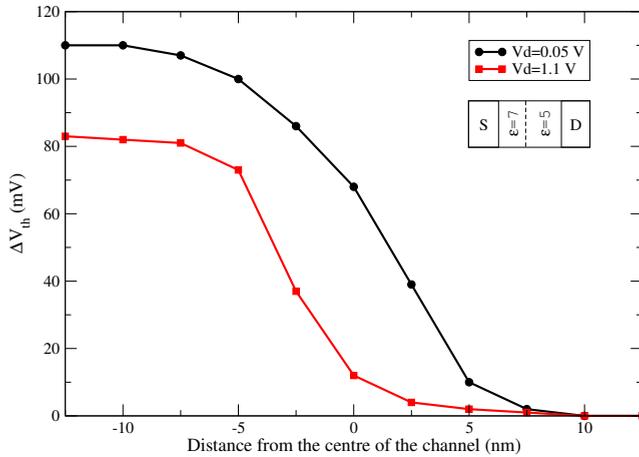


Figura 3.8: Desplazamiento del voltaje umbral en el MOSFET de 25 nm de longitud de puerta producido por una frontera en el dieléctrico moviéndose del extremo de fuente (valores negativos) al de drenador (valores positivos). Los valores del desplazamiento son relativos al de un dispositivo con Si_3N_4 como dieléctrico de puerta.

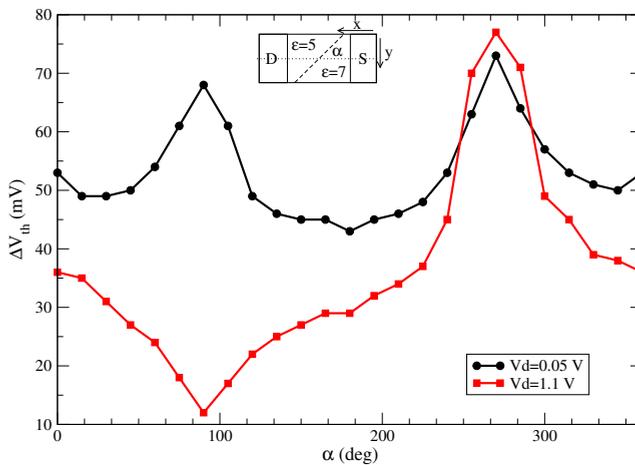


Figura 3.9: Dependencia del desplazamiento de la tensión umbral con el ángulo de la frontera entre los dos cristales y el eje x en el dispositivo de 25 nm de longitud de puerta.

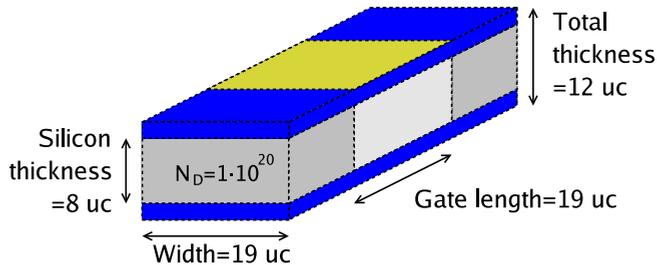


Figura 3.10: Geometría del MOSFET de doble puerta simulado. Las dimensiones están dadas en unidades de la constante de red del silicio ($1 \text{ uc} = 0,543 \text{ nm}$).

3.6.3. Impacto de las posiciones aleatorias de los dopantes

Como primer ejemplo de uso de las mallas atómicas presentamos un estudio del impacto de la posición aleatoria de los dopantes sobre las curvas características I_D-V_G de un MOSFET de doble puerta con longitud de canal de 10 nm (figura 3.10).

La figura 2.39 muestra una malla del cuerpo de silicio del transistor construida utilizando el algoritmo descrito en la sección 2.4. Para generar los dispositivos se colocan dopantes aleatoriamente en los nodos de la malla con una probabilidad dependiente del dopado local en el nodo [Ase98]. Cada una de estas configuraciones produce un perfil de potencial diferente, lo que resulta en una curva I_D-V_G diferente. Un ejemplo del potencial electrostático para una configuración concreta se muestra en la figura 3.11.

En primer lugar comparamos los resultados obtenidos con la malla atómica y con una malla no atómica fina. Los resultados de las simulaciones del dispositivo con dopado continuo se muestran en la figura 3.12. Este resultado confirma que la malla atómica captura de forma apropiada las variaciones en las variables físicas. Esta misma figura también muestra las curvas I_D-V_G simuladas del conjunto de MOSFETs de doble puerta a voltaje de drenador bajo de $0,05 \text{ V}$. Se puede ver una dispersión apreciable en las curvas, reflejada en la desviación estándar del voltaje umbral, $\sigma V_T = 0,005 \text{ V}$. Además, también investigamos el impacto de los dopantes en las corrientes. La figura 3.13 muestra la desviación estándar de la corriente de drenador como función del voltaje de puerta para $V_D = 0,05 \text{ V}$. El comportamiento exponencial de la corriente como función del voltaje de puerta en la zona subumbral resulta en un gran impacto de los dopantes aleatorios, con una variación superior al 35 %.

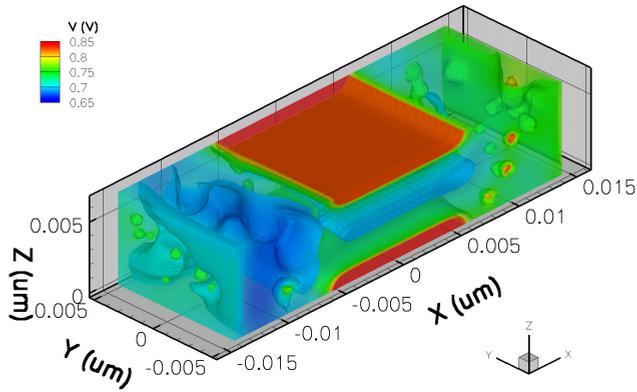


Figura 3.11: Potencial electrostático para una configuración aleatoria de dopantes para un voltaje de drenador de 0,05 V y de puerta de 1,1 V.

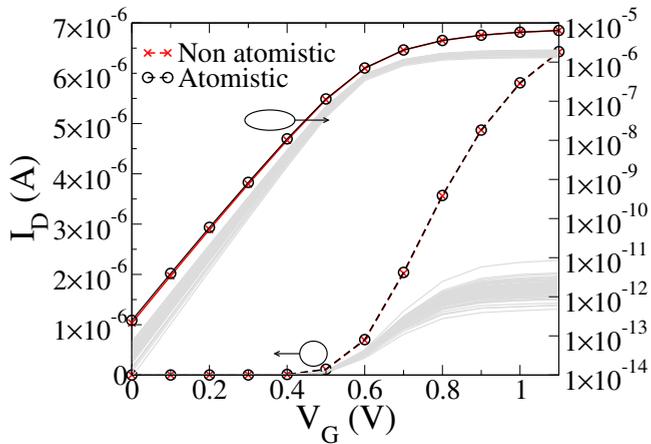


Figura 3.12: Curvas características I_D - V_G para un voltaje de drenador de 0,05 para un grupo de transistores simulados. Las corrientes de drenador usando dopado continuo con las mallas atomística (círculos) y no atomística (cruces) se muestran por comparación.

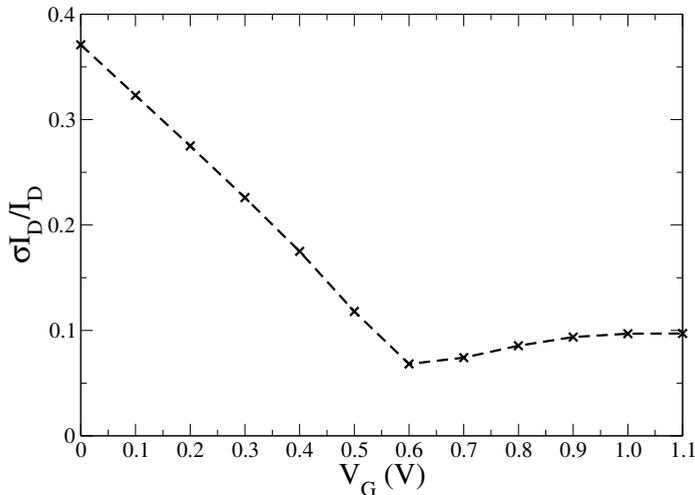


Figura 3.13: Desviación estándar normalizada de la corriente de drenador como función de la tensión de puerta para una polarización de drenador de 0,05 V.

Estas fluctuaciones disminuyen al aumentar la polarización de puerta por encima de la tensión umbral, llegando a $\sigma I_D \simeq 10\%$, cuando la corriente está en la región óhmica y satura. Debido a los grandes valores que se obtienen en la zona subumbral es habitual expresar la dispersión en la corriente a partir de la distribución de $\log(I_D)$, lo que da la variación de la corriente en órdenes de magnitud. En este caso, la desviación estándar que obtenemos es de $\sigma \log(I_D) \simeq 0,17$. De forma similar, la diferencia relativa entre el valor medio de la corriente obtenido del conjunto de dispositivos con fuente y drenador descritos de forma atomística y la corriente obtenida de la simulación usando un dopado continuo disminuye al aumentar el voltaje de puerta como se puede ver en la figura 3.14.

La gran diferencia entre la corriente continua y la media en las simulaciones atomísticas, especialmente a alto voltaje de puerta, se debe a la localización de la carga en las posiciones de los dopantes, donde aparecen pozos coulombianos asociados con los dopantes individuales. Esto es un problema intrínseco a las simulaciones clásicas que asumen estadísticas de Boltzmann o Fermi-Dirac y por tanto pueblan fuertemente los pozos. Sin embargo, el estado fundamental de los pozos está relativamente próximo al eje de la banda de conducción, por lo que esta acumulación inherente al modelo clásico no es realista. Este problema del modelo arrastre-difusión puede ser solucionado mediante la inclusión de

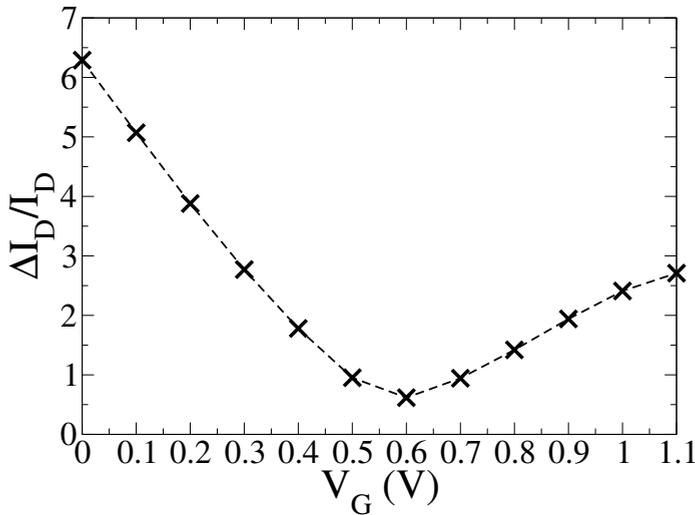


Figura 3.14: Variación relativa de la corriente de drenador como función de la tensión de puerta para una tensión de drenador de 0,05 V.

correcciones cuánticas [RBAL⁺06], aunque la convergencia de las ecuaciones del modelo *density gradient* requiere cierto cuidado. Esto se debe principalmente a estos pozos de potencial, que producen valores de los gradiente muy elevados en zonas muy localizadas y hacen que las ecuaciones resulten difíciles de resolver mediante métodos iterativos si no tenemos una buena solución inicial. Esta solución se puede conseguir, por ejemplo, resolviendo un problema similar en el que los dopantes tienen únicamente una fracción de su carga.

Por último, comparamos las simulaciones de dispositivos con dopado atómico usando las mallas atómica y no atómica. Una vez simulado el conjunto de dispositivos con la malla atómica seleccionamos las configuraciones de dopantes que producen mayor y menor corriente. Estos dispositivos se simulan entonces usando dos esquemas de asignación de carga sobre la malla no atómica: asignando la carga del dopante al nodo más próximo (*Nearest Grid Point, NGP*) y (ii) repartiendo la carga entre los nodos vecinos (*Cloud-in-Cell, CIC*) [HE88]. Los resultados de estas simulaciones se muestran en la figura 3.15. En primer lugar notamos una mejora en la convergencia cuando usamos la malla atómica. La configuración de dopantes que produce la menor corriente no converge cuando se usa la malla no atómica con el esquema de asignación de carga NGP. Sin embargo, las simulaciones con la otra configuración de dopantes producen resultados muy similares cuando se usan la

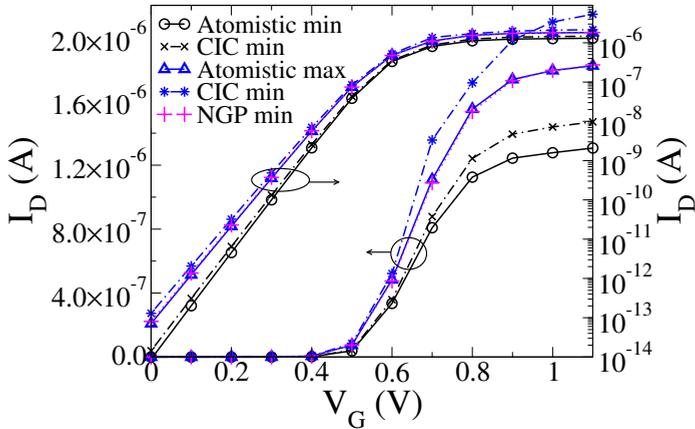


Figura 3.15: Curvas características I_D - V_G de los dispositivos con las corrientes mayor (max) y menor (min) usando la malla atomística y la malla uniforme con dos métodos de asignación de carga, CIC y NGP.

malla atomística y la no atomística con asignación de carga NGP. Esto era de esperar, ya que la malla no atomística es lo suficientemente fina como para que la correlación entre las posiciones de los dopantes introducida por la asignación de carga sea despreciable. La discrepancia con la malla no atomística usando el esquema CIC es una consecuencia del perfil de potencial diferente en las proximidades de las impurezas ionizadas originado por la repartición de la carga [RBAL⁺06], lo que hace complicada una comparación directa.

3.6.4. Impacto de las regiones de transición entre materiales

Como segundo ejemplo de uso de las mallas atomísticas estudiamos el impacto de la región de transición entre Si y SiO₂ en las características de transporte de un MOSFET de doble puerta de 6 nm de longitud de canal con espesor de óxido de 8 Å. El cambio de la estructura atómica en la interfaz Si/SiO₂ lleva a la existencia de estados electrónicos cercanos a la banda de conducción de Si, produciendo una transición gradual del ancho de la banda prohibida y de la permitividad dentro del dióxido de silicio [KYU⁺99, GP05]. Aunque el impacto de esta región de transición en las características de capacitores MOS ya se ha estudiado previamente [Ste77, WMM06, MSR⁺08], el trabajo sobre el impacto en la corriente de transistores MOSFET es más escaso.

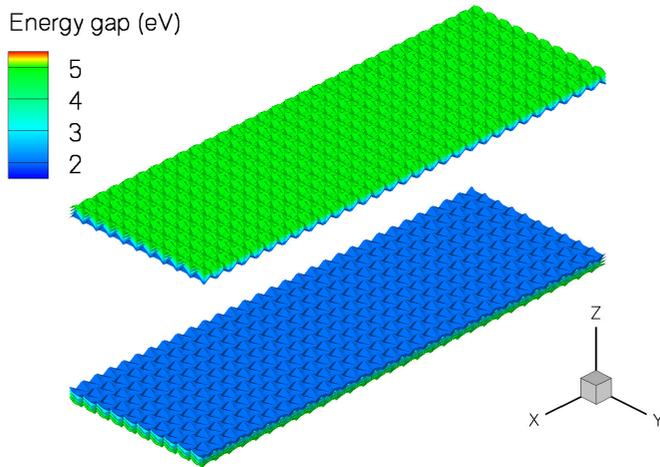


Figura 3.16: Isosuperficies del ancho de la banda prohibida en la región de transición del SiO_2 .

Para realizar las simulaciones, el ancho de la banda prohibida se obtiene a partir de la densidad de estados calculada usando simulaciones DFT de la interfaz y después escalada para reproducir el valor experimental en el material masivo. Por otro lado, el valor de la permitividad se asume lineal en la región de transición. La figura 3.16 muestra isosuperficies del ancho de la banda prohibida en la región de transición del óxido. Se puede ver claramente su variación con la posición atómica.

Una vez obtenidos los nuevos valores del ancho de la banda prohibida y la permitividad para los nodos de la malla, evaluamos el impacto de las regiones de transición en las interfaces superior e inferior en las curvas características I_D - V_G del MOSFET de doble puerta. La figura 3.17 muestra los resultados en escalas lineal y logarítmica para los dos dispositivos, el de la transición abrupta y el de la transición gradual. Observamos una mejora en las características del dispositivo cuando introducimos la región de transición: la pendiente subumbral y la corriente con voltaje de puerta alto son mayores. Este efecto se debe principalmente a la transición en la permitividad, ya que en nuestro modelo la región de transición del ancho de la banda prohibida no mostró prácticamente ningún efecto.

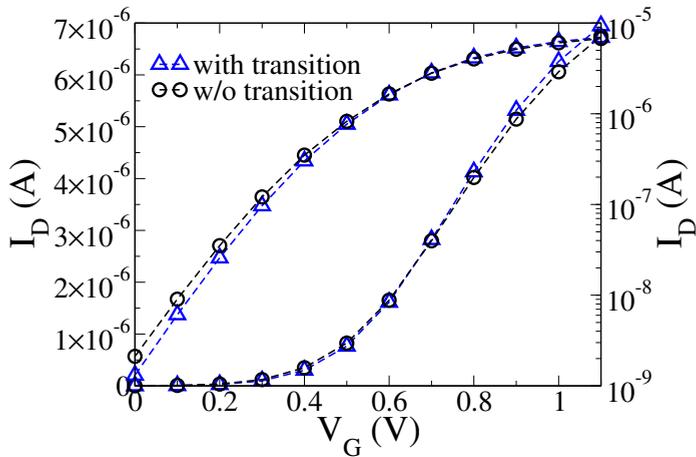


Figura 3.17: Curvas características I_D - V_G para los dispositivos con interfaz abrupta y realista.

3.6.5. Simulación de fluctuaciones en dispositivos IF III-V MOSFET

Como ya comentamos en el capítulo 1, los MOSFETs basados en materiales de alta movilidad electrónica son una de las opciones propuestas para superar las limitaciones en el rendimiento relacionadas con las pobres propiedades de transporte del silicio [KSGL⁺08, CDD⁺05, SLK⁺04]. El desarrollo de un dieléctrico de alta permitividad para el GaAs con una interfaz de alta calidad [Pas05] ha mejorado la perspectiva para la introducción de estos MOSFETs III-V para aplicaciones CMOS de alto rendimiento y baja potencia. Uno de estos candidatos, que será el objeto de nuestro estudio, es la arquitectura *Implant Free* (IF) con canal de $\text{In}_{0.75}\text{Ga}_{0.25}\text{As}$ [KSGL⁺08] que mostramos en la figura 3.18.

Al igual que en los transistores MOSFET convencionales, estos dispositivos se ven afectados por las posiciones aleatorias de los dopantes y otras fuentes de fluctuaciones relacionadas con la naturaleza discreta de la materia. Sin embargo, los IF MOSFETs podrían ser más robustos frente a la variabilidad introducida por los dopantes debido a que están separados físicamente del canal, aunque la proximidad de la capa de dopado δ puede hacer que el efecto todavía sea significativo.

Para realizar las simulaciones hemos extendido el algoritmo descrito en la sección 2.4 para aplicarlo a heteroestructuras de semiconductores III-V. En

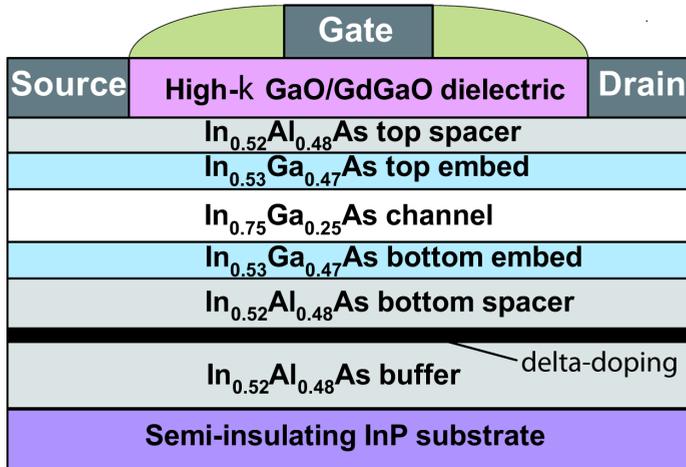


Figura 3.18: Sección transversal de un IF $\text{In}_{0.75}\text{Ga}_{0.25}\text{As}$ MOSFET tipo n con la capa de dopado δ debajo del canal.

este caso suponemos que la constante de red del buffer determina la constante de red en la dirección perpendicular al crecimiento de las capas que están por encima de él, ignorando las posibles dislocaciones. Además, también despreciamos la estructura del buffer, ya que no estudiamos ningún efecto en él y el número de nodos resultaría en simulaciones con requerimientos demasiado elevados. Con estas aproximaciones, el número de nodos y elementos de la malla del dispositivo que utilizamos son 642293 y 3465000 respectivamente para un ancho del dispositivo de aproximadamente 15 nm.

Antes de realizar las simulaciones debemos considerar brevemente las limitaciones del modelo de simulación utilizado. La aproximación de arrastre-difusión utilizada no puede capturar efectos importantes en IF MOSFET con longitudes de puerta muy corta como la dinámica de portadores fuera del equilibrio o el transporte bidimensional en el canal. Sin embargo, cuando calibramos el simulador con resultados obtenidos a partir de simulaciones Monte Carlo [KSGL⁺08], podemos obtener una indicación muy buena de la variabilidad en los parámetros asociados con la electrostática del dispositivo, como es el caso de la tensión umbral [RBAL⁺06].

El dispositivo simulado es un IF $\text{In}_{0.75}\text{Ga}_{0.25}\text{As}$ MOSFET de 15 nm de longitud de puerta. El dopado de la capa de dopado δ es $3 \times 10^{12} \text{cm}^{-2}$ y está situado bajo el canal, como se muestra en la figura 3.18. La estructura de capas es la siguiente: dieléctrico de puerta de alta permitividad ($\text{GaO}_2/\text{GdGaO}$) de

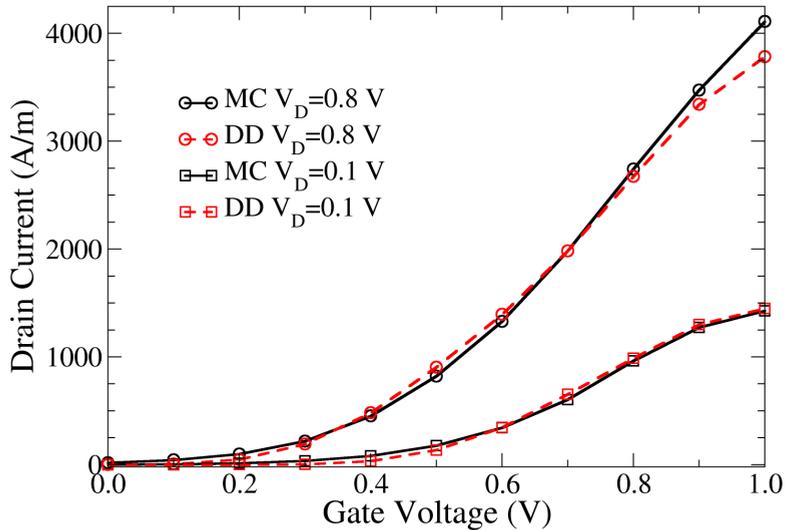


Figura 3.19: Calibración de las curvas I_D - V_G frente a simulaciones Monte Carlo con voltajes de drenador de 0,1 V y 0,8 V para el IF $\text{In}_{0,75}\text{Ga}_{0,25}\text{As}$ MOSFET con longitud de puerta de 15 nm.

1,5 nm de espesor, un separador de 1 nm de espesor de $\text{In}_{0,52}\text{Al}_{0,48}\text{As}$, un canal de 5 nm de espesor de $\text{In}_{0,75}\text{Ga}_{0,25}\text{As}$ incrustado entre dos capas de $\text{In}_{0,53}\text{Ga}_{0,47}\text{As}$, una superior de 0,5 nm y otra inferior de 1 nm, un separador de 2 nm de espesor de $\text{In}_{0,52}\text{Al}_{0,48}\text{As}$, una capa de dopado δ y un buffer de 49 nm de $\text{In}_{0,52}\text{Al}_{0,48}\text{As}$.

Antes de estudiar la influencia de los dopantes aleatorios calibramos el simulador frente a curvas I_D - V_G obtenidas utilizando un simulador Monte Carlo [KDPA07] a voltajes de drenador de 0,1 y 0,8 V. En la figura 3.19 mostramos los resultados de la calibración, obtenidos con una movilidad de campo bajo de $5000 \text{ cm}^2/\text{Vs}$ y una velocidad de saturación de $6,5 \times 10^7 \text{ cm/s}$ en el canal del dispositivo.

Una vez calibrado el dispositivo, simulamos un conjunto de IF MOSFETs microscópicamente diferentes usando la malla atómica. La única fuente de diferencia que consideramos es la naturaleza discreta de los dopantes en la capa de dopado δ . Los dispositivos se crean mediante una técnica de rechazo a partir

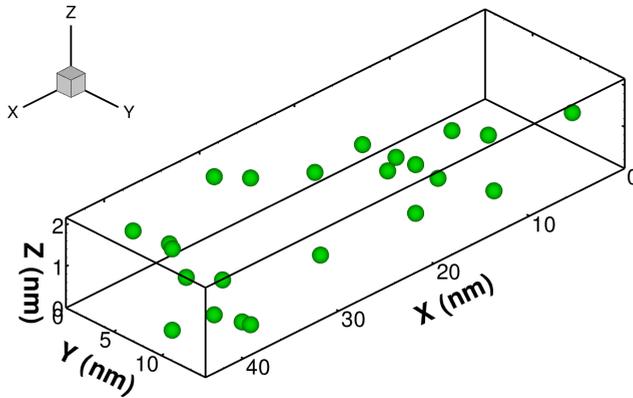


Figura 3.20: Ejemplo de la distribución de dopantes en la capa de dopado δ para un dopado nominal de $1.5 \times 10^{19} \text{ cm}^{-3}$. El número total de átomos en la región es aproximadamente 60000.

del dispositivo con dopado continuo. Para cada nodo de la región, un dopante de Si puede reemplazar un átomo del cristal de InAlAs con una probabilidad determinada por la concentración nominal de dopantes. La figura 3.20 muestra un ejemplo de una capa de dopado δ con 23 átomos de Si.

Cada una de las configuraciones de dopantes crea una configuración del potencial diferente que resulta en diferentes perfiles de concentración a lo largo del canal y diferentes curvas características I-V. Esto se ilustra en las figuras 3.21 y 3.22. La primera muestra la distribución de potencial en equilibrio en la capa de dopado δ y en el canal para una configuración de dopantes, mientras que la segunda muestra el efecto de este dopado aleatorio sobre la concentración electrónica en la parte inferior del canal.

En este estudio hemos simulado más de 200 transistores microscópicamente diferentes con una polarización de drenador de 0,1 V. La figura 3.23 muestra la tensión umbral obtenida calculada usando un criterio de corriente constante [RBAL⁺06] y el número correspondiente de dopantes en la región de dopado δ para todos los transistores del conjunto. Podemos ver claramente una correlación entre el número de dopantes y la tensión umbral. Las simulaciones con un número bajo de dopantes resultan en una tensión umbral mayor debido a que la carga disponible para el transporte es también menor. Además, en las simulaciones con un mismo número de dopantes podemos ver también una dispersión debida a las posiciones aleatorias de los dopantes. La figura 3.23

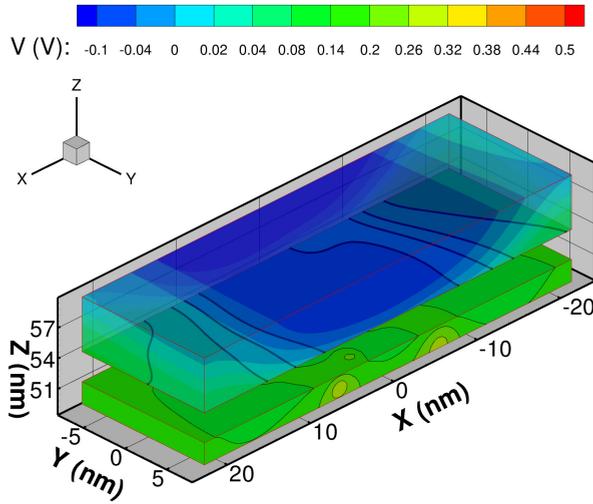


Figura 3.21: Distribución de potencial en equilibrio en la capa de dopado δ y en el canal para una configuración de dopantes generada aleatoriamente.

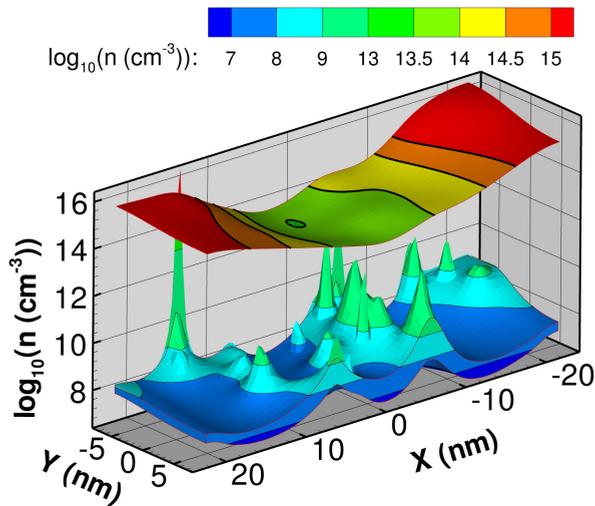


Figura 3.22: Concentración de electrones en equilibrio en la capa de dopado δ y en el canal para una configuración de dopantes generada aleatoriamente.

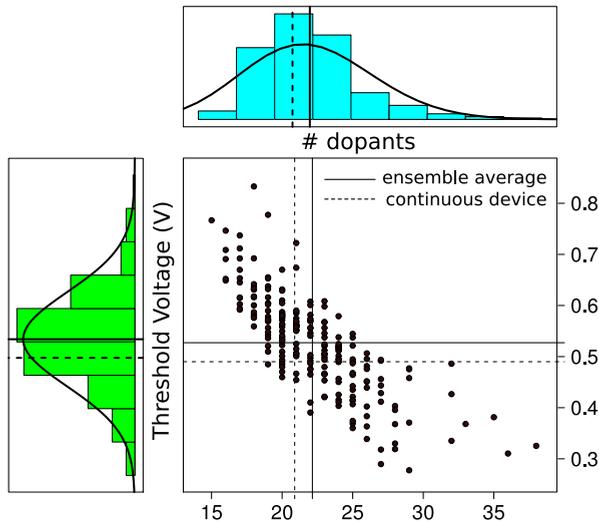


Figura 3.23: Diagrama de dispersión que muestra la tensión umbral y el número de dopantes para cada simulación del conjunto de transistores. En los lados izquierdo y superior se muestran los histogramas de cada variable junto con la función de densidad de probabilidad esperada y el valor de la variable en el caso de la simulación continua.

también muestra una comparación de los valores medios de la muestra con los obtenidos al simular el dispositivo con dopado continuo. Podemos ver que existe un desplazamiento hacia valores de tensión umbral mayores que el dispositivo continuo, aunque este desplazamiento podría tener su origen en parte en la acumulación de carga artificial cerca de los dopantes, lo que provoca un excesivo incremento de la resistividad, al igual que sucedía en el caso de los transistores MOSFET de Si. Además, podemos ver que el número medio de dopantes no coincide exactamente (22.1 de media frente a 20.89 esperados) con la carga en la región de dopado aleatorio. Esto se debe a que, al utilizar un criterio de corriente constante, hay algunas simulaciones con un número bajo de dopantes que no llegan a alcanzar la tensión umbral en el rango de simulación permitido. Por tanto, es de esperar que el desplazamiento del valor medio de la tensión umbral se desplace hacia valores un poco más altos. Pese a estos inconvenientes, la distribución obtenida es todavía cualitativamente válida y da una buena estimación de las fluctuaciones asociadas con la electrostática del dispositivo esperadas por la aleatoriedad de los dopantes en la capa de dopado δ [ABD⁺03]. La dispersión en los valores de la tensión um-

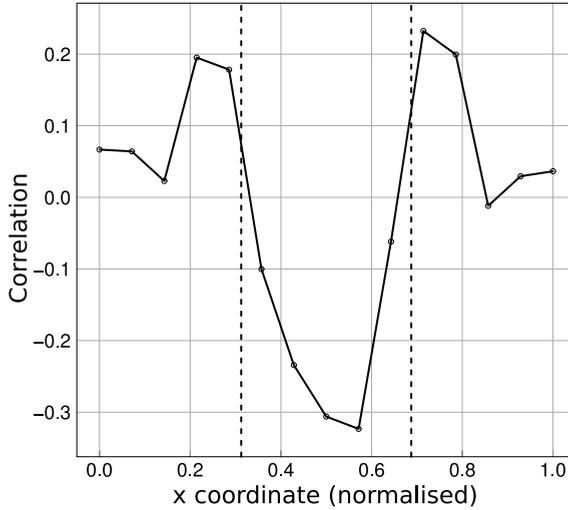


Figura 3.24: Correlación entre la posición de los dopantes a lo largo del eje x y el valor de la tensión umbral.

bral, medida con la desviación estándar de la distribución $\sigma V_T = 94,5 \text{ mV}$, es similar a la previamente publicada para MOSFETs de silicio con longitud de puerta similar [RBAL⁺06] ($\sigma V_T = 100 \text{ mV}$).

Para poder evaluar la contribución a esta variabilidad de la posición aleatoria de los dopantes simulamos un conjunto de dispositivos con un número constante de dopantes, 23, en la capa de dopado δ . La simulación de este conjunto de dispositivos produce un conjunto de valores de tensión umbral con una desviación estándar de $\sigma V_T = 27,7 \text{ mV}$. Esto confirma la gran importancia del papel del número aleatorio de dopantes en la magnitud total de las fluctuaciones que ya se veía en la figura 3.23. Además, con estas simulaciones podemos evaluar también la relación que existe entre la posición de los dopantes y la tensión umbral. En la figura 3.24 podemos ver la correlación existente entre la posición de los dopantes a lo largo del eje x y el valor de la tensión umbral. Vemos que los mayores valores absolutos de la correlación están en la zona del canal. En este caso los valores son negativos, ya que un mayor número de dopantes en la región produce una disminución mayor en la barrera de la banda de conducción y por tanto facilitan el paso de electrones de fuente a drenador, disminuyendo así la tensión umbral. Por otra parte, también vemos picos de correlación positiva en las proximidades de la puerta. Esto se puede

deber al efecto de atrapado de carga que comentamos antes, ya que en estas regiones el efecto sobre la barrera perdería importancia y dominaría el efecto de captura de carga cerca de los dopantes que aumenta la resistividad.

Capítulo 4

El método de simulación Monte Carlo

Cuando el modelo de arrastre-difusión deja de ser válido y/o cuando pierde carácter predictivo, puede ser necesario pasar a utilizar modelos de transporte más complejos. El modelo más completo en un marco semiclásico es el que se basa en la resolución de la ecuación de transporte de Boltzmann. Debido a su elevada complejidad computacional, es un método que todavía no se utiliza de forma universal en aplicaciones que requieren un cálculo rápido del comportamiento de un dispositivo. Sin embargo, es una técnica cada vez más extendida tanto en entornos académicos como industriales, tanto por la mayor capacidad de los recursos computacionales actuales como por la necesidad de utilizar modelos de transporte complejos para predecir el comportamiento de los dispositivos más avanzados.

En la primera parte de este capítulo describimos la ecuación de transporte de Boltzmann, que se resolverá utilizando un método Monte Carlo acoplada con la ecuación de Poisson, resuelta mediante el método de los elementos finitos tal y como se describió en el capítulo anterior. En primer lugar obtendremos la ecuación a partir de ciertas consideraciones clásicas. Después de esto describimos la resolución de la ecuación mediante un método Monte Carlo y su paralelización. Presentaremos además los modelos utilizados para describir las bandas de los materiales y los principales modelos de dispersión utilizados. Finalmente describiremos algunos aspectos concretos de la implementación sobre mallas tetraédricas y algunos resultados de la simulación de transistores MOSFET multipuerta.

4.1. Ecuación de transporte de Boltzmann

El modelo de arrastre-difusión, descrito en el capítulo anterior, se puede ver como una aproximación a la ecuación de transporte de Boltzmann [Sel84] asumiendo que el transporte de portadores tiene lugar en equilibrio (un cambio de estado en el sistema es más lento que la relajación de energía en el sistema) y que la temperatura del electrón es la misma que la de la red cristalina. Sin embargo, con el escalado de los dispositivos, los procesos fuera del equilibrio empiezan a jugar un papel determinante en su comportamiento.

Para estudiar un gas clásico, Boltzmann introdujo una ecuación que describe la evolución temporal de la función de distribución f debida a colisiones binarias entre partículas [Rei65]. Esta función, $f(\mathbf{r}, \mathbf{k}, t)$, es la cantidad fundamental en la ecuación de transporte de Boltzmann y da la probabilidad de encontrar una partícula en un instante t con posición \mathbf{r} y vector de onda \mathbf{k} . En esencia, la ecuación de transporte de Boltzmann describe su evolución en el espacio de fases [JL89].

Una forma de obtener la ecuación es haciendo un balance de las partículas que entran y salen de un volumen infinitesimal del espacio de fases. Teniendo en cuenta que el número de partículas en este volumen es

$$dn = f(\mathbf{r}, \mathbf{k}, t) \frac{d\mathbf{k}}{4\pi^3} d\mathbf{r}, \quad (4.1)$$

podemos calcular su variación en un tiempo dt tanto en el espacio de posiciones como en el espacio k . La variación en el espacio de posiciones (difusión de portadores) es

$$- (\mathbf{v} \nabla_{\mathbf{r}} f) \frac{d\mathbf{k}}{4\pi^3} d\mathbf{r} dt \quad (4.2)$$

y en el espacio k (arrastre de portadores),

$$- \left(\frac{d\mathbf{k}}{dt} \nabla_{\mathbf{k}} f \right) \frac{d\mathbf{k}}{4\pi^3} d\mathbf{r} dt. \quad (4.3)$$

Por último, hay que considerar la variación debida a la interacción de los portadores con perturbaciones locales, lo que se conoce como colisiones. Las variaciones debidas a estos procesos están determinadas por la probabilidad de transición $P(\mathbf{k}, \mathbf{k}')$, que da la probabilidad por unidad de tiempo de pasar de un estado inicial \mathbf{k} a un estado final \mathbf{k}' . El número de electrones que abandonan el volumen del espacio de fases será

$$- f(\mathbf{r}, \mathbf{k}, t) P(\mathbf{k}, \mathbf{k}') [1 - f(\mathbf{r}, \mathbf{k}', t)] \frac{d\mathbf{k}'}{4\pi^3} \frac{d\mathbf{k}}{4\pi^3} d\mathbf{r} dt. \quad (4.4)$$

Análogamente, el número de electrones que entra desde un estado \mathbf{k}' es

$$f(\mathbf{r}, \mathbf{k}', t)P(\mathbf{k}', \mathbf{k})[1 - f(\mathbf{r}, \mathbf{k}, t)] \frac{d\mathbf{k}}{4\pi^3} \frac{d\mathbf{k}'}{4\pi^3} d\mathbf{r}dt. \quad (4.5)$$

Además, hay que considerar todos los posibles estados \mathbf{k}' , por lo que integramos en la primera zona de Brillouin (*BZ*):

$$\begin{aligned} \frac{d\mathbf{k}}{4\pi^3} d\mathbf{r}dt \int_{BZ} \{f(\mathbf{r}, \mathbf{k}', t)P(\mathbf{k}', \mathbf{k})[1 - f(\mathbf{r}, \mathbf{k}, t)] - \\ - f(\mathbf{r}, \mathbf{k}, t)P(\mathbf{k}, \mathbf{k}') [1 - f(\mathbf{r}, \mathbf{k}', t)]\} \frac{d\mathbf{k}'}{4\pi^3}. \end{aligned}$$

Los términos de la forma $(1 - f)$ representan los estados finales libres (principio de exclusión de Pauli), y es bastante común despreciarlos en las simulaciones o incluirlos *a posteriori*.

La variación total de la función de distribución, obtenida a partir de estas contribuciones, está dada por la ecuación de transporte de Boltzmann:

$$\begin{aligned} \frac{\partial f}{\partial t} = -\mathbf{v}\nabla_{\mathbf{r}}f - \frac{d\mathbf{k}}{dt}\nabla_{\mathbf{k}}f + \int_{BZ} \{f(\mathbf{r}, \mathbf{k}', t)P(\mathbf{k}', \mathbf{k})[1 - f(\mathbf{r}, \mathbf{k}, t)] - \\ - f(\mathbf{r}, \mathbf{k}, t)P(\mathbf{k}, \mathbf{k}') [1 - f(\mathbf{r}, \mathbf{k}', t)]\} \frac{d\mathbf{k}'}{4\pi^3}. \quad (4.6) \end{aligned}$$

Es interesante decir que la BTE también se puede obtener a partir de la ecuación de Schrödinger asumiendo un hamiltoniano específico e imponiendo ciertas simplificaciones [KL57].

Las probabilidades de transición por unidad de tiempo se obtienen para cada mecanismo de dispersión utilizando la regla de oro de Fermi [JL89, JR83b]. Por tanto, la probabilidad de transición del estado $|\mathbf{k}, c\rangle$ al estado $|\mathbf{k}', c'\rangle$ estará dada por:

$$P(\mathbf{k}, c; \mathbf{k}', c') = \frac{2\pi}{\hbar} |\langle \mathbf{k}', c' | H' | \mathbf{k}, c \rangle|^2 \delta[\varepsilon(\mathbf{k}, c) - \varepsilon(\mathbf{k}', c')], \quad (4.7)$$

donde $\varepsilon(\mathbf{k}, c)$ representa la energía del sistema y H' es el hamiltoniano de la perturbación, que depende del mecanismo de dispersión en particular. Para calcular estas probabilidades se asume que el estado $|\mathbf{k}, c\rangle$, que representa al electrón en el que estamos interesados junto con el resto del sistema, se puede escribir como el producto directo $|\mathbf{k}, c\rangle = |\mathbf{k}\rangle |c\rangle$, donde $|\mathbf{k}\rangle$ y $|c\rangle$ representan los estados no perturbados del electrón y del cristal respectivamente. Para obtener

la probabilidad de transición para el electrón $P(\mathbf{k}, \mathbf{k}')$ habría que promediar sobre los posibles estados $|c\rangle$.

El término de colisiones en (4.6) es una suma sobre eventos de dispersión individuales. Esto implica que la duración de las colisiones τ_c debe ser mucho menor que cualquier otra escala temporal del problema [KL57]. En otras palabras, las colisiones se consideran procesos instantáneos que interrumpen los vuelos libres. Esto supone un acoplamiento débil electrón-fonón y electrón-electrón durante su interacción mutua. La validez de la BTE requiere:

$$\tau_c \ll \tau_L, \quad (4.8)$$

donde el tiempo de tránsito τ_L se define como el cociente l/\bar{v} , donde l es el recorrido libre medio de los portadores y \bar{v} es una velocidad típica de los portadores. Bajo la condición (4.8), uno espera una respuesta instantánea de la función de distribución a los campos y la dispersión y la distribución debería evolucionar, en un sentido markoviano, a la ecuación local del movimiento. Sin embargo, esta aproximación falla en algunos casos [Hes91]:

- Si las tasas de dispersión son altas, f puede cambiar apreciablemente en una escala temporal comparable a la de los procesos de colisión. Una posibilidad para añadir los efectos de las tasas de dispersión altas es sustituir la función $\delta[\varepsilon(\mathbf{k}, c) - \varepsilon(\mathbf{k}', c')]$ en (4.7) por una función con ancho finito como una lorentziana.
- Si los campos eléctricos son muy altos, los portadores pueden acelerar apreciablemente durante el tiempo de una colisión (efecto campo intracolisional [Bar73]).

La función de distribución de electrones $f(\mathbf{r}, \mathbf{k}, t)$ representa esencialmente los elementos diagonales de la matriz densidad de electrones $\rho(\mathbf{r}, \mathbf{k}, t)$ en el espacio de momentos. Esta construcción sólo tiene sentido práctico si:

1. los estados del momento $|\mathbf{r}, \mathbf{k}\rangle$ son buenas aproximaciones a los estados electrónicos reales,
2. la función de distribución satisface una ecuación cerrada del movimiento,
3. se pueden construir condiciones de contorno correctas para f .

La primera condición no es restrictiva, ya que siempre podemos elegir un conjunto mejor de estados. El vector de onda del electrón está definido por

el centro de gravedad del paquete de ondas $|\mathbf{r}, \mathbf{k}\rangle$, lo que implica que la extensión del paquete de ondas en el espacio k debe ser pequeña comparada con las dimensiones de la zona de Brillouin. La segunda condición es más difícil de satisfacer. Si la ecuación de Liouville-von Neumann [Sch68] se proyecta formalmente en ecuaciones para f , también incluirá la dinámica de los fonones. Si estas ecuaciones se resuelven para los elementos no diagonales en términos de f , las ecuaciones para f sólo pueden ser cerradas en la forma integro-diferencial. Sin embargo, si los electrones no son independientes como en el caso de que la dinámica de los fonones afecte a la de los electrones, las ecuaciones para f no pueden ser cerradas y tienen que ser resueltas iterativamente.

4.2. El método Monte Carlo aplicado a la simulación del transporte en semiconductores

Probablemente la forma más común de resolver la ecuación de transporte de Boltzmann es el uso de métodos de Monte Carlo. Estos métodos [MU49] se usan ampliamente en varios campos de la física (física nuclear, de estado sólido, estadística, etc.). Cuando se aplican al transporte de portadores en sólidos, este método proporciona una solución numérica exacta de la BTE sin la necesidad de resolverla directamente. En otras palabras, la función de distribución $f(\mathbf{r}, \mathbf{k}, t)$ obtenida del método Monte Carlo satisface la BTE.

Este método es el más popular para la resolución de la BTE porque permite obtener una solución usando algoritmos eficientes y relativamente sencillos cuando se comparan con otros métodos numéricos directos. Además, este método permite una interpretación microscópica satisfactoria de los procesos simulados. Su esencia consiste en la simulación del movimiento de un conjunto de portadores tanto en el espacio k como en el espacio \mathbf{r} . El movimiento de cada portador está gobernado por las ecuaciones del movimiento clásicas y por colisiones aleatorias con diferentes perturbaciones (por ejemplo fonones, iones u otros portadores). Estas colisiones causan transiciones instantáneas entre estados Bloch no perturbados con una probabilidad obtenida mediante la regla de oro de Fermi. Usando estas probabilidades, el tiempo de vuelo libre, el canal de dispersión y el estado final se generan de forma aleatoria. Durante la simulación, cualquier magnitud física dependiente de \mathbf{k} , \mathbf{r} o t puede ser calculada.

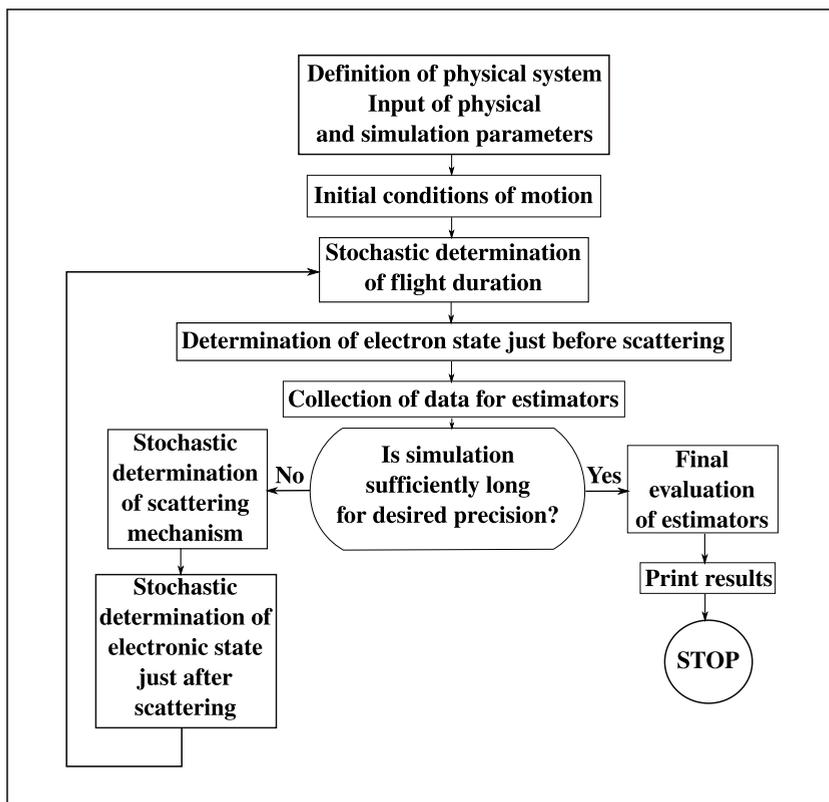


Figura 4.1: Flujo de un programa Monte Carlo típico.

4.2.1. Vuelo libre de los portadores

El punto inicial del programa MC (ver figura 4.1) es la definición del sistema físico de interés, incluyendo los parámetros materiales y valores de magnitudes físicas tales como la temperatura T_0 o el campo eléctrico \mathbf{E} [JLFS91]. A este nivel también definimos los parámetros que controlan la simulación: duración de cada subhistoria y paso temporal dt , la precisión deseada en el resultado, etc. El siguiente paso es un cálculo preliminar de cada tasa de dispersión como función de la energía del portador. La elección de la relación de dispersión $\varepsilon(\mathbf{k})$ suele depender del problema de transporte simulado. En nuestro caso hemos decidido usar una estructura de bandas analítica, que proporciona un buen compromiso entre precisión y eficiencia, ya que nuestra intención es la simulación del transporte de MOSFETs en la escala de los nanómetros. En este caso se espera que el voltaje de alimentación esté por debajo de 1 V y no tenemos como objetivo la simulación de ningún efecto de ruptura a campos elevados [PDG04]. Por tanto, empleamos una ley de dispersión no parabólica que incluye la anisotropía de los valles de los materiales:

$$\sum \frac{\hbar^2 k_\alpha^2}{2m_\alpha} = \varepsilon(1 + \alpha\varepsilon), \quad \alpha = x, y, z. \quad (4.9)$$

El siguiente paso es la generación de la duración de los vuelos (figura 4.1). El vector de onda del electrón \mathbf{k} cambia de forma continua durante un vuelo libre debido a los campos aplicados. Por tanto, si $\lambda[\mathbf{k}(t)]$ es la probabilidad de dispersión para un electrón en el estado \mathbf{k} durante el intervalo temporal dt , la probabilidad de que el electrón, que ya ha sufrido una colisión en $t = 0$, no haya colisionado después de un tiempo t es:

$$\exp \left[- \int_0^t dt' \lambda[\mathbf{k}(t')] \right], \quad (4.10)$$

que da la probabilidad de que el intervalo $(0, t)$ no contenga ninguna dispersión. Consecuentemente, la probabilidad $P(t)$ de que el electrón sufra su próxima colisión durante el intervalo dt en el tiempo t es:

$$P(t)dt = \lambda[\mathbf{k}(t)] \exp \left[- \int_0^t dt' \lambda[\mathbf{k}(t')] \right] dt. \quad (4.11)$$

El tiempo de vuelo libre t puede ser generado a partir de la ecuación:

$$r = \int_0^t dt' P(t'), \quad (4.12)$$

donde r es un número aleatorio entre 0 y 1.

4.2.2. Dispersión de los portadores

Una vez terminado el vuelo libre del electrón hay que elegir el mecanismo de dispersión. El peso del i -ésimo mecanismo (donde hay n mecanismos diferentes) está dado por:

$$P_i(\mathbf{k}) = \frac{\lambda_i(\mathbf{k})}{\lambda(\mathbf{k})}, \quad \lambda(\mathbf{k}) = \sum_{i=1}^n \lambda_i(\mathbf{k}). \quad (4.13)$$

Se genera un número aleatorio entre 0 y 1 y se comprueban las siguientes desigualdades:

$$\sum_{i=1}^{j-1} \frac{\lambda_i(\mathbf{k})}{\lambda(\mathbf{k})} < r < \sum_{i=1}^j \frac{\lambda_i(\mathbf{k})}{\lambda(\mathbf{k})}, \quad j = 1, \dots, n. \quad (4.14)$$

Aceptamos el j -ésimo mecanismo si la j -ésima desigualdad se verifica. El cálculo del tiempo de vuelo libre y del canal de dispersión pueden ser simplificados introduciendo la “autodispersión” λ_0 [Ree69, Ree68].

Finalmente hay que determinar el estado \mathbf{k}' al que pasará el electrón después de la dispersión. Si el vuelo libre terminó con una autodispersión, entonces \mathbf{k}' será igual a \mathbf{k} , el estado antes de la dispersión. Cuando suceda una dispersión real, el estado \mathbf{k}' debe ser generado de forma aleatoria de acuerdo con la sección eficaz diferencial del mecanismo de dispersión particular.

4.3. Técnicas de paralelización de simuladores Monte Carlo

Podemos considerar dos formas principales de paralelizar el simulador Monte Carlo. La primera de ellas consiste en la repartición de superpartículas entre procesadores usando la misma estrategia de descomposición de dominios que en el caso de la resolución de la ecuación de Poisson. Con este método, cada procesador simularía las superpartículas que se encuentran en el interior de su subdominio asignado. Al utilizar esta aproximación es recomendable hacer particiones que tengan en cuenta la densidad de partículas que habrá en cada zona del dispositivo, ya que si no nos podríamos encontrar con un desbalanceo importante en el caso de la subrutina que resuelve el transporte. Además, también tendrá influencia en el balanceo la probabilidad de dispersión en cada región del dispositivo. Por último, en este caso es necesario considerar la

posibilidad de que una partícula cambie de dominio durante un vuelo libre. Si no se tiene cuidado se pueden dar iteraciones en las que cada partícula cambie varias veces de dominio. Este problema se puede aliviar mediante el uso de regiones de solapamiento entre dispositivos, que permiten terminar la mayor parte de los vuelos en el mismo procesador en el que empiezan. En este caso, las partículas que terminaran el paso temporal en la región de solapamiento se enviarían al procesador vecino apropiado.

La segunda estrategia de paralelización, que es la que hemos adoptado, consiste en simular en cada procesador una cantidad aproximadamente igual de superpartículas independientemente de su posición en el dominio de simulación. Esto requiere replicar la malla completa con algunas de sus propiedades en todos los procesadores. El primer problema asociado con este método es la necesidad de almacenar la malla completa en cada procesador. Sin embargo, esto no es una restricción importante, ya que la cantidad de memoria necesaria es de unos pocos MB incluso para una simulación relativamente grande. El otro problema es la necesidad de comunicaciones de arrays del tamaño de la malla para obtener el valor del potencial en los otros procesadores y para actualizar las densidades de carga necesarias en la resolución de Poisson.

En este último caso los vuelos libres se realizan completamente en paralelo hasta la etapa de inyección de partículas en los contactos necesaria para que se verifiquen las condiciones de contorno en los contactos óhmicos. En este momento es necesaria una reducción para obtener la carga total en los contactos, que se comparará con el dopado para comprobar si se verifica la condición de equilibrio térmico [GP96] y si no es así, inyectar las superpartículas necesarias. Una vez inyectadas es necesario hacer una reducción sobre la densidad de carga en los nodos para poder usarla en la resolución de la ecuación de Poisson en cada subdominio. Una vez resuelta, los valores locales del potencial electrostático deben ser enviados a todos los procesadores para usarlos en el cálculo de los campos eléctricos necesarios para el vuelo libre de los portadores de la siguiente iteración temporal. La figura 4.2 muestra un esquema del proceso completo.

Para evaluar la eficiencia paralela del código simulamos un DG MOSFET con longitud de canal de 4 nm, espesor del cuerpo de silicio de 3.3 nm y 0.54 nm de espesor de óxido de puerta. El dopado de las regiones de fuente y drenador es constante e igual a $1 \times 10^{20} \text{ cm}^{-3}$, mientras que la región de canal se mantiene sin dopar. Para las medidas de la aceleración y la eficiencia las simulaciones se ejecutaron únicamente 100 iteraciones temporales en un cluster HP Superdome con 128 procesadores Itanium 2 a 1.5 GHz.

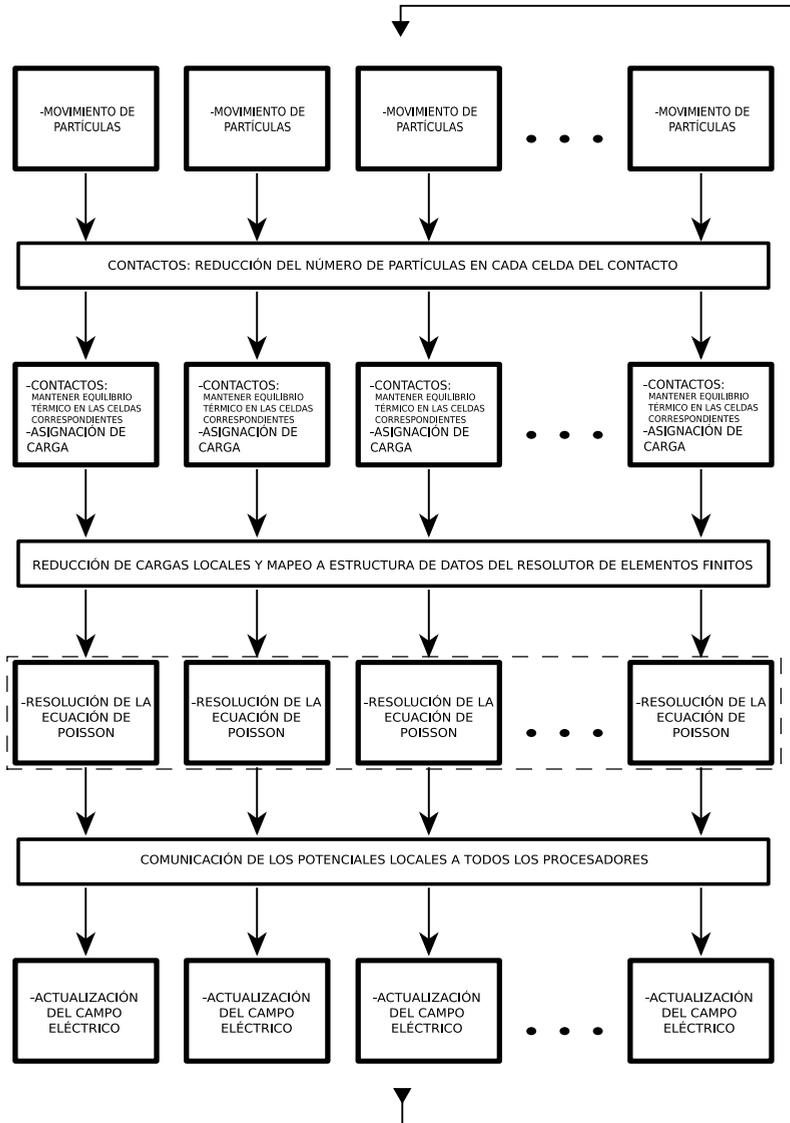


Figura 4.2: Diagrama de la subrutina Monte Carlo paralelizada.

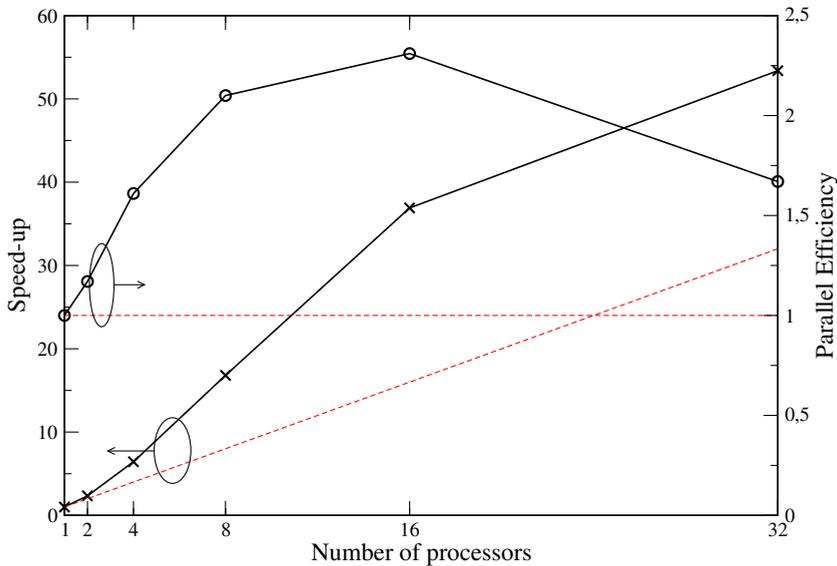


Figura 4.3: Aceleración (cruces) y eficiencia paralela (círculos) de un paso temporal. Los valores de escalado ideal (líneas discontinuas) se muestran por comparación.

Para estudiar la escalabilidad del programa utilizaremos como medida la aceleración (S) y la eficiencia paralela (PE), definidas para p procesadores como:

$$S(p) = \frac{t_1}{t_p}, \quad (4.15)$$

$$PE(p) = \frac{t_1}{t_p p}, \quad (4.16)$$

donde t_1 y t_p son los tiempos de ejecución en uno y p procesadores respectivamente. En la figura 4.3 se muestra su dependencia con el número de procesadores. Se puede ver que obtenemos supereficiencia, efecto que se debe al escalado del resolutor de Poisson y empieza a desaparecer cuando aumentamos el número de procesadores, ya que los tiempos de las operaciones que no escalan con el número de procesadores y las comunicaciones empiezan a ser del mismo orden de magnitud que los tiempos de la resolución de los sistemas lineales o de los vuelos libres.

La tabla 4.1 muestra los tiempos de ejecución de las diferentes partes de la iteración principal del lazo Monte Carlo. La escalabilidad de los vuelos libres es muy buena, y está limitada principalmente por la fase de inyección

Tabla 4.1: Tiempos medios, en segundos, de la simulación de un paso temporal ($t_{iteration}$), de una resolución de la ecuación de Poisson ($t_{Poisson}$), de los vuelos libres (t_{flight}), de la etapa de inyección en los contactos ($t_{cont.}$), de la comunicación de la concentración de electrones ($t_{reduc.}$), de la comunicación del potencial ($t_{com.pot.}$) y de la actualización del campo eléctrico (\vec{E}_{update}) para diferente número de procesadores (n_p).

| n_p | $t_{iteration}$ | $t_{Poisson}$ | t_{flight} | $t_{cont.}$ | $t_{reduc.}$ | $t_{com.pot.}$ | \vec{E}_{update} |
|-------|-----------------|---------------|--------------|-------------|--------------|----------------|--------------------|
| 1 | 27.50 | 24.9 | 0.68 | 0.35 | 0.013 | 0.019 | 0.207 |
| 2 | 11.74 | 11.1 | 0.33 | 0.20 | 0.010 | 0.007 | 0.204 |
| 4 | 4.27 | 3.87 | 0.173 | 0.146 | 0.0085 | 0.0054 | 0.203 |
| 8 | 1.64 | 1.08 | 0.102 | 0.212 | 0.0188 | 0.0050 | 0.198 |
| 16 | 0.745 | 0.386 | 0.0420 | 0.101 | 0.005 | 0.0042 | 0.200 |
| 32 | 0.515 | 0.186 | 0.0209 | 0.079 | 0.009 | 0.006 | 0.206 |

de partículas en los contactos por las comunicaciones que requiere para determinar las partículas necesarias para mantener la condición de equilibrio. La otra etapa que vemos que limita la escalabilidad es la actualización del campo eléctrico, que se replica por completo en todos los procesadores. En el caso de la actualización del campo eléctrico se podría paralelizar, pero dependería del algoritmo que utilicemos para su cálculo.

4.4. Simulaciones Monte Carlo del transporte en silicio

Al simular el transporte en un cristal semiconductor debemos tener en cuenta las interacciones con todos los átomos constituyen. Bajo ciertas condiciones podemos tener en cuenta el efecto de la red cristalina únicamente mediante la estructura de bandas del material, que relaciona la energía con el vector de onda del portador. Además, es importante tener en cuenta la interacción con imperfecciones en forma de desviaciones de la periodicidad, como son las impurezas, otros portadores, vibraciones de la red, etc. En esta sección describiremos brevemente estos dos aspectos de las simulaciones, centrándonos en las características más relevantes para el transporte en silicio.

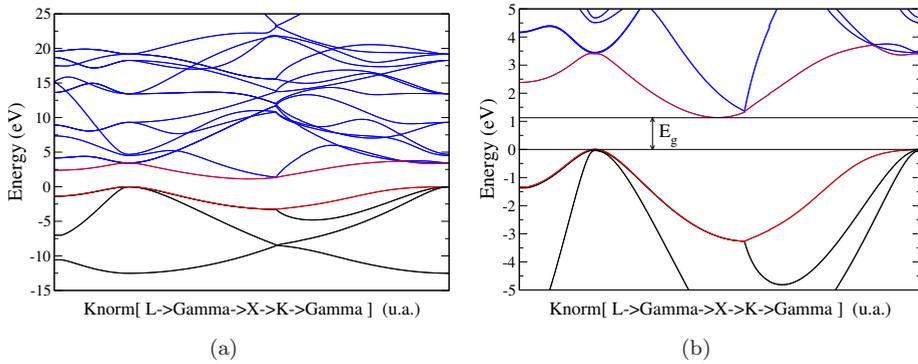


Figura 4.4: (a) Estructura de bandas del Si calculada usando la aproximación de electrones fuertemente ligados. (b) Detalle de la región próxima a la banda de energías prohibidas. En ambos casos las primeras bandas de valencia y conducción están resaltadas en rojo. El resto de las bandas de conducción están en azul y las de valencia en negro.

4.4.1. Estructura de bandas y aproximación analítica

El cálculo de la estructura de bandas de un material es un problema complejo que se ha estudiado en detalle. Existen numerosos métodos para su cálculo, tanto en materiales simples como para aleaciones, heterouniones, sistemas confinados, etc. [RAF04, SRG⁺08, NCA⁺06]. Estos cálculos tienen una gran importancia, ya que la estructura de bandas determina muchas de las propiedades del transporte y debe conocerse para la simulación predictiva de dispositivos semiconductores. Además, estos cálculos permiten obtener la influencia de factores como la tensión o el confinamiento sobre las bandas de energía lo que es de especial importancia para los dispositivos actuales [ZRL⁺05, SKBA08]. En la figura 4.4 mostramos las bandas para el silicio calculadas mediante la aproximación de electrones fuertemente ligados (*Tight Binding*, TB). En la figura 4.5 mostramos la primera zona de Brillouin de una red fcc, como es el caso del silicio, con los puntos de simetría usados en la figura 4.4.

Sin embargo, en muchos casos no es necesario tener en cuenta la estructura de bandas al completo, sino que es suficiente el comportamiento cerca del mínimo de la banda de conducción o del máximo de la banda de valencia. En estos casos es suficiente un modelo que permita describir de forma correcta estas regiones. Uno de los modelos más utilizados es el de la aproximación no

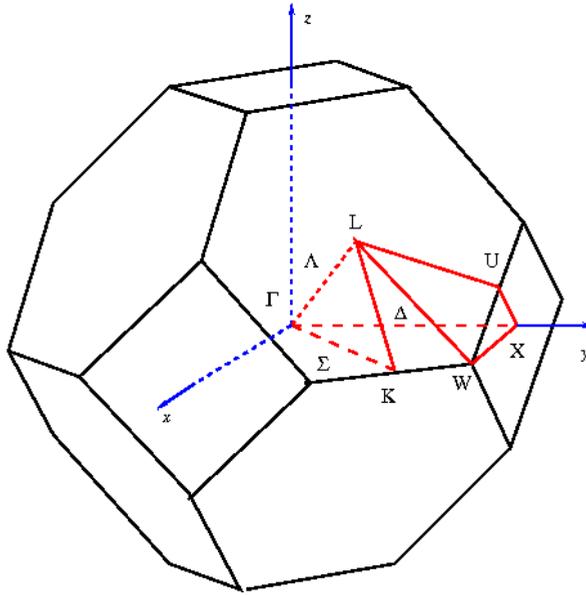


Figura 4.5: Primera zona de Brillouin para una red fcc en la que se muestran puntos de alta simetría.

parabólica obtenido a partir de la teoría $\mathbf{k} \cdot \mathbf{p}$:

$$\varepsilon(\mathbf{k})(1 + \alpha\varepsilon(\mathbf{k})) = \frac{\hbar^2}{2}\mathbf{k}\mathbf{M}^{-1}\mathbf{k}, \quad (4.17)$$

donde α es el factor de no parabolicidad y \mathbf{M}^{-1} es la inversa del tensor de masa efectiva,

$$\mathbf{M}^{-1} = \left\{ \frac{1}{\hbar} \frac{\partial^2 \varepsilon(\mathbf{k})}{\partial k_i \partial k_j} \right\}. \quad (4.18)$$

En el caso del silicio el factor de no parabolicidad es $\alpha = 0,5 \text{ eV}^{-1}$ y los valles principales para el transporte son los X, que se modelan como elipsoides con masas efectivas longitudinal y transversal de $0,98m_0$ y $0,19m_0$ respectivamente, como mostramos en la figura 4.6. Además, el modelo de bandas utilizado también incluye otros valles de menor importancia en el silicio como son los valle L y Γ .

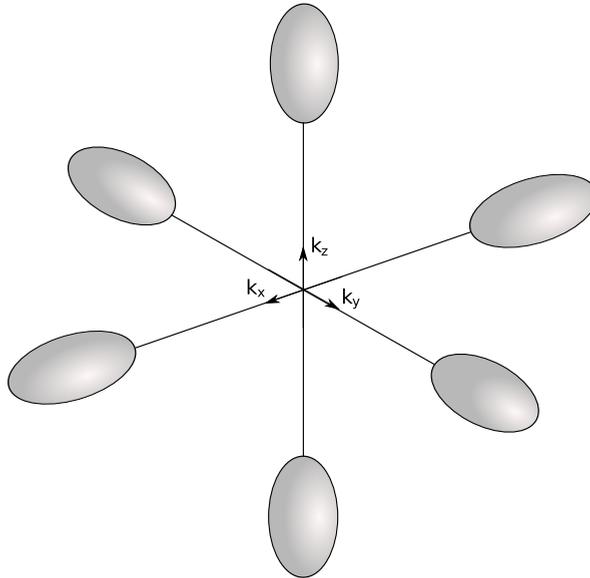


Figura 4.6: Superficies de energía constante próximas al mínimo de la banda de conducción.

4.4.2. Principales mecanismos de dispersión

El movimiento de los portadores en el semiconductor está compuesto por fases de arrastre en las que son acelerados por los campos eléctricos y colisiones, que interrumpen estos vuelos libres. Durante la simulación consideraremos las colisiones, como suele ser habitual, utilizando la regla de oro de Fermi, que surge de la teoría de perturbaciones de primer orden. Con esta aproximación podemos calcular la probabilidad de transición entre dos estados, $P(\mathbf{k}, \mathbf{k}')$, debida a la acción de una perturbación H' mediante la ecuación (4.7). Además, una vez obtenida esta expresión para una perturbación dada, debemos integrar a todos los estados finales \mathbf{k}' para obtener la probabilidad de dispersión $W(\mathbf{k})$ con la perturbación.

Los mecanismos de dispersión incluidos en nuestras simulaciones son: fonones acústicos, fonones ópticos no polares intravalle, fonones ópticos no polares intervale para transiciones X-X (procesos g y f), fonones ópticos no polares intervale para transiciones X-L (procesos p) y X- Γ e impurezas ionizadas.

Dispersión por fonones

En un potencial periódico perfecto los electrones no sufrirían dispersiones. Sin embargo, en un cristal semiconductor existen desviaciones de la periodicidad. Las vibraciones de la red tienen una naturaleza oscilatoria y pueden ser cuantizadas como fonones, que influenciarán el movimiento de los electrones mediante las llamadas interacciones electrón-fonón, que se encuentran entre los principales mecanismos de dispersión en los dispositivos semiconductores actuales. Hay dos tipos de modos de fonones: los de la rama acústica y los de las ramas ópticas. En el primer caso los átomos de la celda unidad se mueven en la misma dirección, por lo que se produce un desplazamiento efectivo de la celda unidad al completo. En este caso, para longitudes de onda lo suficientemente grandes, las amplitudes de vibración cambiarán poco entre celdas vecinas, por lo que la estructura es poco determinante y se puede aproximar el medio como continuo [BS50]. Así, la perturbación producida por las variaciones en la constante de red pueden expresarse en función de la tensión inducida, $\nabla\mathbf{u}(\mathbf{r}, t)$ [Tom93]. En este caso el hamiltoniano de la perturbación puede expresarse como

$$H' = \Xi_d \nabla\mathbf{u}(\mathbf{r}, t), \quad (4.19)$$

donde la constante de proporcionalidad Ξ describe el desplazamiento de las bandas por unidad de deformación. Utilizando este hamiltoniano en (4.7) se puede llegar a la siguiente expresión para la probabilidad de dispersión con fonones acústicos, incluyendo los procesos de emisión y absorción [Tom93]:

$$W(\mathbf{k}) = \frac{\sqrt{2}(m^*)^{3/2} k_B T \Xi_d^2}{\pi \hbar^4 c_l} \sqrt{E_{\mathbf{k}}} (1 + 2\alpha E_{\mathbf{k}})(1 + \alpha E_{\mathbf{k}})^{1/2}, \quad (4.20)$$

donde c_l es la constante elástica del material.

En el caso de los fonones ópticos el hamiltoniano de la perturbación se supone proporcional al desplazamiento, y no a su derivada, al orden más bajo de aproximación. Aunque la suposición de medio continuo utilizada en el caso de los fonones acústicos para longitudes de onda largas no es consistente, el resultado final obtenido con la teoría correcta es el mismo en el caso de suponer una longitud de onda igual a la constante de red para reflejar el hecho de que átomos en la misma celda unidad vibran en sentidos opuestos. Además, teniendo en cuenta que la energía de los fonones ópticos es prácticamente independiente del vector de onda, la energía y el número medio de fonones se asumen constantes ($\hbar\omega_0$ y n_0) [JL89]. Con estas aproximaciones podemos

llegar a la siguiente expresión para la probabilidad de dispersión en el caso de dispersiones intravalle:

$$W(\mathbf{k}) = \frac{D_0^2(m^*)^{3/2}}{\sqrt{2\pi\hbar^3\rho\omega_0}} \binom{n_0}{n_0+1} \sqrt{E_{\mathbf{k}} \pm \hbar\omega_0} \cdot \sqrt{1 + \alpha(E_{\mathbf{k}} \pm \hbar\omega_0)}(1 + 2\alpha(E_{\mathbf{k}} \pm \hbar\omega_0)), \quad (4.21)$$

donde D_0 es la constante óptica del potencial de deformación, ρ la densidad del cristal y los símbolos superior e inferior hacen referencia a los procesos de absorción y emisión respectivamente. En el caso de semiconductores cúbicos, como el silicio, esta dispersión está restringida a los valles L , estando prohibida para los X y los Γ . Además, los fonones ópticos no polares también pueden inducir transiciones electrónicas entre valles diferentes. En este caso las probabilidades de transición son equivalentes a la de transiciones intravalle, sólo que también hay que considerar un cambio en la energía por el cambio de valle, ΔE_v , y además aparece un factor de multiplicidad γ que tiene en cuenta los valles equivalentes finales:

$$W(\mathbf{k}) = \frac{D_0^2(m^*)^{3/2}}{\sqrt{2\pi\hbar^3\rho\omega_0}} \binom{n_0}{n_0+1} \gamma \sqrt{E_{\mathbf{k}} \pm \hbar\omega_0 - \Delta E_v} \cdot \sqrt{1 + \alpha(E_{\mathbf{k}} \pm \hbar\omega_0 - \Delta E_v)}(1 + 2\alpha(E_{\mathbf{k}} \pm \hbar\omega_0 - \Delta E_v)). \quad (4.22)$$

Para todos los tipos de dispersión por fonones que consideramos aquí, la dispersión es isotropa, por lo que podemos calcular fácilmente la dirección del electrón tras la dispersión con dos números aleatorios uniformes $r_1, r_2 \in [0, 1)$:

$$\phi = 2\pi r_1, \quad (4.23)$$

$$\cos \theta = 2r_2 - 1. \quad (4.24)$$

Dispersión por impurezas ionizadas

Los portadores en un dispositivo semiconductor habitualmente tienen que atravesar zonas dopadas en las que las impurezas ionizadas afectan notablemente al transporte a través de interacciones de tipo coulombiano. Sin embargo, el potencial electrostático en el material semiconductor está apantallado por la presencia de portadores libres. Existen dos formas principales de tratar este apantallamiento: las aproximaciones de Conwell-Weisskopf y de Brooks-Herring [Tom93]. En la primera aproximación se utiliza un potencial

coulombiano con un parámetro de corte para las distancias menores que la distancia media entre dopantes, mientras que en la segunda aproximación se usa un potencial apantallado de la forma $V(r) = C \exp(-r/L_D)$. En este caso la probabilidad de dispersión que se obtiene es [Tom93]:

$$W_{BH}(\mathbf{k}) = \frac{q^4 N}{\sqrt{m^*} 2\sqrt{2\pi}\varepsilon^2 E_\beta^2} \frac{\sqrt{E_{\mathbf{k}}(1 + \alpha E_{\mathbf{k}})}(1 + 2\alpha E_{\mathbf{k}})}{1 + 4E_{\mathbf{k}} \frac{1 + \alpha E_{\mathbf{k}}}{E_\beta}}, \quad (4.25)$$

donde N es el dopado y E_β es una energía relacionada con la longitud de apantallamiento β_s [CQ81]:

$$E_\beta = \frac{\hbar^2}{2m^*} \beta_s^2 = \frac{q^2 \hbar^2 n}{2\varepsilon m^* k_B T} \frac{F_{-1/2}(\eta_c)}{F_{1/2}(\eta_c)}. \quad (4.26)$$

Más adelante, Ridley [Rid77] introdujo lo que es un puente entre las dos aproximaciones. Para este modelo se obtiene la siguiente probabilidad de dispersión [dRW86]:

$$W(\mathbf{k}) = v \left[1 - \exp\left(-\frac{W_{BH}}{v}\right) \right], \quad (4.27)$$

con v dado por

$$v = a_I \sqrt{\frac{2}{m^*} \frac{\sqrt{E_{\mathbf{k}}(1 + \alpha E_{\mathbf{k}})}}{1 + 2\alpha E_{\mathbf{k}}}}, \quad (4.28)$$

donde a_I da la inversa de la distancia media entre impurezas ionizadas, $a_I = (2\pi N)^{1/3}$.

Este mecanismo de dispersión es anisótropo, por lo que el cálculo de la dirección final es algo diferente al caso de la dispersión por fonones. Los ángulos que definen la dirección final en este caso los calculamos usando el método directo descrito en [dRW86]:

$$\phi = 2\pi \cdot r_{nd1}, \quad (4.29)$$

$$\cos \theta = 1 - 2 \frac{(W_{BH} - r_3)}{W_{BH} + 4r_3 E_{\mathbf{k}}(1 + \alpha E_{\mathbf{k}})/E_\beta}, \quad (4.30)$$

con

$$r_3 = -r_1 \ln [r_2 + (1 - r_2) \exp(-W_{BH}/r_1)], \quad (4.31)$$

$$r_1 = a_I \sqrt{2/m^*} \sqrt{E_{\mathbf{k}}(1 + \alpha E_{\mathbf{k}})}/(1 + 2\alpha E_{\mathbf{k}}), \quad (4.32)$$

$$r_2 = 1 - r_{nd2}, \quad (4.33)$$

donde usamos dos número aleatorios uniformes $r_{nd1}, r_{nd2} \in [0, 1)$.

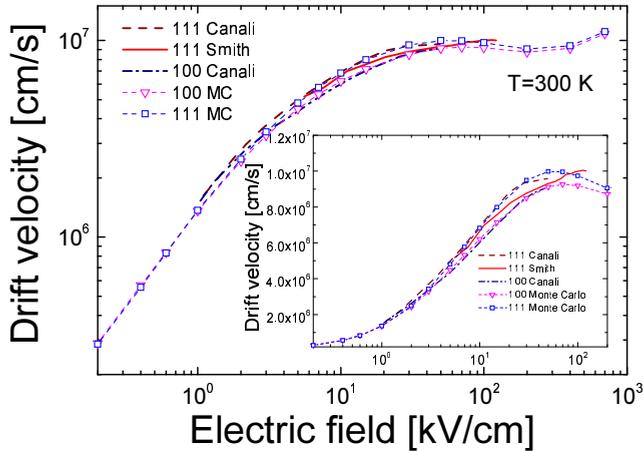


Figura 4.7: Velocidad media de los electrones frente al campo eléctrico para el silicio en las direcciones $\langle 100 \rangle$ y $\langle 111 \rangle$ comparadas con datos experimentales en escalas logarítmica y lineal.

Dispersión por rugosidad superficial

Por último, consideramos el efecto que la rugosidad de las interfaces entre óxido y silicio tienen sobre el transporte en dispositivos. Una de las formas más habituales de incluir su efecto en simulaciones semiclásicas sin correcciones cuánticas es el uso de unas condiciones de contorno específicas en las interfaces entre los dos materiales. En este caso cuando una partícula colisiona con la interfaz existe una cierta probabilidad de que no sea reflejado de forma especular, sino que se añade una componente de aleatoriedad a la dirección de salida de la reflexión. Sin embargo, este modelo no es apropiado cuando se introducen correcciones cuánticas, ya que el número de colisiones es muy bajo al aparecer un potencial cuántico repulsivo. En este caso es posible añadir la dispersión por rugosidad superficial como un mecanismo de dispersión adicional [RF03]. El modelo original fue desarrollado para transistores de una puerta, y aunque hay trabajos sobre reformulación para otras arquitecturas [GRCC⁺01, JFT07b, JFT07a], por el momento no hemos introducido el modelo en las simulaciones de transistores MOSFET.

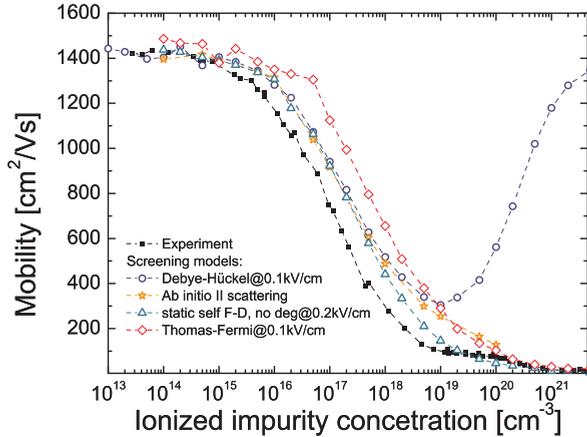


Figura 4.8: Movilidad del silicio para diferentes concentraciones de impurezas comparadas con datos experimentales (cuadrados) y simulaciones con la dispersión por impurezas ionizadas introducida *ab initio* (estrellas) [ARA08]. Los datos con los modelos de apantallamiento se corresponden con: modelo de Debye-Hückel (límite no-degenerado, círculos), modelo de Thomas-Fermi (límite degenerado, rombos) y modelo estático descrito en el texto (triángulos).

4.4.3. Resultados

Utilizando el modelo analítico de bandas y los mecanismos de dispersión anteriormente descritos realizamos la simulación de transporte en silicio para validar los resultados frente a medidas experimentales. En primer lugar mostramos, en la figura 4.7, las curvas velocidad-campo para el silicio para dos direcciones de transporte, la $\langle 100 \rangle$ y la $\langle 111 \rangle$. Podemos ver un buen acuerdo con las medidas experimentales en las dos direcciones para el rango de valores del campo eléctrico simulados.

Además, también hemos simulado la movilidad para diferentes valores de la concentración de impurezas ionizadas y usando tres modelos para el apantallamiento, como se puede ver en la figura 4.8. El primero de ellos, el de Debye-Hückel se obtiene como límite en el caso de no degeneración de la expresión que usamos para la longitud inversa de apantallamiento (4.26):

$$\beta_s^2 = \frac{q^2 n}{\epsilon k_B T} \frac{F_{-1/2}(\eta_c)}{F_{1/2}(\eta_c)} \rightarrow \frac{q^2 n}{\epsilon k_B T}. \quad (4.34)$$

Vemos que esta aproximación funciona bien hasta valores de la concentración de impurezas ionizadas de hasta 10^{17} - 10^{18} cm^{-3} pero después lleva a un aumento excesivo que no se ve en los experimentos.

Otro límite que mostramos es el de alta degeneración, que lleva al modelo de apantallamiento de Thomas-Fermi:

$$\beta_s^2 = \frac{q^2 n}{\varepsilon k_B T} \frac{F_{-1/2}(\eta_c)}{F_{1/2}(\eta_c)} \rightarrow \frac{q^2 m^*}{\pi^2 \varepsilon \hbar^2} (3\pi^2 n)^{1/3}. \quad (4.35)$$

En este caso la movilidad se sobreestima más que con los otros modelos para concentraciones de impurezas bajas, pero el comportamiento a concentraciones altas corrige los problemas que tiene la aproximación anterior.

Pese a que su cálculo es el más costoso, vemos que para cubrir todo el rango de concentraciones de impurezas ionizadas es necesario usar la expresión original:

$$\beta_s^2 = \frac{q^2 n}{\varepsilon k_B T} \frac{F_{-1/2}(\eta_c)}{F_{1/2}(\eta_c)}. \quad (4.36)$$

4.5. Simulación de dispositivos: uso de mallas tetraédricas

El uso de mallas tetraédricas en la simulación Monte Carlo implica algunas complicaciones o cambios con respecto a los algoritmos utilizados en mallas cartesianas. En esta sección comentamos algunos de estos cambios. En primer lugar describimos la implementación de las correcciones cuánticas, aunque el cambio principal proviene de la discretización de elementos finitos necesaria descrita en el capítulo 3. A continuación describimos los procesos de asignación de carga e interpolación del campo eléctrico, especialmente importantes porque su correcta elección permite minimizar las autofuerzas sobre las partículas. Finalmente describimos los procesos de inyección de carga e integración de las ecuaciones del movimiento, que, pese a tener la misma forma que los usados en mallas cartesianas, tienen algunas particularidades asociadas con el uso de mallas tetraédricas.

4.5.1. Correcciones cuánticas

Para reflejar los efectos del confinamiento cuántico en las simulaciones Monte Carlo hemos implementado el modelo *density gradient* descrito en el

capítulo 3. Al resolver las ecuaciones de este modelo podemos obtener un potencial cuántico V_Q que utilizamos para corregir la fuerza sobre las partículas. El uso de la discretización basada en potenciales en lugar de en densidades de portadores permite una solución más suave, especialmente cuando se usa de forma autoconsistente [wTW04, SMGGR06], ya que las simulaciones Monte Carlo producen una densidad muy ruidosa. Sin embargo, inicialmente sólo hemos implementado la corrección sin autoconsistencia, utilizando el resultado del modelo arrastre-difusión con *density gradient* para obtener un valor de V_Q que mantenemos fijo a lo largo de toda la simulación Monte Carlo para la polarización dada. Usando este potencial calculamos el campo en la posición de las partículas como

$$\mathbf{E} = -\nabla(V_{SC} + V_Q^{FF}), \quad (4.37)$$

donde V_Q^{FF} es el potencial cuántico procedente de la solución inicial y V_{SC} el potencial electrostático que se obtiene en cada iteración del Monte Carlo. Pese a la falta de autoconsistencia en la corrección cuántica, en [RA08] se muestra, para un voltaje de drenador bajo, que los resultados obtenidos con esta aproximación son muy similares a los obtenidos con una aproximación totalmente autoconsistente. Finalmente comentar una ventaja numérica que nos proporciona el uso de correcciones cuánticas en la simulación. La aparición de una fuerza que repele a las partículas de la interfaz hace menos frecuentes las colisiones con la interfaz, lo que supone una reducción considerable de operaciones y una mayor estabilidad numérica del código, especialmente cuando usamos dominios no rectangulares, donde estas operaciones son más costosas.

Vemos un ejemplo del uso de este algoritmo para la corrección cuántica en la figura 4.9, donde mostramos la concentración media de portadores en dos secciones transversales de un TriGate MOSFET de 40 nm de longitud de puerta [KDD⁺06] a una polarización de drenador de 0,05 V y de puerta de 1,1 V. El efecto de la corrección cuántica queda reflejado en el vaciamiento de portadores en la interfaz óxido-silicio, tanto en las zonas de fuente y drenador como bajo los tres lados cubiertos por la puerta.

4.5.2. Autofuerzas: asignación de carga e interpolación del campo eléctrico

En simulaciones que utilizan partículas y que además deben estar acopladas de forma autoconsistente a una malla del dominio de simulación pueden aparecer fuerzas no físicas que surgen de este acoplamiento. Por ejemplo, en

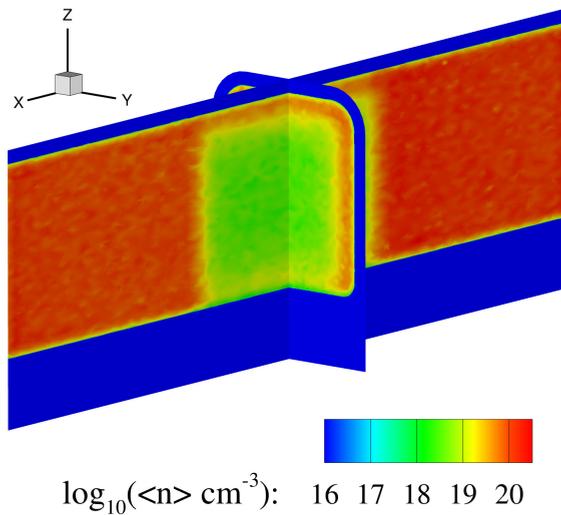


Figura 4.9: Concentración media de portadores en dos secciones transversales de un TriGate MOSFET de 40 nm de puerta ($V_D=0,05$ V y $V_G=1,1$ V) usando un potencial cuántico constante en toda la simulación.

nuestro caso las cargas deben ser asignadas a los nodos circundantes para resolver la ecuación de Poisson. Una vez resuelta podemos obtener los campos eléctricos sobre los nodos de la malla, de forma que debemos hacer otro proceso de interpolación a la posición de la partícula para conocer la fuerza sobre ella. En el primer proceso de mapeo de magnitudes del espacio “real” en el que se mueven las partículas al espacio discretizado en el que se resuelven las ecuaciones de campo se pierde resolución espacial que se trata de recuperar después mediante interpolación. Veremos que este proceso puede ser hecho de tal forma que no produzca estas fuerzas falsas. Sin embargo, la irregularidad de la malla sobre la que se resuelven las ecuaciones provoca que se pierda la simetría esférica en la solución de la ecuación de Poisson para una carga puntual, lo que hace que el proceso de interpolación final del campo no produzca fuerza cero en la posición original de la partícula aunque se resolviera sobre un dominio discretizado infinito sin interferencia de condiciones de contorno.

El impacto de las autofuerzas en simulaciones que usan partículas ha sido muy estudiado [HE88, Lau96] y se han propuesto varios métodos para minimizarlas. Sin embargo, la mayoría de estos métodos sólo obtienen resultados satisfactorios para mallas ortogonales [Lau96]. Formalmente, se ha demostrado

que las autofuerzas son nulas para mallas ortogonales uniformes si la asignación de carga y el esquema de interpolación de fuerzas son equivalentes [HE88]. Sin embargo, es importante decir que esto no es una condición necesaria como se demuestra en [Lau96], donde se presentan varios esquemas para obtener autofuerza cero.

En dispositivos reales tenemos condiciones de contorno e inhomogeneidades, así que la fuerza sobre una partícula sola no tiene que ser cero. Sin embargo, aunque la falta de resolución espacial para reflejar estas inhomogeneidades también llevará a fuerzas incorrectas sobre la partícula, esto se podría remediar usando una malla más fina. Esto hace este efecto diferente de las autofuerzas, que no pueden ser eliminadas refinando la malla.

El problema de las autofuerzas ha sido especialmente problemático con mallas no estructuradas, y pocos resultados positivos se han obtenido en este caso. En [Lau96], el tratamiento de las autofuerzas sobre mallas no estructuradas asume que se conoce el potencial electrostático en los nodos y que de él se puede obtener únicamente un valor constante del campo eléctrico en cada elemento. Con esta suposición, se demuestra que sólo es posible obtener autofuerza cero si los elementos son equiláteros. Esto se podría hacer asignando la misma cantidad de carga de cada partícula a todos los nodos del elemento. Sin embargo, es claro que la aplicación de este resultado a dispositivos reales es muy limitada. Sin embargo, si obtenemos el valor del campo eléctrico en los nodos de la malla, podemos definir una asignación de carga y un esquema de interpolación del campo eléctrico de tal forma que la autofuerza fuera cero. En lo que sigue asumiremos que conocemos el campo eléctrico sobre los nodos de la malla, $\mathbf{E}(\mathbf{r}_p)$. Más adelante volveremos sobre el problema de obtener estos valores.

En [HE88], se utilizan dos condiciones para demostrar que la autofuerza es cero:

1. *La asignación de carga y el esquema de interpolación de fuerzas deben ser iguales.* Sea $\phi(\mathbf{r} - \mathbf{r}_p)$ la función de interpolación asociada con el nodo p del elemento considerado en el punto \mathbf{r} . La contribución a la carga en el nodo \mathbf{r}_p de una carga q_k en \mathbf{r} será

$$q(\mathbf{r}_p) = q_k \phi(\mathbf{r} - \mathbf{r}_p), \quad (4.38)$$

Por otro lado, el campo eléctrico en \mathbf{r} será

$$\mathbf{E}(\mathbf{r}) = \sum_{p \in \text{nodos}} \phi(\mathbf{r} - \mathbf{r}_p) \mathbf{E}(\mathbf{r}_p). \quad (4.39)$$

2. *El campo eléctrico definido sobre los nodos de la malla debe ser esféricamente simétrico.* El campo eléctrico en un nodo \mathbf{r}_j generado por una carga en el nodo \mathbf{r}_i debe tener la misma magnitud y sentido opuesto al campo eléctrico en el nodo \mathbf{r}_j generado por la misma carga en el nodo \mathbf{r}_i . Asumiendo que hay una dependencia lineal entre el campo eléctrico y la carga,

$$\mathbf{E}(\mathbf{r}_p) = \sum_{p' \in \text{nodes}} G(\mathbf{r}_p; \mathbf{r}_{p'}) q(\mathbf{r}_{p'}), \quad (4.40)$$

donde la matriz de proporcionalidad deber ser antisimétrica:

$$G(\mathbf{r}_p; \mathbf{r}_{p'}) = -G(\mathbf{r}_{p'}; \mathbf{r}_p). \quad (4.41)$$

Condición 1: esquemas de asignación de carga e interpolación de campo eléctrico

Una vez obtenido el campo eléctrico en los nodos de la malla es fácil hacer que se verifique la condición 1. Para ello usamos las funciones de interpolación lineales del método de elementos finitos sobre tetraedros que usamos en la discretización de la ecuación de Poisson. Para un elemento e , las funciones asociadas a cada uno de sus nodos $p \in e$ [BCO81] son:

$$\begin{cases} \phi(\mathbf{r} - \mathbf{r}_p) = \phi_p^e(\mathbf{r}) = \frac{v_p}{V_e} & \mathbf{r} \in \Omega_e \\ \phi(\mathbf{r} - \mathbf{r}_p) = \phi_p^e(\mathbf{r}) = 0 & \text{en otro caso,} \end{cases} \quad (4.42)$$

donde V_e es el volumen del elemento e que contiene al punto \mathbf{r} y v_p el volumen del subtetraedro formado por este punto y los tres nodos de la cara opuesta a p .

El uso de esta interpolación para la asignación de carga surge de forma natural al discretizar la ecuación de Poisson usando cargas puntuales. En primer lugar introducimos la densidad de carga obtenida a partir de la posición de las partículas en la formulación débil de la ecuación de Poisson. Para ello expresamos la densidad de electrones en el espacio como:

$$n = \sum_{k \in p} q_k \delta(\mathbf{r}_k). \quad (4.43)$$

Sustituyendo en (3.113) obtenemos, para el término en n ,

$$\int_{\Omega} n v \, d\Omega = \int_{\Omega} \sum_{k \in p} q_k \delta(\mathbf{r}_k) v \, d\Omega. \quad (4.44)$$

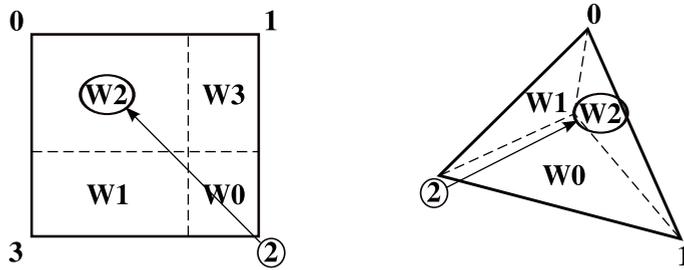


Figura 4.10: Interpretación geométrica de la asignación de carga para un elemento cartesiano (izquierda) y uno triangular (derecha). W_i es el peso asociado al nodo i y corresponde al área del subelemento opuesto al nodo. Los subelementos están definidos por un punto que sería la posición de la carga.

Usando la aproximación de Galerkin para v , $v = \sum v_i \theta_i$, quedándonos sólo con el término i -ésimo de la suma sobre i y pasando a trabajar con funciones de forma, obtenemos la contribución de la densidad de electrones a (3.116),

$$\int_{\Omega} \sum_{k \in p} q_k \delta(\mathbf{r}_k) \theta_i d\Omega = \int_{\Omega} \sum_{k \in p} q_k \delta(\mathbf{r}_k) \sum_{e \ni i} \phi_i^e d\Omega = \sum_{k \in p} \sum_{e \ni i} q_k \phi_i^e(\mathbf{r}_k). \quad (4.45)$$

En la figura 4.10 se muestra un ejemplo en dos dimensiones de esta asignación de carga y se compara con la asignación CIC (*cloud-in-cell*) estándar usada en mallas ortogonales.

Condición 2: antisimetría del campo eléctrico

La condición 2 se verifica si el campo eléctrico en los nodos es exacto. Sin embargo, los valores serán aproximaciones y no satisfarán esta condición perfectamente. Esto es una fuente de autofuerzas en el dispositivo con una magnitud dependiente de la precisión del valor del campo eléctrico.

Con estas condiciones, usando (4.38), (4.39) y (4.40), el campo eléctrico en un punto \mathbf{r} en un dominio con una única partícula k se puede escribir como:

$$\mathbf{E}_k(\mathbf{r}) = \sum_{p=1}^s \sum_{p'=1}^s q_k \phi(\mathbf{r} - \mathbf{r}_p) G(\mathbf{r}_p; \mathbf{r}_{p'}) \phi(\mathbf{r} - \mathbf{r}_{p'}). \quad (4.46)$$

Si se verifica la condición (4.41), entonces esta expresión es idénticamente nula. Asumiendo que el campo eléctrico es lineal, $\mathbf{E}(\mathbf{r}) = \sum_i \mathbf{E}_i(\mathbf{r})$, las autofuerzas serán cero en todo el dominio. Sin embargo, en general la condición

no se verificará de forma exacta y el campo no será nulo. La magnitud de la autofuerza dependerá, por tanto, de lo próximo que esté el campo eléctrico a la antisimetría.

Por tanto, una vez definidos los esquemas de asignación de carga y campo eléctrico, vemos que probablemente el paso más crítico es el de la obtención de un campo que verifique la propiedad de antisimetría. En principio, la forma más consistente de calcular el campo eléctrico consistiría en formular el problema variacional y resolverlo usando el método de elementos finitos. Partiendo de la definición,

$$\mathbf{E} = -\nabla\psi, \tag{4.47}$$

podemos escribir la forma variacional:

$$\int_{\Omega} (\mathbf{E} + \nabla\psi) v \, d\Omega = 0. \tag{4.48}$$

Usando la aproximación de Galerkin,

$$\int_{\Omega} \left(\sum_{j=1}^N \mathbf{E}_j \theta_j + \sum_{j=1}^N \psi_j \nabla \theta_j \right) \sum_{i=1}^N v_i \theta_i \, d\Omega = 0. \tag{4.49}$$

Debido a la arbitrariedad de v , esto resulta en el siguiente sistema de ecuaciones vectorial:

$$\sum_{j=1}^N \left[\mathbf{E}_j \int_{\Omega} \theta_j \theta_i \, d\Omega + \psi_j \int_{\Omega} \nabla \theta_j \theta_i \, d\Omega \right] = 0 \quad \forall i = 1, \dots, N. \tag{4.50}$$

Por tanto, usando este método deberíamos resolver tres sistemas de ecuaciones adicionales por iteración, además de la ecuación de Poisson. Otra forma más económica de obtener el campo eléctrico en los nodos de la malla consiste en obtener el valor en cada elemento a partir del valor del potencial en sus nodos y después hacer una interpolación a los nodos. Para un elemento e , el campo eléctrico se obtendría como

$$\mathbf{E}_e = -\nabla\psi_e = -\nabla \left(\sum_{i \in e} \psi_i \theta_i \right) = - \sum_{i \in e} \psi_i \nabla \theta_i. \tag{4.51}$$

A partir de estos valores obtendríamos el campo eléctrico en los nodos como

$$\mathbf{E}(\mathbf{r}_i) = \sum_{e \ni i} w_e \mathbf{E}_e. \tag{4.52}$$

Al probar distintas posibilidades para los pesos w_e encontramos que había una dependencia débil con la elección. Aún así, probando con potenciales analíticos comprobamos que el menor error lo obteníamos usando la siguiente interpolación:

$$\mathbf{E}(\mathbf{r}_i) = \sum_j \frac{A}{d_e} \mathbf{E}_e, \quad (4.53)$$

donde la suma es sobre todos los tetraedros vecinos y d_j la distancia entre el nodo \mathbf{r}_i al baricentro del tetraedro vecino e . La constante A se elige de tal forma que $\sum \frac{A}{d_e} = 1$ para cada nodo, así que $A = \left(\sum \frac{1}{d_e}\right)^{-1}$.

Una vez definidos los métodos para interpolar el campo eléctrico en los nodos, evaluamos su calidad en función de la proximidad a la antisimetría. Para ello resolvemos la ecuación de Poisson en un dominio cúbico con densidad de carga diferente de cero en un único nodo de tal forma que las fronteras estén lo suficientemente lejos para que las condiciones de contorno afecten poco a la solución en las proximidades de la carga. Repetimos este cálculo para los cuatro nodos de un mismo elemento. Con estos datos comprobamos los campos eléctricos en cada nodo originados por cada una de las cuatro localizaciones de la carga. En el caso ideal la matriz resultante debería ser antisimétrica. En primer lugar utilizamos una malla tetraédrica estructurada. En este caso, para las componentes x de los campos eléctricos obtuvimos la siguiente matriz (normalizada al valor más elevado):

$$G^x \simeq \begin{pmatrix} 4,1 \cdot 10^{-8} & -0,22826 & -0,32240 & -0,99933 \\ 0,22827 & 1,1 \cdot 10^{-5} & -0,16331 & -0,51641 \\ 0,32241 & 0,16333 & 1,1 \cdot 10^{-5} & -0,52093 \\ 1,00000 & 0,51709 & 0,52161 & 6,8 \cdot 10^{-4} \end{pmatrix}$$

Podemos ver que aunque no sea antisimétrica, la desviación está próxima al 0.1%, por lo que el uso de este tipo de interpolación es suficiente.

Después de esto hicimos una prueba sobre una malla no estructurada. En este caso obtenemos la siguiente matriz para la componente x del campo eléctrico:

$$G^x \simeq \begin{pmatrix} 0,01719 & -0,70076 & -0,14044 & 0,05078 \\ 0,70327 & -0,11414 & 0,72234 & 1,00000 \\ 0,20840 & -0,77500 & 0,03204 & 0,31582 \\ -0,01774 & -0,99745 & -0,32233 & 0,03744 \end{pmatrix}$$

Podemos ver que en este caso la matriz está más lejos de la antisimetría, con una desviación de hasta cerca del 100 % en uno de los elementos (G_{41}^x, G_{14}^x). En este caso también calculamos el campo usando (4.50) para comprobar si mejora el comportamiento. La matriz que se obtiene es:

$$G^x \simeq \begin{pmatrix} -0,00785 & -0,69734 & -0,04820 & 0,06307 \\ 0,67289 & -0,06546 & 0,79161 & 0,94406 \\ 0,19568 & -0,84617 & 0,09457 & 0,36391 \\ -0,06996 & -1,00000 & -0,26227 & -0,01267 \end{pmatrix}$$

Se puede ver que el comportamiento es similar y se caracteriza por la misma irregularidad.

Aunque el valor del campo calculado sea algo más preciso, el coste de la resolución de los sistemas de ecuaciones no compensa la mejoría en las propiedades del campo con respecto a la antisimetría.

4.5.3. Inyección de portadores sobre los contactos óhmicos

Otro proceso que requiere adaptación a mallas tetraédricas es la inyección de portadores en los contactos. En este caso hay que clasificar los tetraedros frontera en función del número de nodos sobre el contacto. Sólo los que tengan una cara sobre la superficie del contacto pueden ser utilizados para la inyección de partículas, pero todos contribuyen al cálculo de la carga total adyacente al contacto. El número total de partículas que deben ser inyectadas se calcula mediante la diferencia entre la carga fija (dopantes ionizados) en el volumen adyacente al contacto y la carga de las superpartículas en este mismo volumen. La inyección de estas partículas se realiza de acuerdo a la densidad de carga obtenida en la solución de las ecuaciones de arrastre difusión. Para esto se emplea un método equivalente al utilizado en la selección del mecanismo de dispersión. En primer lugar, a cada triángulo sobre el contacto se le asigna una probabilidad proporcional a su área y su densidad de carga. Después, se genera un número aleatorio y se compara con los productos normalizados *área·densidad* de los triángulos:

$$r < \sum_{i=1}^j \frac{a_i n_i}{AC}, \quad j = 1, \dots, n,$$

donde a_i es el área del triángulo i , AC es el producto *área·densidad* total, $AC = \sum_{i=1}^n a_i n_i$, y n es el número de triángulos en el contacto. Una vez

seleccionado el triángulo, se escoge una posición aleatoria sobre él:

$$\begin{cases} \mathbf{r} = \mathbf{r}_0 + r_1 \cdot \mathbf{r}_{01} + r_2 \cdot \mathbf{r}_{02} \\ r_1 + r_2 \leq 1 \\ r_1, r_2 \geq 0 \end{cases}$$

donde \mathbf{r}_0 , \mathbf{r}_1 y \mathbf{r}_2 son las posiciones de los nodos del triángulo, r_0 y r_1 dos números aleatorios y \mathbf{r}_{ij} el vector $\mathbf{r}_j - \mathbf{r}_i$.

4.5.4. Integración de las ecuaciones del movimiento

El movimiento de las partículas en un campo externo (el campo eléctrico \mathbf{E} en nuestro caso) está gobernado por las ecuaciones del movimiento. La trayectoria de la partícula puede ser obtenida integrando estas ecuaciones, lo que es un proceso, en principio, independiente de la malla. Sin embargo, la necesidad de interpolar el campo en la posición de las partículas hace que este proceso presente más complejidades en el caso del uso de tetraedros que en el caso de mallas hexaédricas cartesianas.

En la primera parte de la sección mostraremos las ecuaciones que debemos resolver. A continuación comentaremos las dificultades específicas que presenta el uso de mallas tetraédricas.

Las ecuaciones que debemos resolver en los vuelos libres en el modelo semiclásico son:

$$\mathbf{v} = \frac{d\mathbf{r}}{dt} = \frac{1}{\hbar} \nabla_{\mathbf{k}} \varepsilon(\mathbf{k}) \quad (4.54)$$

y

$$\frac{d\mathbf{k}}{dt} = \frac{-e\mathbf{E}}{\hbar}, \quad (4.55)$$

donde $\varepsilon(\mathbf{k})$ está dado por la estructura de bandas del material. En nuestro caso estamos usando un modelo de bandas analítico anisótropo y no parabólico:

$$\varepsilon(\mathbf{k})(1 + \alpha\varepsilon(\mathbf{k})) = \frac{\hbar^2}{2} \mathbf{k} \mathbf{M}^{-1} \mathbf{k} = \gamma = \frac{\hbar^2 |\mathbf{k}^s|^2}{2m^*}, \quad (4.56)$$

donde \mathbf{M} es el tensor de masa efectiva. La masa efectiva m^* la introducimos mediante las transformaciones de Herring-Vogt [SS83]. Por ejemplo, para los valles X, los más importantes para el transporte en el silicio,

$$k_a^s = \sqrt{\frac{m^*}{m_a}} k_a, \quad a = x, y, z. \quad (4.57)$$

La simetría del tensor de masa efectiva inverso \mathbf{M}^{-1} lleva a la siguiente ecuación para la trayectoria de las partículas:

$$\frac{d\mathbf{r}}{dt} = \frac{\hbar\mathbf{M}^{-1}\mathbf{k}}{\sqrt{1+4\alpha\gamma}} = \frac{\hbar\mathbf{k}^*}{\sqrt{1+4\alpha\gamma}}. \quad (4.58)$$

Nuevamente para el valle X, tenemos

$$k_a^* = \frac{k_a}{m_a}, \quad a = x, y, z. \quad (4.59)$$

Como el campo eléctrico debe ser constante en cada vuelo libre entre dos soluciones consecutivas de la ecuación de Poisson para evitar las autofuerzas,

$$\mathbf{k}^*(t) = \mathbf{k}^*(0) - \frac{q\mathbf{E}^*}{\hbar}t \quad (4.60)$$

y

$$\mathbf{r}(t) = \mathbf{r}(0) + \hbar \int_0^t \frac{\mathbf{k}^*(0) - \frac{q\mathbf{E}^*}{\hbar}t}{\sqrt{1+4\alpha\gamma(t)}} dt. \quad (4.61)$$

La dependencia temporal de γ está dada por

$$\gamma = \hbar^2 \frac{\mathbf{k}^s \cdot \mathbf{k}^s}{2m^*} = \frac{\hbar^2}{2m^*} \left[|\mathbf{k}^s(0)|^2 - 2\frac{q}{\hbar}\mathbf{E}^s \cdot \mathbf{k}^s(0)t + \frac{q^2}{\hbar^2}|\mathbf{E}^s|^2 t^2 \right]. \quad (4.62)$$

En este punto podemos hacer una aproximación permitiendo que γ sea constante durante los vuelos libres. Más adelante comentaremos esta aproximación, pero por el momento continuamos incluyendo la dependencia temporal de γ . En este caso tenemos que resolver la siguiente integral:

$$\int_0^t \frac{a+bt}{\sqrt{c+dt+et^2}} dt, \quad (4.63)$$

donde a, b, c, d y e son constantes. Esta integral se puede resolver analíticamente:

$$\begin{aligned} \int_0^t \frac{a+bt}{\sqrt{c+dt+et^2}} dt &= \frac{b}{e} \sqrt{c+dt+et^2} + \\ &+ \left(\frac{a}{\sqrt{e}} - \frac{bd}{2e\sqrt{e}} \right) \operatorname{arcsenh} \frac{2et+d}{\sqrt{4ec-d^2}} - \\ &- \frac{b}{e} \sqrt{c} - \left(\frac{a}{\sqrt{e}} - \frac{bd}{2e\sqrt{e}} \right) \operatorname{arcsenh} \frac{d}{\sqrt{4ec-d^2}}. \end{aligned} \quad (4.64)$$

Sin embargo, resolver esta ecuación para encontrar la posición de la partícula al final del vuelo libre puede ser demasiado costoso. Como comentamos anteriormente, para simplificar los cálculos aproximamos las trayectoria de la partícula usando un valor de γ constante durante cada vuelo libre. En este caso la integral es

$$\int_0^t \frac{a + bt}{\sqrt{1 + 4\alpha\gamma}} dt = \frac{at + \frac{bt^2}{2}}{\sqrt{1 + 4\alpha\gamma}} \quad (4.65)$$

y la trayectoria de la partícula es una parábola:

$$\mathbf{r}(t) = \mathbf{r}(0) + \frac{\hbar\mathbf{k}^*(0)}{\sqrt{1 + 4\alpha\gamma}}t - \frac{q\mathbf{E}^*}{2\sqrt{1 + 4\alpha\gamma}}t^2. \quad (4.66)$$

Esto es equivalente al resultado obtenido usando un modelo de bandas parabólico pero con un valor de la masa modificado por la energía del electrón. La figura 4.11 muestra el valor de la integral para los tres casos: aproximación de bandas no parabólicas, aproximación de bandas no parabólicas con γ constante durante los vuelos libres y, por último, aproximación de bandas parabólicas. Además, también mostramos el error en los dos últimos casos cuando los comparamos con el primero. Los valores de la gráfica se obtuvieron con las siguientes condiciones iniciales: $\mathbf{k} = (10^8, 0, 0)$ [m^{-1}] y $\mathbf{E}^* = (10^8, 0, 0)$ [V/m]. Sin embargo, la diferencia entre ambas aproximaciones aumenta para diferentes condiciones iniciales, por ejemplo, en el caso de empezar con un campo eléctrico más bajo como el que se encontraría en las regiones de fuente o drenador.

Para terminar la sección comentaremos las particularidades de la implementación de los vuelos libres sobre mallas tetraédricas. En primer lugar tenemos que considerar la identificación del elemento que contiene a la partícula. En el caso de mallas hexaédricas cartesianas este proceso consiste únicamente en la comparación de las coordenadas de las partículas con las de los planos de los elementos. Sin embargo, en el caso de mallas tetraédricas el proceso no es tan directo. Podemos considerar dos opciones principales: hacer comparaciones en el espacio sin transformar con planos de orientaciones arbitrarias o hacer una transformación a un espacio en el que las comparaciones sean más directas. En cualquiera de los dos casos hacen falta más operaciones que con mallas cartesianas, lo que resulta en un proceso no sólo más lento, sino también con una precisión menor. Afortunadamente este es un proceso que no es necesario realizar muchas veces durante la ejecución del código. El proceso que presenta más dificultades es el de la determinación del tiempo de vuelo

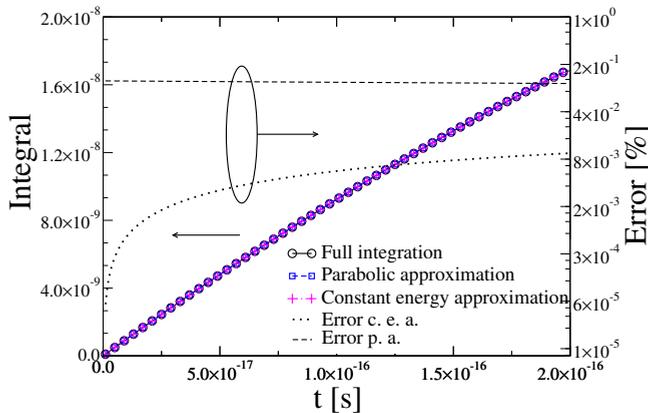


Figura 4.11: Valor de la integral usada en el cálculo de las trayectorias con un paso temporal de 0.2 fs. Para las aproximaciones parabólica y de γ constante también se muestran los errores.

hasta que la partícula alcanza una frontera del elemento en el que está. En el caso de mallas tetraédricas, aunque sólo hace falta comprobar el corte con cuatro planos, estos son arbitrarios, por lo que el proceso es más costoso que en el caso de tener que comprobar únicamente planos cartesianos que permiten desacoplar completamente las ecuaciones en las tres direcciones x, y, z . La ecuación para t que se debe resolver en general es:

$$Ax(t) + By(t) + Cz(t) + D = 0, \tag{4.67}$$

donde A, B, C y D son los coeficientes que definen el plano de cada cara del elemento. En mallas cartesianas las seis ecuaciones resultantes son $x(t) = x_0, x(t) = x_1, y(t) = y_0, y(t) = y_1, z(t) = z_0$ y $z(t) = z_1$. Sin embargo, en mallas tetraédricas debemos resolver la ecuación (4.67) para cuatro planos arbitrarios. Aunque esto no supone una diferencia apreciable en el número de operaciones, sí que resulta en un error mayor en la determinación de la posición final de la partícula, que puede no quedar exactamente sobre la frontera que atraviesa. Esto hace necesario trabajar con tolerancias mayores para evitar problemas con la elección del elemento al que va la partícula tras cruzar el plano y en el movimiento ese mismo elemento. Por supuesto, esta tolerancia depende de la aproximación que usemos para la trayectoria de la partícula, y será mayor si usamos (4.64) que si usamos la aproximación (4.65), ya que la incertidumbre en la solución de la ecuación será mayor.

4.6. Simulación de un DG MOSFET de 10 nm de puerta

Para mostrar el funcionamiento del simulador utilizamos un transistor MOSFET de doble puerta. El transistor tiene longitud de puerta de 10 nm con un espesor del cuerpo de silicio de 6 nm y un espesor equivalente del óxido de puerta de 0,5 nm. La fuente y el drenador tienen un dopado constante de $1 \times 10^{20} \text{ cm}^{-3}$ y el canal sin dopar.

En primer lugar esta estructura sirvió para estudiar el impacto de las autofuerzas al usar una malla tetraédrica, principalmente por su alta simetría. Como se comentó en la sección 4.5.2, cuando simulamos un dispositivo, no todas las fuerzas que aparecen sobre una partícula sola en el dispositivo son autofuerzas, sino que algunas tienen su origen en las condiciones de contorno o en la inhomogeneidad del dispositivo debida, por ejemplo, a la presencia del dieléctrico de puerta. Sin embargo, en el caso del transistor de doble puerta, debido a su simetría, en el centro del dispositivo la fuerza que sienta una partícula debería tener su origen únicamente en la malla. Por tanto, para comprobar la magnitud de esta fuerza hacemos simulaciones de la estructura del dispositivo en las que la única carga es una carga puntual en diferentes posiciones del dispositivo. Las figuras 4.12 y 4.13 muestran la magnitud del campo eléctrico en diferentes regiones del dispositivo. Queda clara una correlación entre la magnitud de una componente del campo eléctrico y la distancia a las fronteras perpendiculares a esta. Esto se debe a que para cada componente la ruptura de la simetría esférica es más pronunciada en esa dirección. Además también mostramos el módulo del campo eléctrico, que podemos ver que es mínimo es en centro del dispositivo.

Además de probar con parejas asignación-interpolación que producen teóricamente autofuerza cero, también probamos con otras dos más sencillas que no anulan las autofuerzas. Una de ellas utiliza el valor del campo en cada elemento constante y la otra asigna la carga a los nodos y la introduce en la ecuación (3.116) como una densidad. Para ello la carga en cada nodo se divide por el volumen asociado a este nodo, que es la suma de la cuarta parte de los volúmenes de todos los tetraedros que contienen al nodo. Una vez resuelta la ecuación de Poisson se interpola el campo eléctrico a la posición de la partícula. Este último método de asignación de carga es el equivalente al CIC utilizado habitualmente en simulaciones usando el método de diferencias finitas. La figura 4.14 muestra la reducción en la fuerza ejercida sobre la partícula en el MOSFET de doble puerta simulado cuando comparamos nuestro esquema

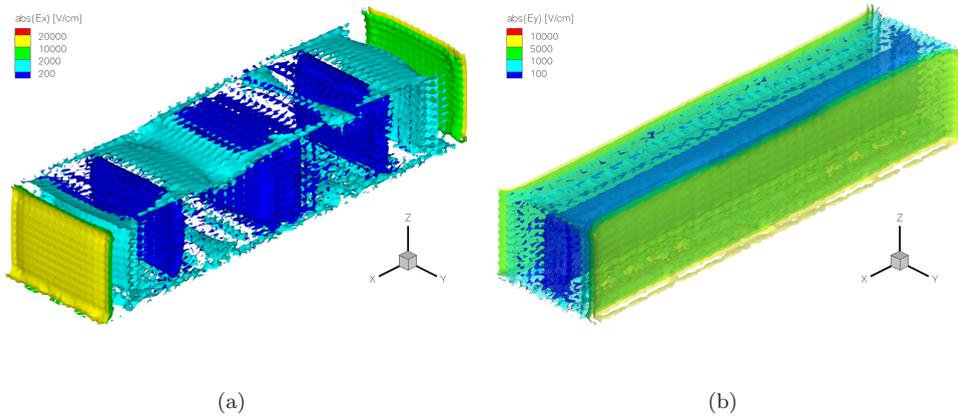


Figura 4.12: Isosuperficies (a) de la componente x y (b) de la componente y del campo eléctrico actuando sobre una partícula situada en el interior del dispositivo.

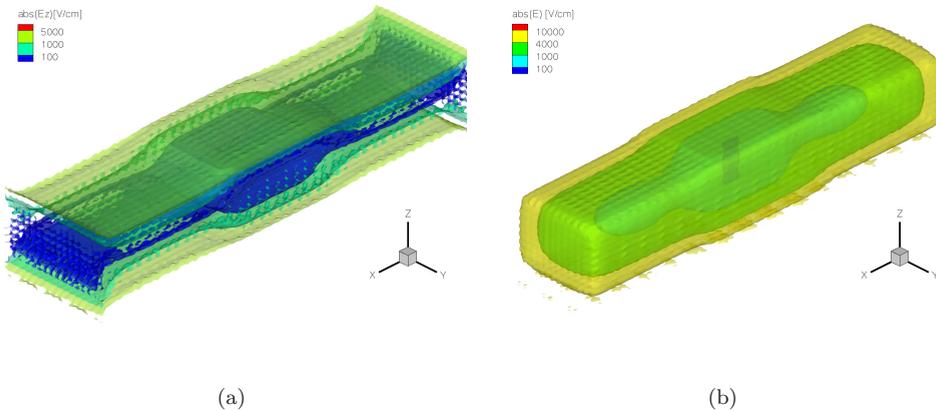


Figura 4.13: Isosuperficies (a) de la componente z y (b) del módulo del campo eléctrico actuando sobre una partícula situada en el interior del dispositivo.

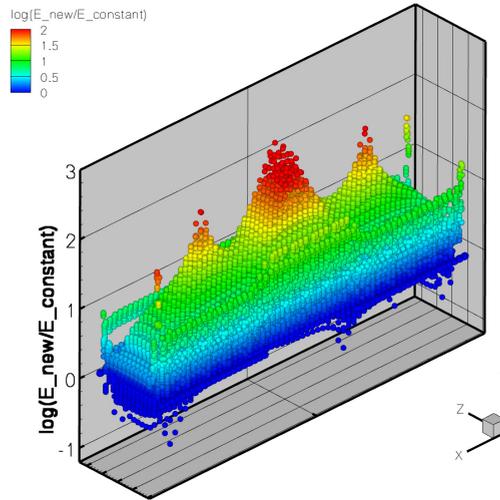


Figura 4.14: Reducción del campo eléctrico, en órdenes de magnitud, sobre la posición de una partícula en el transistor MOSFET de doble puerta de 10 nm de longitud de canal.

con el de campo constante sobre cada elemento [Lau96]. Podemos ver que la mejora más importante está lejos de las fronteras, concretamente en el centro del dispositivo, donde la contribución de las condiciones de contorno se cancela parcialmente. Sin embargo, cerca de las fronteras, donde la principal contribución proviene de la ruptura de simetría, la reducción es menor. Esto sugiere que la mejora está asociada a un mejor esquema de asignación-interpolación que supone una reducción en las autofuerzas, lo que confirma que una elección apropiada de estos dos procesos puede reducir en gran medida las autofuerzas al descomponer el dominio en tetraedros.

Los diferentes esquemas llevarán a diferentes fuerzas y por tanto, tendrán un impacto sobre la corriente en el dispositivo. Para cuantificar este impacto simulamos el dispositivo usando los diferentes esquemas asignación-interpolación sobre las curvas I_D - V_G del dispositivo (figura 4.15). Como las autofuerzas son más importantes cuando los campos reales son bajos, realizamos las simulaciones con un voltaje de drenador de 50 mV. Además, también comparamos los resultados con una simulación arrastre-difusión usando Medici [med] y con una simulación Monte Carlo no autoconsistente. Las curvas de las simulaciones autoconsistentes están etiquetadas como SC MC-1, SC MC-2 y SC MC-3 para los diferentes esquemas asignación-interpolación: SC MC-1

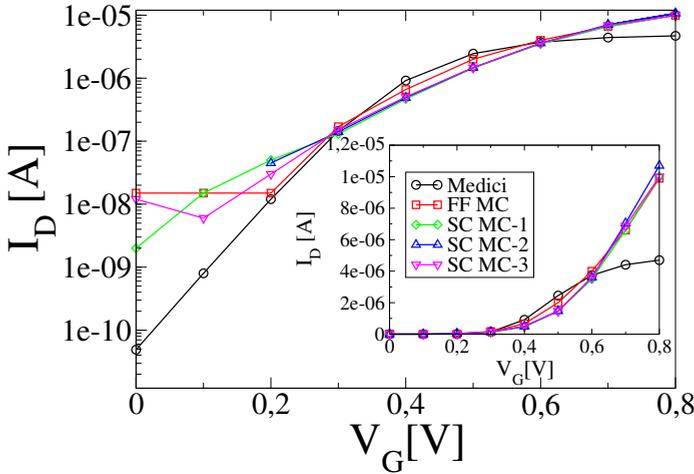


Figura 4.15: Curvas I_D - V_G a $V_D = 0,05$ V del DG MOSFET de 10 nm de longitud de puerta obtenidas con diferentes métodos de simulación: arrastre-difusión (Medici [med]), Monte Carlo no autoconsistente (FF MC) y Monte Carlo autoconsistente (SC MC-1,2,3).

es el previamente descrito como equivalente al CIC estándar, SC MC-2 el esquema que usa un campo constante en cada elemento y SC MC-3 el esquema que produce autofuerzas nulas teóricamente. Vemos en la figura que el impacto sobre la corriente con un voltaje de puerta alto es muy pequeño debido a que los campos del dispositivo dominan sobre las autofuerzas, aunque parece que SC MC-2 sobreestima un poco la corriente. Sin embargo, sí que notamos una cierta diferencia en la región subumbral, donde parece que el esquema SC MC-3 se comporta mejor.

Conclusiones

La simulación de dispositivos electrónicos se ha convertido en una herramienta imprescindible en el diseño de circuitos integrados, puesto que su uso permite acortar y abaratar las fases de diseño y evitar costosos y numerosos experimentos al evaluar el rendimiento *a priori*, descartando opciones y realizando optimizaciones de parámetros.

El escalado de dispositivos hacia dimensiones cada vez más reducidas hace necesaria la investigación de nuevas alternativas a las tecnologías actuales. Además, también han de ser tenidos en cuenta efectos que cobran cada vez más importancia con la reducción de las dimensiones físicas características de los dispositivos. Todo esto hace necesario el uso de simulaciones tridimensionales que permitan tanto predecir el rendimiento de nuevas propuestas como de evaluar el impacto de no idealidades sobre los dispositivos. Sin embargo, estos dos objetivos tienen a veces requisitos diferentes en cuanto a los modelos de simulación utilizados. Por ejemplo, en el caso de querer estudiar el impacto de la aleatoriedad en la posición de los dopantes sobre la tensión umbral de los dispositivos, puede ser suficiente el uso del modelo arrastre-difusión, pero este modelo no será suficiente cuando queramos evaluar el posible rendimiento al escalar un transistor TriGate MOSFET a longitudes de puerta cercanas a los 15 nm.

Pese a esta diferencia, en ambos casos el coste computacional es muy elevado, por lo que la paralelización de los códigos de simulación es muy importante para los dos modelos. Además, para la optimización de las simulaciones también es de especial relevancia la malla sobre la que se discretizan las ecuaciones, ya que un exceso de nodos puede llevar a simulaciones inviables desde un punto de vista computacional.

Para abordar estas problemáticas hemos dividido el trabajo en tres líneas directamente relacionadas entre sí:

Generación de mallas

Hemos desarrollado dos algoritmos diferentes de generación de mallas para la discretización de los dominios que surgen en la simulación de dispositivos semiconductores. El primero de ellos está basado en la utilización de un árbol octal y está optimizado para la discretización de estructuras de tipo Manhattan, compuestas por paralelepípedos rectangulares. La flexibilidad que permite al definir regiones de refinado es especialmente adecuada para simular efectos espacialmente localizados que produzcan variaciones bruscas de magnitudes como el potencial electrostático o la densidad de portadores. El segundo algoritmo está basado en situar un nodo de la malla en la posición de cada átomo de la red cristalina. Mediante simulaciones es posible obtener la estructura relajada de interfaces entre materiales. Utilizando estos datos para dos sistemas, Si/SiO₂ y Si/SiO₂/HfO₂, generamos patrones que pueden ser utilizados en la generación de mallas de dispositivos en las que podemos, por ejemplo, situar cada dopante en su sitio exacto de la red, sin tener limitaciones en la resolución o estructura de la malla. Finalmente, desarrollamos un algoritmo de refinado adaptativo de mallas basado en el algoritmo de bisección. Este algoritmo puede utilizar diferentes variables o funciones de variables para guiar el refinado. Su utilización posibilita mejorar la convergencia de las ecuaciones sin tener que recrear la malla y, en general, ahorrando nodos.

Estos algoritmos fueron posteriormente utilizados en simulaciones, tanto basadas en la aproximación de arrastre-difusión para estudiar fluctuaciones, como en la aproximación Monte Carlo para evaluar nuestra implementación.

Aproximación de arrastre-difusión

El primer paso en cuanto a la simulación utilizando la aproximación de arrastre-difusión fue la implementación de nuevas fuentes de no idealidades en los dispositivos. En concreto, empezamos el estudio evaluando el impacto de la granularidad tanto de la puerta de polisilicio como del óxido de puerta. En el primer caso incluimos una carga interfacial para imitar los estados que aparecen en la unión de los granos del polisilicio. Comprobamos la influencia que tendría una única frontera en diferentes zonas de la puerta para dos dispositivos diferentes con longitudes de puerta de 80 y 25 nm. En este caso encontramos un impacto mucho mayor en el transistor de 25 nm de longitud de puerta, especialmente con polarización de drenador alta. En el caso del óxido de puerta estudiamos un caso particular que puede surgir, la cristalización del óxido en el proceso de fabricación. En este modelo sencillo consideramos

únicamente dos posibles orientaciones de los cristales y, por tanto, dos constantes dieléctricas. Encontramos una fuerte influencia tanto de la orientación de la frontera como de la polarización de drenador. En este caso sólo analizamos el MOSFET de 25 nanómetros de puerta, puesto que este tipo de estudio requería una puerta de dimensiones poco mayores que el tamaño típico de los cristales.

Para comprobar la convergencia de las ecuaciones discretizadas sobre las mallas atomísticas realizamos un estudio del impacto de la posición aleatoria de los dopantes en la fuente y el drenador de un transistor MOSFET de doble puerta de 10 nm de longitud de canal. Validamos los resultados comparando las corrientes con las obtenidas sobre una malla estructurada muy fina. Comparamos el caso continuo, en el que los valores de las corrientes son prácticamente idénticos y dos configuraciones con dopantes aleatorios, las correspondientes a la distribución que produce la máxima corriente y a la que produce la mínima corriente. Al comparar los resultados con la simulación usando asignación de carga CIC vemos que la corriente subumbral es muy parecida, aunque ligeramente más baja en la simulación con la malla atomística. Sin embargo, para corrientes de puerta elevadas esta diferencia aumenta, debido a la diferencia en los pozos de potencial asociados a los dopantes. Al compararlo con la asignación de carga NGP vemos que la corriente es prácticamente la misma para una de las configuraciones pero que para la otra configuración el sistema no converge sobre la malla estructurada. Esto da una idea de la mejora en la convergencia al utilizar la malla atomística.

También sobre la misma malla estudiamos la influencia que tiene considerar la región de transición entre materiales en la interfaz óxido-semiconductor. Utilizando los datos de simulaciones DFT de la interfaz asignamos a cada átomo sus características propias. Encontramos que el único efecto se debe a la disminución de la permitividad en el óxido en las proximidades del silicio. Esto resulta en una pendiente subumbral un poco mayor y un ligero incremento de la corriente a voltajes de puerta altos.

Finalmente estudiamos el impacto de la posición aleatoria de los dopantes en la capa de dopado δ de un MOSFET III-V con arquitectura *Implant Free*. Para realizar este estudio adaptamos la generación de mallas atomísticas a esta estructura debido a las buenas propiedades de convergencia que habíamos obtenido en el estudio previo utilizando un MOSFET de doble puerta. Tras la calibración del simulador frente a curvas obtenidas usando el método Monte Carlo, centramos este estudio en dos aspectos: la variabilidad introducida por los dopantes y la influencia de la posición de estos en el dispositivo. Para

el primer estudio simulamos más de 200 dispositivos y encontramos que la dispersión en los valores de la tensión umbral es muy semejante a la de los MOSFETs de arquitectura convencional. Por otro lado, para estudiar la influencia de la posición simulamos otro conjunto de dispositivos, pero este con un número de dopantes constante en la región de dopado δ . Estudiando la correlación entre la tensión umbral y la posición de los dopantes encontramos que debajo de la puerta los dopantes producen un aumento de la corriente, mientras que los dopantes cercanos a la puerta pero fuera de ella producen el efecto contrario. En el caso de los dopantes alejados de la puerta no existe una correlación con el voltaje umbral.

El método de simulación Monte Carlo

Hemos integrado un simulador Monte Carlo para silicio con el simulador 3D de elementos finitos. El nuevo simulador utiliza los resultados del método arrastre-difusión como condiciones iniciales para la inicialización de partículas en la simulación Monte Carlo. Hemos realizado esta integración de tal forma que se mantiene la implementación paralela del código. El esquema de paralelización escogido es diferente al utilizado para la resolución de los sistemas lineales. En lugar de repartir las partículas según los subdominios de la malla, realizamos un reparto aleatorio de las partículas entre los procesadores. Podemos ver en las medidas de tiempos que este esquema es muy eficiente para el caso estudiado en el rango de número de procesadores que empleamos (hasta 32). Para reflejar parcialmente los efectos cuánticos en los dispositivos simulados implementamos las correcciones cuánticas mediante el método *density gradient*. El potencial cuántico obtenido de la solución inicial se suma al potencial electrostático en cada iteración para calcular la fuerza efectiva sobre los portadores.

Para la implementación del vuelo de las partículas sobre mallas tetraédricas tuvimos que prestar especial atención a algunos aspectos: inyección de carga en los contactos, movimiento, transmisión entre elementos y colisiones con fronteras en el espacio real en geometrías arbitrarias y asignación de carga e interpolación del campo eléctrico. Este último aspecto es probablemente el más importante, ya que una elección equivocada de los algoritmos de asignación e interpolación pueden llevar a la aparición de autofuerzas, que pueden falsear el resultado final obtenido. Formulamos el problema de una manera consistente con el método de los elementos finitos, de tal forma que obtenemos valores teóricos de las autofuerzas nulos. Sin embargo, también en-

contramos que estábamos limitados por la calidad de los valores del campo eléctrico calculados. Sobre mallas arbitrarias no encontramos un esquema de cálculo que produjese un campo con la propiedad de antisimetría con buena precisión. En cambio, usando mallas tetraédricas estructuradas encontramos que esta propiedad se verifica con una precisión de hasta el 0,1 %. A continuación, para comprobar los algoritmos implementados simulamos una estructura de doble puerta por su alta simetría. Encontramos que el uso de esquemas asignación-interpolación adecuados reduce en gran medida las autofuerzas. Comparando las curvas I-V obtenidas usando diferentes esquemas de asignación-interpolación vimos que el cambio más significativo se producía en la región subumbral, mientras que al aplicar un voltaje de puerta más elevado esta diferencia disminuía, encontrando corrientes muy similares para todos los esquemas probados.

Trabajo futuro

La continuación de este trabajo en un futuro puede orientarse en diferentes direcciones en cualquiera de las tres líneas descritas. En el marco de la generación de mallas sería interesante implementar un generador que permitiera una mayor flexibilidad a la hora de definir los dominios para incluir fronteras curvas para las simulaciones de nuevas arquitecturas. Además, también sería muy interesante generalizar la generación de mallas atomística a regiones de materiales de forma arbitraria.

En cuanto a las simulaciones hay muchos aspectos interesantes que han quedado sin estudiar para MOSFETs de silicio. Por ejemplo, sería interesante estudiar el impacto de los óxidos de alta permitividad desde diferentes puntos de vista. Con el modelo arrastre-difusión podríamos hacer estudios estadísticos del impacto electrostático de las inhomogeneidades en el dieléctrico y con el método Monte Carlo el impacto sobre el transporte, para lo que habría que incorporar nuevos mecanismos de dispersión. Además, con los simuladores podríamos evaluar el comportamiento ante el escalado de diferentes arquitecturas y su robustez ante la influencia de fluctuaciones en los parámetros intrínsecos. Una vez que el simulador Monte Carlo sea debidamente optimizado podríamos abordar también el estudio de fluctuaciones con él. Otra continuación interesante del trabajo incluye la incorporación de nuevos materiales en el simulador Monte Carlo, lo que sería facilitado por el modelo analítico de tres bandas que usa el código. Esto sería completado con la extensión del simu-

lador para heteroestructuras para poder así complementar a las simulaciones usando la aproximación de arrastre-difusión que usamos actualmente.

Conclusions

The simulation of electron devices has become the main tool in the design of integrated circuits, since it allows to shorten and reduce the price of design stages and to avoid expensive and time consuming experiments.

The scaling of devices to nanometric dimensions makes necessary the research on new alternatives to present technologies. Furthermore, new effects begin to play an important role and must be taken into account. To treat this problems, 3D simulations are compulsory, to evaluate the performance of new architectures or the impact of non ideal properties on the behaviour of the devices. However, these studies may have very different requirements in terms of simulation models. For example, to study the impact of random doping on the threshold voltage of a device, the drift-diffusion model could be enough, but it would not be valid to asses the performance of a TriGate MOSFET scaled down to 15 nm gate length.

In both cases, computational cost is very high, so the parallelisation of the simulation codes is very interesting. Furthermore, the simulation mesh is also very important in the optimisation of the problem, since too many nodes may lead to unfeasible simulation from a computational point of view.

To deal with these problems we have divided the work in three lines strongly correlated among them:

Mesh generation

We have developed two different mesh generation algorithms for the discretisation of the simulation domains. The first one is based on the use of an octree and is optimised for the discretisation of Manhattan type structures. The flexibility in the specification of the refinement is especially appropriate to simulate spatially localised effects producing sharp variations in magnitudes such as electrostatic potential or carrier density.

The second algorithm is based on placing a node of the mesh in every position of an atom of the crystal lattice. We obtained the relaxed structure of two systems, Si/SiO₂ and Si/SiO₂/HfO₂, using first principles simulations and, from these data, we generated patterns to be used in the generation of a device mesh. These meshes can be used, for example, to place the dopants in their exact positions without being limited by the mesh size or structure.

Finally, we have developed an adaptive refinement algorithm based in the bisection method. This algorithm can use different variables to guide the refinement. Its use improves the convergence without the necessity to rebuild the mesh and saving nodes.

These algorithms were used in simulations based on the drift-diffusion model to study fluctuations and in the Monte Carlo method to evaluate our implementation.

The drift-diffusion approach

The first step in the simulations using the drift-diffusion approach was the implementation of new sources of variability. We began the study evaluating the impact of granularity of the polysilicon gate and the gate oxide. For the first one, we included an interface charge to mimic the states appearing in the boundaries between grains. We studied the impact of a sole boundary in different regions of the gate for two different devices with gate lengths of 80 and 25 nm. We found a much higher impact in the 25 nm gate length device, especially for a high drain bias. In the case of the gate oxide we studied one particular situation that may arise, the crystallisation of the dielectric in the fabrication process for the 25 nm gate length MOSFET. In this simple model we consider only two different dielectric constants corresponding to two different orientations. We found a very strong influence in the polarisation of the drain contact and in the orientation of the boundary.

To study the convergence properties of the atomistic meshes and their suitability for our simulation problems, we studied the impact of the random doping in the source and drain regions of a 10 nm gate length MOSFET. First we validated the results comparing the drain current with that obtained from simulations using a structured tetrahedral mesh. We found that their values were almost identical for the device with continuous doping. We also compared the results for two different dopant configurations: the one producing the highest current and the one producing the lowest current. When we compared the results with those obtained using the CIC charge assignment, we found

that the subthreshold current was very similar, although higher for the CIC charge assignment, but in the on-current this difference increases. This effect is caused by the different shape of the potential wells associated to the dopants. Comparing the result with NGP charge assignment simulations we found that the current was almost the same for one of the configurations, but for the other one the system did not converge on the structured mesh. This is an indication of the improvement in the properties of the linear systems when we use the atomistic mesh.

Also on this atomistic mesh we studied the influence of the transition region between silicon and silicon dioxide on the characteristics of the device. We found that the only effect comes from the decrease in the permittivity in the oxide in the region near the silicon. This results in a slightly better subthreshold characteristics and a small increase in the on-current.

Finally, we studied the impact of random doping in the δ doping layer of an Implant Free MOSFET. To accomplish this study we adapted the atomistic mesh generation for this structure. After the calibration of the device against Monte Carlo simulations, we focused the study in two aspects: dopant induced variability and influence of the position of the dopants. For the first one, we simulated more than 200 devices and found that the dispersion in threshold voltage is very similar to that obtained for conventional SG MOSFETs of similar gate lengths. For the second study we simulated another ensemble but with a fixed number of dopants in the δ doping layer. Studying the correlation between the threshold voltage and the dopant position we found that the dopants below the gate produce an increase in the current, whereas the dopants near the gate produce the opposite effect. For the dopants far from the gate there is no correlation between dopant position and threshold voltage.

The Monte Carlo simulation method

We have integrated a Monte Carlo simulator for bulk silicon in our 3D finite element simulator. The results of the drift-diffusion model are used for the initialisation of the particles. The integration has been carried out maintaining the parallel implementation of the code, but the scheme for the Monte Carlo algorithm is different from the one used to solve the linear equations. Instead of sharing the particles according to their real physical position, we assign them randomly to the processors. We found that the scheme is very efficient for the studied device and for the number of processor employed (up to 32). To reflect quantum effects in the simulated devices we have implemen-

ted quantum correction using the density gradient approach. The quantum potential obtained in the drift-diffusion solution is then kept and added to the self-consistent electrostatic potential to calculate the effective force on the particle.

To implement the flight of the particles on tetrahedral meshes we had to pay especial attention to some aspects: charge injection in the contacts, movement, transmission between elements and collision with boundaries in real space for arbitrary geometries, charge assignment and electric field interpolation. These last two points are probably the most important ones, since a wrong combination of them may lead to self-forces. We have formulated the problem consistently with the finite element method in such a way that we obtain, theoretically, zero self-force. However, we found that are limited by the quality of the calculated electric field. On arbitrary meshes we did not found any scheme producing an electric field with the required symmetry properties. Nevertheless, using structured tetrahedral meshes we found that the field has the required symmetry within a margin of 0,1 %. Finally, to validate the implemented algorithms, we simulated a DG MOSFET structure due to its high symmetry. We found that the appropriate combination of charge assignment and field interpolation greatly reduces the self-forces. Comparing the I-V characteristics for different charge assignment-field interpolation pairs we found that the most important difference is in the subthreshold region, whereas in the on-current region the difference was quite small for all the schemes.

Future work

The continuation of this work can be oriented in different directions in any of the research lines treated in the dissertation. In the field of mesh generation, it would be very interesting to implement an algorithm with greater flexibility in the definition of the domains but which would allow to maintain the ease in the refinement. It would also be very interesting to extend the atomistic mesh generation framework to arbitrary interfaces.

Regarding the simulations there are many interesting aspects which could be studied. For example, it would be interesting to study the impact of high- κ dielectrics from a different point of view. We could study the impact of inhomogeneities, fixed or interface charges on the threshold voltage with the drift-diffusion model and the impact on the transport with the Monte Carlo method, adding the relevant scattering mechanisms for this case. We could

also study the behaviour of new architectures when they are scaled and their robustness against intrinsic parameter fluctuations. Once the Monte Carlo simulator is properly optimised, we could also use it in studies of fluctuations. Another interesting continuation would be the inclusion of new materials in the Monte Carlo simulator, which would be facilitated by the band model implemented. This would be completed with the extension to heterostructures in order to complement the simulations using the drift-diffusion model we are currently using.

Apéndice A

Development of tools for the numerical simulation of MOSFETs

A.1. Modified octree mesh generation for Manhattan type structures

One of the most important steps in the process of semiconductor device simulation, or any other numerical simulation based on finite elements, finite differences or similar standard techniques, is the discretisation of the domain of the problem. A mesh must be generated, and its properties determine the stability of the numerical solver, computational time and quality of the solution.

We have modified the classical model for *octree* generation to optimise the program for special regions of interest in the semiconductor device problem, Manhattan structures with very narrow layers. Using this technique, several meshing patterns have been tested and compared.

Octree based mesh generation has some advantages in the problems we are trying to solve. Its main drawback, the adaptation to complex boundaries, does not appear in our interest area, so we found a fast and robust method, very well adapted to our geometry requirements.

In this section we describe the generation of the *octree* adjusted to the boundary and different tetrahedra generation methods.

A.1.1. Standard octree generation

The first step in the generation is the creation, based on the information given in the input data, of a cuboid enclosing the complete region, O_r (root cell of the *octree*). This cuboid is recursively subdivided into eight octants while they are not outside the region and some refinement criteria are not achieved. There are different ways to indicate these criteria, possibly being the most used the distribution of points over the region with a density depending on the zone, higher where we require more refinement and lower where high resolution is not necessary. Besides, it is common to introduce an extra requirement to the *octree*, the one level difference rule. This is intended to simplify the tetrahedralization process subsequent to the generation of the *octree* and to avoid badly shaped tetrahedra. To fulfill this condition a subsequent phase to balance the *octree* is introduced. To take into account the boundaries, the terminal octants intersecting them are treated in a special and complex way [Yan94, MV00], which is what we want to avoid.

Once the *octree* is finished there are several methods to generate the tetrahedral mesh. The simplest and fastest ones are based on templates, although Delaunay tetrahedralizations are also possible using a posterior step to adjust the boundary or a constrained Delaunay triangulation.

A.1.2. Modified octree generation

In first place we have to read the description of the region to mesh and the desired refinement. The boundaries of the region are described using its vertices, edges and planes. The refinement is specified as a global level of deepness and, if necessary, a level in the tree in some given zones which can be planes, cuboids, straight lines or sets of points for any other region.

Since the regions we are interested in are composed by rectangular parallelepipeds, we have made a modification in the classical scheme to obtain an algorithm optimized to the situation.

In first place we introduce some notation. We call O to the set of all the octants in the *octree*, both the internals, O^* , and the externals, O_e , to the region Ω and o_j to the octants:

$$O = O^* \cup O_e$$

where

$$O^* = \{o_j \in O / o_j \cap \Omega \neq \emptyset\}$$

and

$$O_e = O - O^*$$

In addition to this we divide the set O^* in two subsets, O_d and O_t ,

$$O^* = O_d \cup O_t$$

where

$$O_d = \{o_j \in O^* / \mathcal{D}(o_j) = \emptyset\}$$

being $\mathcal{D}(o_j)$ the offspring of the octant o_j . Finally, the set of all the planes of the structure is called \mathbf{P} .

We divide the *octree* generation in three steps. In the first one we generate an intermediate uniformly refined *octree* O_i with depth given as the input parameter n_D , determined by the size of the different layers. In this point we make the adaptation to the planes of the structure, both the internal and the external ones. In the last step we refine the tree in the regions given by the input parameters as points, lines, planes and volumes:

1. $O_r \longrightarrow O_i^* / \mathcal{N}(o_j) = n_D \quad \forall o_j \in O_{i,d}$, where $\mathcal{N}(o_j)$ is the level in the tree of the octant o_j
2. Deform $o_j \in O_{i,d} / o_j \cap \mathbf{P} \neq \emptyset$ and transmit these deformation through their descent to avoid problems in step 3
3. Build the final *octree* O^* from O_i^* guided by the control refinement regions

Step 1 is carried out in the usual way for the building of an octree. Step 2 begins with the detection of terminal octants intersecting any of the boundaries. Each of these octants are deformed to fit the planes of the region and an operation to go through its descent and modify them whenever it is necessary is performed. After step 2 we have an octree with nodes at the same level representing cuboids with different shapes and volumes.

Now we have the final *octree* we can construct the embedded tetrahedral mesh, process described in the next section.

A.1.3. Tetrahedral mesh generation algorithms

With the *octree* finished we can use several templates to generate the tetrahedra. The four main differentiating characteristics are the number of elements generated, its quality, the distribution of connectivities and whether

it is necessary to check the state of the neighboring octants. We have used this last factor to select the algorithms to implement. The main drawback of this choice is the subsequent incapability to use the octant division scheme which produces the lowest number of tetrahedra, five per terminal octant in the best situation. The algorithms we have implemented produce six, twelve, sixteen, twenty and twenty four tetrahedra per octant if there is no neighbor octant with a higher level of refinement. These algorithms present different levels of anisotropy, which makes some of them more appropriate for some device structures.

The algorithms consist of two parts. In the first one the nodes of the mesh are generated, whereas in the second one the connectivities among them are defined, finishing then the tetrahedra generation. In the first part it is clearly necessary the knowledge of the level in the tree of the neighboring octants, but the second step is completely independent for each octant, fact derived from the decision made about the sort of tetrahedralization algorithms implemented.

Now we briefly describe the five algorithms implemented and its basic characteristics, analysed more deeply below with some examples. Fig. A.1 show the connectivities of a terminal octant for each algorithm in the case there are no neighbors with higher deepness in the tree.

Algorithm A

This algorithm generates the lowest number of nodes, eight in the best case, and tetrahedra, six per octant when there is no neighbors with higher level of refinement. If there is any of these more refined neighbors we introduce an internal node in the octant to avoid a very high number of cases to consider, which would greatly complicate the coding, being furthermore very error prone.

Algorithm B

This algorithm generates superficial triangulations equivalents to the previous one, but a higher number of nodes due to the introduction of a node in the center of every octant, that is, twelve tetrahedra and nine nodes per octant in the best case.

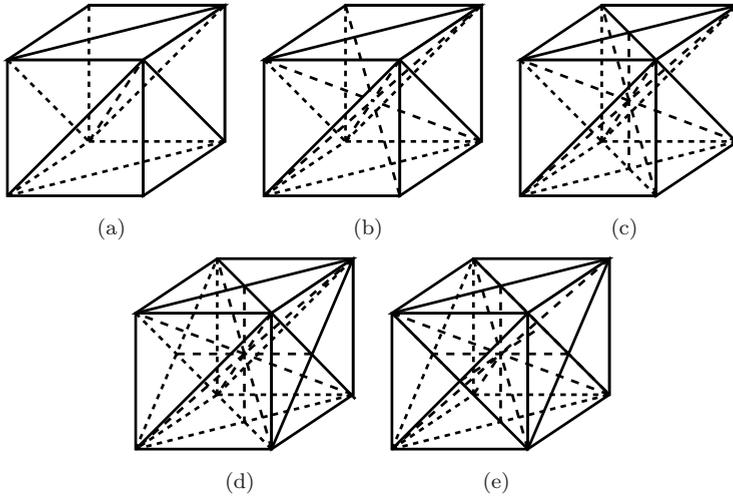


Figura A.1: Conectividades among nodes in a terminal octant for (a) algorithm A, (b) algorithm B, (c) algorithm C, (d) algorithm D and (e) algorithm E

Algorithm C

In this case we introduce a certain level of anisotropy in the meshing of an octant changing the superficial triangulation in two opposite faces of every octant. The strategy used to generate the tetrahedra is the same as the used in the previous algorithm. The resultant number of nodes and tetrahedra in the best case are eleven and sixteen respectively.

Algorithm D

This is the logical extension of the previous algorithm, changing the triangulation of another two opposite faces, therefore obtaining thirteen nodes and twenty tetrahedra per terminal octant with no neighbor of higher deepness in the tree.

Algorithm E

In this case we have completely changed the initial way of generating the triangulation of the octants to develop this algorithm. We divide every face of the terminal octant in four triangles, requiring the creation of fifteen nodes

and twenty four tetrahedra in the best case.

A.1.4. Mesh quality

As stated above, when a mesh is used in simulations, the shape of the tetrahedra has a great importance, since it partially determines the convergence of the numerical solver. However, the optimal mesh is problem dependent, so a universal quality measure can not be defined. In this point, an element quality measure satisfying the definition proposed in [DLGC98] has been used. Specifically, the *aspect ratio*, γ , has been the selected tetrahedron shape measure:

$$\gamma = \frac{12}{\sqrt{6}} \frac{r_{in}}{\max_{i,j} l_{ij}} \quad (\text{A.1})$$

where r_{in} is the radius of the inscribed sphere and l_{ij} ($l_{01}, l_{02}, l_{03}, l_{12}, l_{13}, l_{23}$) are the tetrahedron edge lengths. The constant has been chosen to obtain a maximum value of one. To calculate the inradius we use the following formula:

$$r_{in} = \frac{3v}{\sum_{i=0}^3 s_i} \quad (\text{A.2})$$

being v the volume of the tetrahedron and s_i the areas of each face of the tetrahedron.

Several parameters characterising the distributions of the quality of the meshes for two different semiconductor devices, a HEMT and a MOSFET, and the five algorithms are shown in Table A.1: average aspect ratio ($\langle \gamma \rangle$), standard deviation (σ), asymmetry coefficient (AC) and maximum and minimum aspect ratios (γ_{max} and γ_{min} respectively). It can be seen the increase in the quality with the inclusion of more tetrahedra per octant, derived from the higher proximity of the tetrahedra generated to the regular tetrahedron used as standard in this measure. We can also see an increase in the standard deviation of the distribution, possibly caused by the higher quantity of different tetrahedra arising in level transitions when we use the algorithms which divide a face in four triangles. We can see that algorithm A produces some very high quality tetrahedra. However, as seen, its small number is not enough to raise the average over the obtained with algorithm E. Another important aspect is the noticeable difference in the number of nodes and elements, which may convert the simulation memory and time requirements in the main decision factor to select the algorithm.

| Algorithm/Device | $\langle \gamma \rangle$ | σ | AC | γ_{max} | γ_{min} |
|------------------|--------------------------|----------|-------|----------------|----------------|
| A/HEMT | 0.565 | 0.0901 | -42.3 | 0.963 | 0.249 |
| A/MOSFET | 0.520 | 0.132 | 214 | 0.940 | 0.137 |
| B/HEMT | 0.543 | 0.0798 | 156 | 0.963 | 0.248 |
| B/MOSFET | 0.515 | 0.114 | -2121 | 0.940 | 0.137 |
| C/HEMT | 0.577 | 0.0888 | -182 | 0.963 | 0.249 |
| C/MOSFET | 0.547 | 0.117 | -2407 | 0.924 | 0.184 |
| D/HEMT | 0.599 | 0.0950 | 299 | 0.963 | 0.281 |
| D/MOSFET | 0.561 | 0.128 | 72699 | 0.924 | 0.195 |
| E/HEMT | 0.609 | 0.103 | -4909 | 0.752 | 0.259 |
| E/MOSFET | 0.566 | 0.141 | 9266 | 0.752 | 0.156 |

Tabla A.1: Parameters describing quality distributions of the tetrahedra for the studied meshes

A.2. Atomistic mesh generation for the simulation of nanoscale MOSFETs

High quality mesh generators are required in order to accurately resolve atomic scale effects in the simulation of complementary metal-oxide-semiconductor (CMOS) devices with nanometre dimensions. At these dimensions, the discreteness of matter begins to play an important role and strongly affects device characteristics [Ase98]. For example, the effects of random dopants will introduce variations in the behaviour of semiconductor devices [Ase98, Fra02, BAW02]. However, in these studies, to our best knowledge, the exact positions of the dopants in the silicon crystal lattice have not been accurately resolved.

We present a tetrahedral mesh generation framework allowing to reproduce the exact position of the atoms in a semiconductor device. We apply this framework to two material systems: (i) a silicon semiconductor with α -quartz SiO_2 dielectric layer and (ii) a silicon semiconductor with the monoclinic HfO_2 on the top of α -quartz SiO_2 layer.

A.2.1. Meshing approach for crystalline regions

We have chosen a method based on an octree algorithm [APGL06, HK05] with unit cells of materials as terminal octants and the possibility to utilise different decomposition schemes in the same mesh. As a first step, tetrahedra

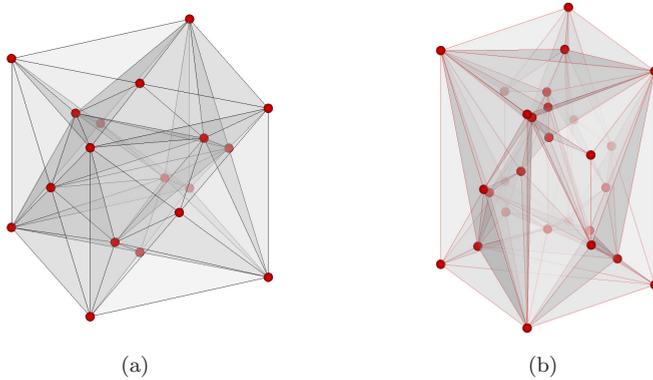


Figura A.2: Mesh of the terminal octant using the atoms in the unit cell of (a) Si and (b) crystalline SiO_2 as nodes.

decomposition schemes for the extended unit cells of the materials in the structure must be defined. These extended unit cells (unit cells from now on) include extra dummy atoms in order to complete a volume which covers the full space when it is replicated, but they will be removed later in the replication process. The decomposition of the unit cells into tetrahedra can be carried out in different ways, the only requirement is the symmetry in the surface of opposite faces so that we obtain a conformal mesh in the process of replication. In Fig. A.2 we show, as an example, the decomposition of the unit cells of silicon (Si) and crystalline silicon dioxide ($\alpha\text{-SiO}_2$).

A second step must be the decomposition of the periodic structure at the interface between different materials, e.g., the Si/ SiO_2 interface. Such a decomposition cannot be done simply using the bulk crystalline structures of corresponding materials because of the deformation of their local atomic structure at the interface. We have adopted the topology of this interface from Ref. [YKU⁺01], while the local atomic structure of the interface has been obtained using the density functional theory (DFT) calculations as described in Ref. [GSPS07]. An example of the decomposition of an interface cell is shown in Fig. A.3 for the Si/ SiO_2 interface. Note that the upper part of the decomposition of the cell is the same as that shown in Fig. A.2(b), but the bottom part, the interface between Si and α -quartz, is different.

Now we have the basic units to build the mesh, we can proceed to the definition of the structure we want to mesh. This includes the specification of materials and approximate dimensions of different layers. The information

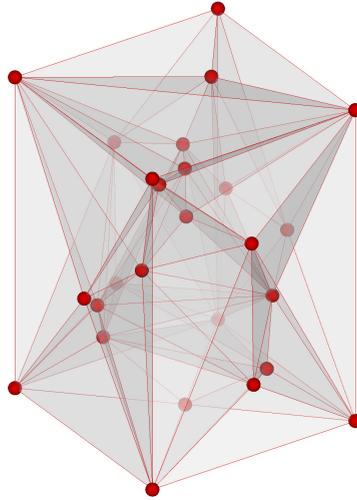


Figura A.3: Pattern used to build the mesh of the interface between crystalline SiO_2 and crystalline Si.

is used to obtain the number of unit cells per material to be replicated in each direction, which is done in the next step. We have to take into account that a material layer could have zero width if the specified dimension is too small. In this case, the whole width of the region would be assigned to the interface. Once the mesh of the materials with the given dimensions is built, the displacement of the nodes is carried out in order to match the actual relative positions of atoms.

The final step in building the mesh is the introduction of the interfaces, which always have only one layer to join the materials above and below. The coordinates of the nodes are displaced directly according to the mesh of the materials they are interfacing. Fig. A.4 shows an example of the mesh constructed using the described technique for a $\text{SiO}_2/\text{Si}/\text{SiO}_2$ structure. The relaxed atomistic structure of the Si/SiO_2 interface has been obtained from first principles calculations [YKU⁺01] using DFT. Fig. A.5 shows another example of our methodology applied to a $\text{Si}/\text{SiO}_2/\text{HfO}_2$ structure.

Again, the $\text{SiO}_2/\text{HfO}_2$ interface was extracted from the atomic distribution obtained using first principles calculations [GS07]. In this case, the growth of the monoclinic HfO_2 at the top of Si/SiO_2 structure did not produce a crystalline interface. However, it still had some patterns which we extracted in order to obtain the cell which represents the interface. With this modification, the

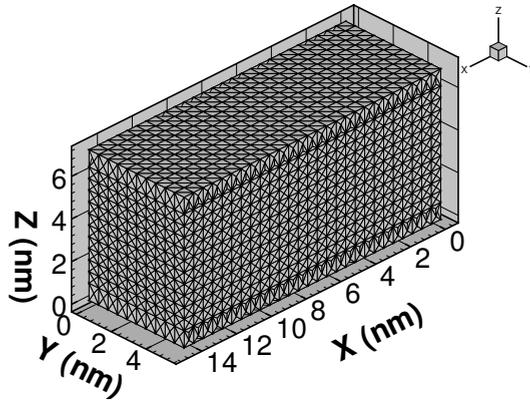


Figura A.4: Mesh of a DG-MOSFET with crystalline SiO_2 as gate dielectric using patterns.

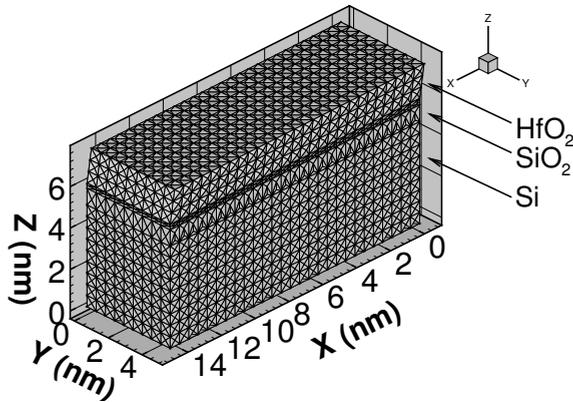


Figura A.5: Mesh of a MOSFET with gate dielectric composed of one layer of crystalline SiO_2 and monoclinic HfO_2 .

obtained mesh does not replicate exactly the initial atom positions. However, in the next section we present a technique to modify these crystalline meshes making them closer to the real structures.

A final difficulty, which must be addressed building the atomic resolution meshes, is the difference between the lattice constants of the materials forming the layered structure. As the first approximation we suppose that the crystalline oxide layer is thin enough to adapt its lattice constant to that of the Si [Yua99]. The crystal of the oxide layers with the dimensions of Si in the plane parallel to the interface is therefore created.

A.2.2. Meshing of amorphous layers

In the previous section, we have shown how our mesh generation strategy creates meshes for crystalline materials. However, in actual devices, the dielectric layer is usually amorphous. This requires an extension of the mesh generator to allow the handling of amorphous structures.

The description of the structure of amorphous materials and their interfaces is a very complex subject not completely understood yet even for an extensively used material as SiO_2 [FCBA06], whose interface with Si has been the key for the success of CMOS technology. The generation of realistic Si/ SiO_2 interface structures requires the use of computational techniques such as density-functional-theory (DFT) and classical molecular dynamics simulations [FCBA06, PHC96, BPP00, BP03].

As a first approximation to the problem, we will accomplish a local transformation of the mesh in order to remove the crystallinity in the location of the nodes of the oxide to mimic an amorphous node distribution. This can be used to study qualitatively the impact of the random variations of the material properties in the amorphous layer on the device behaviour. However, this transformation results in atom distributions whose physical properties such as distribution of bond angles do not match the experimental ones. Further work should include the inclusion of amorphous oxides and interfaces from first principles simulation data.

The main idea of the algorithm is the formation of an amorphous material layer from the crystalline one we have meshed already with the previously described algorithm. We also would like to keep the same adjacency graph for the mesh to be able to use the same data structures in the simulator, speeding up the simulation of microscopically different devices.

The implemented algorithm performs an adjustable randomisation of the

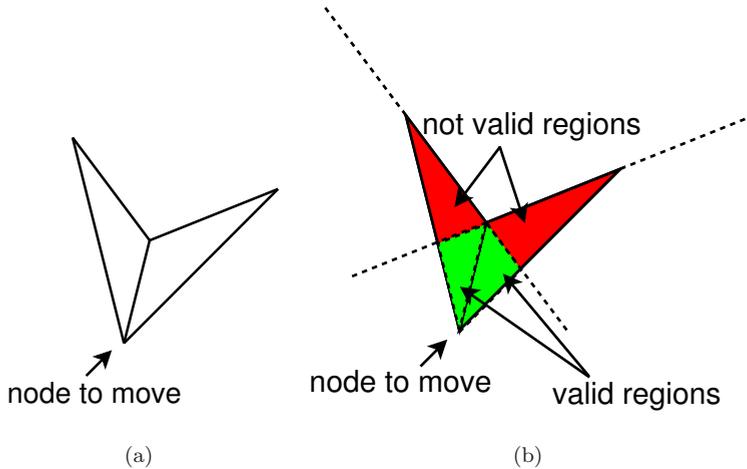


Figure A.6: (a) Two triangles delimiting a non-convex surface. (b) Regions of the surface which can and cannot be used to move the node.

positions of the nodes of the original mesh. The degree of allowed displacement of the nodes is given by a parameter $\beta \in (0, 1]$. A value near zero means that the allowed movement is very small, whereas a value of one specifies a movement in the whole allowed volume. To avoid intersections between tetrahedra when moving nodes we define the allowed volume in two steps. In the first step we make a list of the neighbour tetrahedra. They define a volume surrounding completely the target node. If this volume is convex, then it is the volume where the node can be freely moved without any need to change connectivities. If the volume is not convex, then we have to restrict the volume to avoid intersections. Figs. A.6(a) and A.6(b) show an example of non-convex mesh elements and their adjustment, respectively, in two dimensions.

In three dimensions the situation is equivalent, but we have to restrict the movement using planes instead of straight lines. We also have to take into account that a node may belong to an arbitrary number of tetrahedra, thus defining an arbitrary polyhedron. Hence, the valid volume to move the node will be arbitrary and difficult to define in terms of the tetrahedra containing this given node. Therefore, we have decided to use a rejection technique to generate the final position of the node. Firstly, we generate an arbitrary point in the whole volume defined by the neighbour tetrahedra using a uniform distribution. If the vector defined by the old and new positions of the node do not cross any of the planes defined by all the opposite faces to that node we

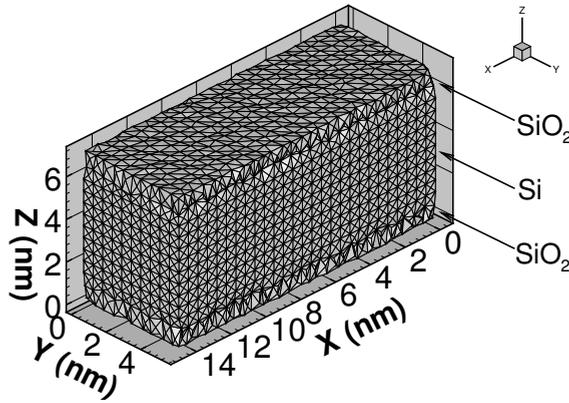


Figura A.7: Mesh with amorphous regions obtained from the mesh of the crystalline materials.

keep the change, otherwise we generate a new random candidate. In this way we obtain a random movement in the valid volume with uniform probability. Running the randomising function several times, we get different amorphous layers from the same crystalline structure. Fig. A.7 shows one mesh generated with this method for a $\text{SiO}_2/\text{Si}/\text{SiO}_2$ structure.

A.2.3. Mesh quality

The characteristics of the systems of equations that describe the discretised partial differential equations which must be solved strongly depend on the properties of the mesh. We have used the aspect ratio, defined in (A.1), as the quality measure.

Table A.2 presents parameters characterising the distributions of γ for the different unit cells, including the average aspect ratio ($\langle \gamma \rangle$), standard deviation (σ), asymmetry coefficient (AC) and maximum and minimum aspect ratios (γ_{max} and γ_{min} respectively).

To avoid highly malformed elements, the movement of the nodes in the amorphous mesh generation includes a second constraint related to the quality. The new position of a node cannot produce any tetrahedron with aspect ratio lower than a given value. Although high values of the threshold would be ideal from the mesh quality point of view, a trade off is required since too high values could make the generation too slow or even impossible.

Tabla A.2: Parameters describing quality distributions of the tetrahedra for the meshes of different materials. The parameters given for the amorphous meshes are $\beta_1 = 0,05$ and $\beta_2 = 1,0$.

| Material | $\langle \gamma \rangle$ | σ | AC | γ_{max} | γ_{min} |
|--|--------------------------|----------|--------|----------------|----------------|
| Si | 0.51 | 0.08 | 1.031 | 0.662 | 0.458 |
| SiO ₂ | 0.31 | 0.09 | 0.399 | 0.553 | 0.151 |
| HfO ₂ | 0.46 | 0.12 | -0.089 | 0.777 | 0.200 |
| Si-SiO ₂ | 0.39 | 0.12 | 0.789 | 0.739 | 0.183 |
| SiO ₂ -HfO ₂ | 0.40 | 0.10 | -0.838 | 0.575 | 0.127 |
| <i>a</i> -SiO ₂ (β_1) | 0.31 | 0.09 | 0.372 | 0.595 | 0.132 |
| <i>a</i> -SiO ₂ (β_2) | 0.29 | 0.12 | 0.789 | 0.845 | 0.102 |

A.3. Study of the impact of grain boundaries in MOSFETs using drift-diffusion simulations

Two important new sources of fluctuations in nanoscaled MOSFETs are the polysilicon gates and the introduction of high- κ gate dielectrics. Using a 3D parallel drift-diffusion device simulator, we study the influence of the polycrystal grains in polysilicon and in the high- κ dielectric on the device threshold for MOSFETs with gate lengths of 80 and 25 nm.

A.3.1. Polysilicon grain boundary

First, we present the results obtained in the simulation of the transistors with a polysilicon gate composed of two grains. In the simulations, we have introduced the interfacial charge producing a potential distribution pinning approximately in the middle of the bandgap which corresponds to Fermi level. Fig. A.8 shows the valence and conduction band profiles along a line crossing the grain boundary.

In Fig. A.9 we present the threshold voltage displacement in the 67 nm effective gate length device associated with the move of a single grain boundary perpendicular to the direction of the current flow from the source end to the drain end of the polysilicon gate. The displacement is calculated with respect to the threshold voltage obtained in the device with no polysilicon grain boundary in its gate.

The grain boundary has the strongest impact when aligned with the maximum of the potential barrier between the source and the drain. At low drain

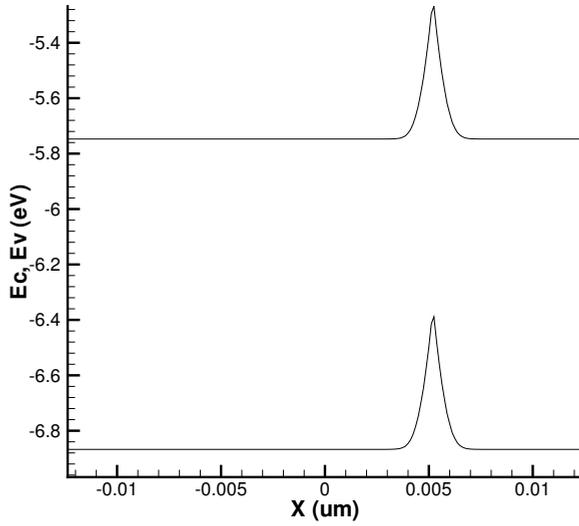


Figura A.8: Conduction and valence bands of the polysilicon in a line crossing perpendicularly to the plane of the boundary.

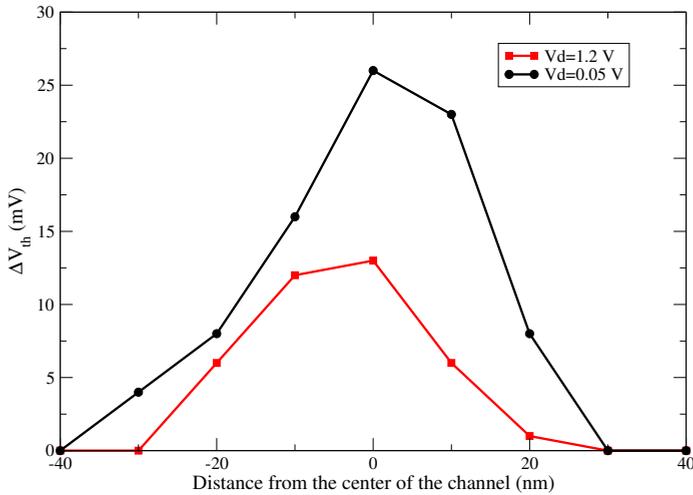


Figura A.9: Threshold voltage displacement in the 80 nm physical gate length device produced by a boundary moving from the source end (negative values) to the drain end (positive values) of the gate.

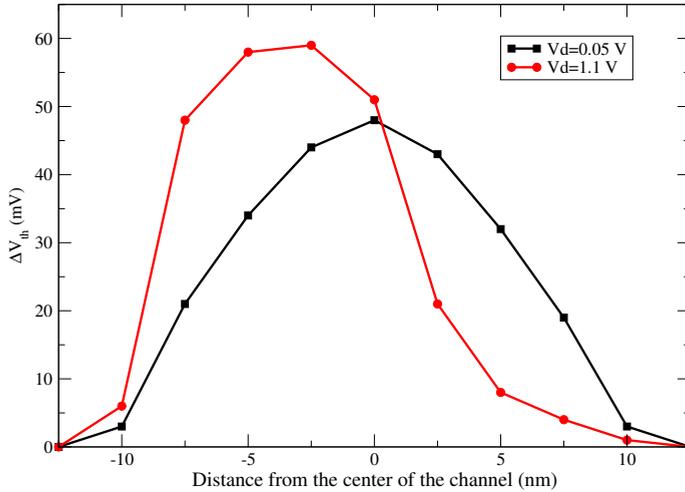


Figura A.10: Threshold voltage shift in the 25 nm gate length device produced by a grain boundary moving from the source end (negative values) to the drain end (positive values) of the gate.

voltage the maximum is close to the middle of the channel but moves towards the source with the increase of the drain voltage.

The simulations in the case of the 25 nm gate length MOSFET illustrated in Fig. A.10 show similar behaviour but different magnitude of the threshold voltage shift. In this case, the values of the threshold voltage displacement are larger by approximately 75 % for the peak displacement at low drain voltage and by 400 % at high drain voltage. As expected, when the boundary is near the source or drain ends the displacement vanishes in both devices.

A.3.2. Gate dielectric granularity

We have also studied the threshold voltage variations in the 25 nm gate length MOSFET caused by the existence of two crystals with a different dielectric constant in the gate dielectric. First, we analyse the influence of the position of the boundary between the crystals and perpendicular to the direction of the current flow. Then, we study the effect of its orientation. We consider that one of the crystals has a dielectric constant of 7 and the other one has dielectric constant of 5. Fig. A.11 shows the threshold voltage displacement as a function of the position of the boundary, perpendicular to the

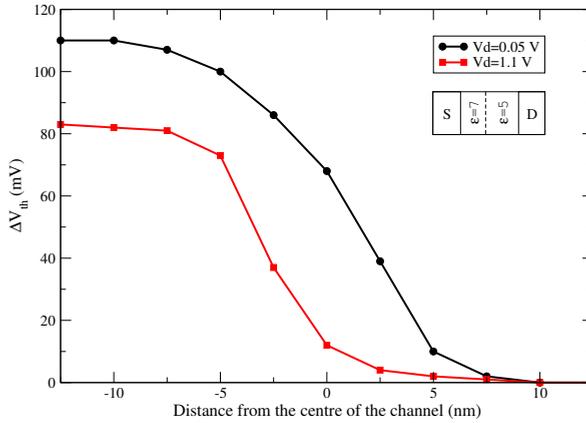


Figure A.11: Threshold voltage displacement in the 25 nm gate length device produced by a boundary in the dielectric moving from the source end (negative values) to the drain end (positive values) of the gate. Displacement values are relative to the device with a Si_3N_4 gate dielectric.

direction of the current flow. At a low drain bias the variation in the threshold voltage is larger. The effective zone, where the boundary introduces strong variation in the threshold voltage, is reduced to approximately 10-15 nm (90% of the variation is in this region). At a high drain bias, the transitional region of influence is further reduced.

Finally, we show the dependence of the threshold voltage displacement on the angle that the grain boundary forms with the direction of the current flow in Fig. A.12. At a low drain bias, we find the highest values of the displacement for boundaries perpendicular to the current flow. These values diminish when the boundary approaches angles of 0 and 180 degrees, although the minimum near the zero angle is reached approximately at 15 degrees. At a high drain bias, the behaviour is noticeably different. In this situation, the most important parameter is the dielectric constant in the source half of the gate, resulting in an oscillation with a period twice as large as the one obtained at a low drain voltage.

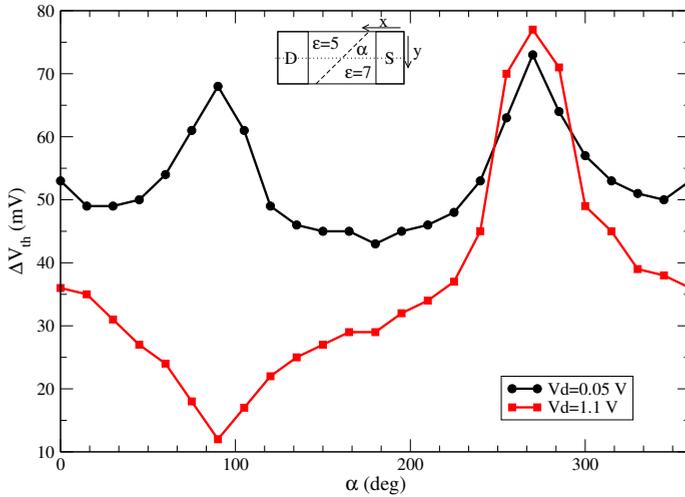


Figure A.12: Dependence of the threshold voltage displacement on the angle of the boundary between the two crystals and the x axis in the 25 nm gate length device.

A.4. Study of random dopant induced fluctuations using drift-diffusion simulations

As an example of the use of the atomistic meshes, we present a study of the impact of random dopants on the I_D - V_G characteristics of a 10 nm gate length DG thin body Si MOSFET (Fig. A.13) using drift-diffusion simulations.

The previously described meshing approach is employed to create an atomistic, high quality mesh for the Si body of the transistor. Then, a Monte Carlo approach is used to place dopants by replacing silicon atoms thus creating a particular distribution of random dopants following the continuous doping distribution [Ase98].

First, we compare the results obtained with the atomistic mesh and the results obtained with a very fine non-atomistic mesh. The results of the simulations of the device with continuous doping is shown in Fig. A.14. This confirms that the atomistic mesh is able to accurately capture the variations in the physical quantities. Fig. A.14 also shows the simulated I_D - V_G characteristics of the whole ensemble of DG MOSFETs with 10 nm gate length at a drain voltage of 0,05 V. There is a significant spread of the curves, reflected in the standard deviation of the threshold voltage, $\sigma V_T = 0,005$ V. In addition,

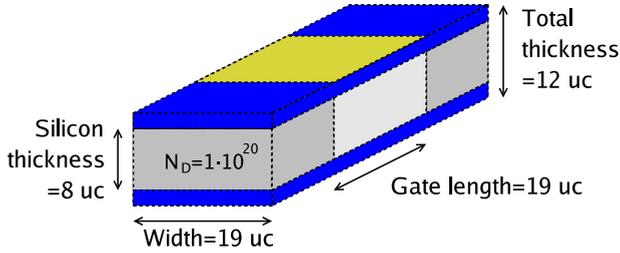


Figura A.13: Geometry of the simulated 10 nm gate length double gate MOSFET. Dimensions are in units of the Si unit cell (1 uc = 0,543 nm).

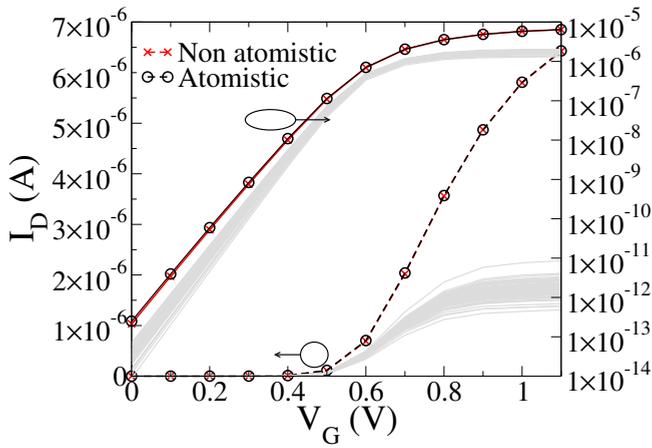


Figura A.14: I_D - V_G characteristics at a drain voltage of 0,05 V for an ensemble of simulated transistors. The drain current obtained using continuous doping with the non-atomistic (crosses) and atomistic (circles) meshes are highlighted.

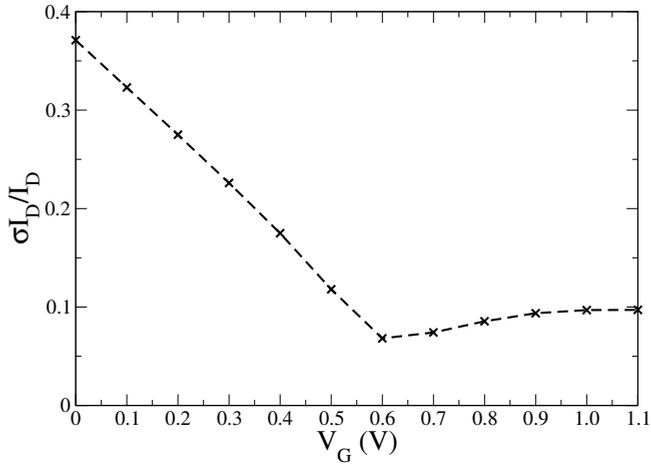


Figura A.15: Normalised standard deviation of the current as a function of the gate voltage for a drain bias of 0,05 V.

an impact of the fluctuations on the on-current has been also investigated. Fig. A.15 shows the normalised standard deviation of the drain current as a function of the gate bias at a low drain bias. The exponential-like behaviour of the current as a function of the gate bias in the subthreshold region results in a large (more than 35 %) impact of the random dopants. The fluctuations are diminishing with the increasing gate bias above the threshold voltage when the current is in Ohmic regime and then saturates. In the subthreshold region, it is common to express a spread in the drain current from the distribution of $\log(I_D)$ which gives a standard deviation of $\sigma \log(I_D) \simeq 0,17$. At a high gate voltage, the relative dispersion of the results, $\sigma I_D \simeq 10\%$, is lower as shown in Fig. A.15. Similarly, the difference between the average drain current obtained from the ensemble of simulations with the atomistically described source/drain and the current obtained from a simulation using a continuous doping in the source/drain becomes smaller at a high gate voltage as can be seen in Fig. A.16.

The large difference between the continuous and the average current in the atomistic simulations is due to the trapping of charge in the sharply resolved Coulomb wells associated with the individual donors. This is a drawback of the drift-diffusion approach which assumes either Boltzmann or Fermi-Dirac statistics and therefore heavily populates the wells. In the same time the ground state in the wells is very close to the conduction band edge and the heavy trapping inherent to the drift-diffusion approach is physically unrealistic. This

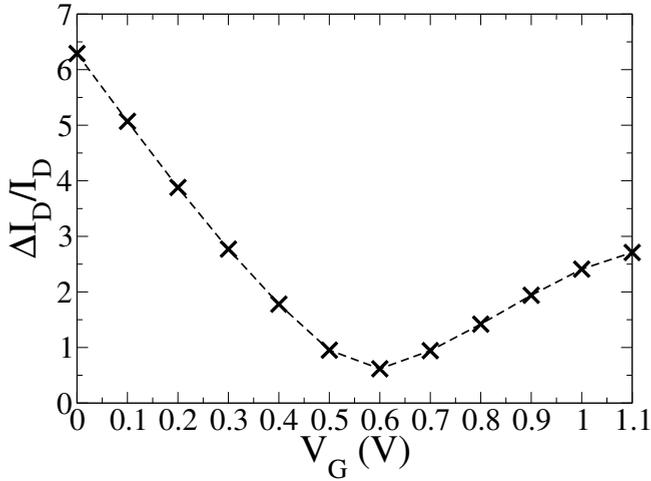


Figure A.16: Relative variation of the drain current for a drain bias of 0,05 V.

drawback of the drift-diffusion simulations can be remedied by the introduction of density gradient quantum corrections [RBAL⁺06]. The large variation of the on current in our simulation is a reflection of this artifact.

We have also compared the simulations of devices with atomistic doping using the atomistic and non-atomistic meshes. After simulating the ensemble of devices with the atomistic mesh, we chose the configuration of random dopants exhibiting the highest and the lowest currents. The devices were then simulated using two different charge assignment schemes on the non-atomistic mesh: (i) assigning charge of the dopant to the nearest node in the mesh (so-called *Nearest Grid Point, NGP* scheme) [HE88] and (ii) splitting the charge among all the neighbouring nodes (so-called *Cloud-in-Cell, CIC* scheme) [HE88]. The results of such simulations are shown in Fig. A.17. First, we can see the improvement in the convergence when using the atomistic mesh. The configuration of dopant atoms which produces the minimum current did not converge when using the non-atomistic mesh (uniform structured tetrahedral mesh) for the NGP charge assignment scheme. However, the simulations using the other configuration of dopants give very similar results when using both the atomistic and the non-atomistic (using NGP) meshes. This is expected due to the very fine non-atomistic mesh, since the distance between two nodes is small enough to minimise the correlation between dopant positions introduced by the charge assignment scheme. The discrepancy with the non-atomistic mesh using CIC

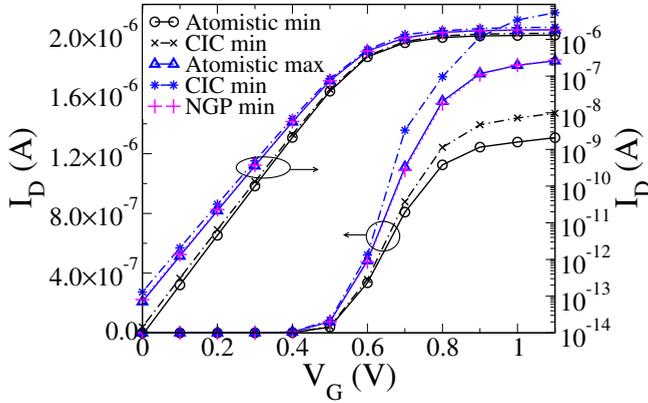


Figure A.17: I_D - V_G characteristics for the devices showing the lowest (min) and highest (max) currents using the atomistic mesh and the uniform mesh with two different charge assignments: cloud-in-cell (CIC) and the nearest grid point (NGP) scheme.

charge assignment is a consequence of a different charge trapping in the nodes originating from the different potential profile in the vicinity of the ionised impurities [RBAL⁺06] (discussed in the previous paragraph) which makes a fair comparison complicated.

A.5. Simulation of variability in InGaAs Implant-Free NanoMOSFETs using drift-diffusion simulations

The discreteness of charge and matter in nano-transistors is recognised as an important source of variability and has become a major bottleneck for their scaling and integration [WFS⁺99]. The implant free MOSFET promises robustness against dopants induced variability as dopants are separated by a spacer layer from the device channel. However, the proximity of the n -type, δ -doping plane may still significantly affect the electron transport in the channel.

We have simulated a 15 nm gate length implant free $\text{In}_{0,75}\text{Ga}_{0,25}\text{As}$ MOSFET with a δ -doping layer with concentration of $3 \times 10^{12} \text{cm}^{-2}$ below the channel as shown in Fig. A.18. The layer structure of the investigated 15 nm gate length IF MOSFET has a 1,5 nm thick $\text{GaO}_2/\text{GdGaO}$ high- κ gate stack with a metal gate (assuming a workfunction of 4,7 eV), a 1 nm thick $\text{In}_{0,52}\text{Al}_{0,48}\text{As}$

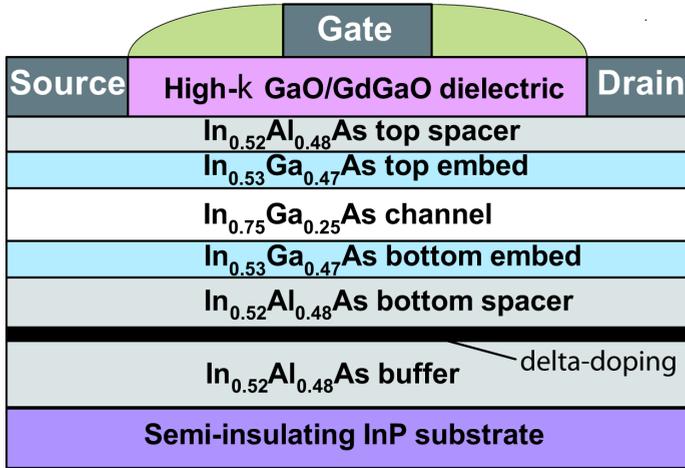


Figura A.18: Cross-section of n -type implant free $\text{In}_{0.75}\text{Ga}_{0.25}\text{As}$ MOSFET with a δ -doping layer below the channel.

oxide spacer, a 0,5 nm thick $\text{In}_{0.53}\text{Ga}_{0.47}\text{As}$ top embed layer, an $\text{In}_{0.75}\text{Ga}_{0.25}\text{As}$ channel with a thickness of 5 nm, a 1 nm thick bottom $\text{In}_{0.53}\text{Ga}_{0.47}\text{As}$ embed layer separated by a 2 nm thick $\text{In}_{0.52}\text{Al}_{0.48}\text{As}$ spacer from the δ -doping layer, all grown on a 49 nm $\text{In}_{0.52}\text{Al}_{0.48}\text{As}$ buffer as depicted in Fig. A.18.

Prior to the study of the influence of random dopants, the 3D D-D simulator has been calibrated against I_D - V_G characteristics at drain biases of 0,1 and 0,8 V obtained from a finite element heterostructure Monte Carlo device simulator [KDPA07] as shown in Fig. A.19. The calibration process yields a low field mobility of $5000 \text{ cm}^2/\text{Vs}$ and a saturation velocity of $6,5 \times 10^7 \text{ cm/s}$ in the channel of the implant free MOSFET.

After the calibration, we have simulated an ensemble of microscopically different implant free MOSFETs using the created atomistic mesh. The difference is due to the discrete nature and the random position of the dopants in the n -type, δ -doping layer. The different devices are obtained using a rejection technique from the device with continuous doping. For every node in the δ -doping layer, a Si dopant may replace an atom position in the InAlAs crystal lattice with a probability determined by the corresponding concentrations and enforced by a rejection technique.

Every random dopant configuration creates a different potential landscape which results in different electron concentration profiles along the channel leading to different I-V characteristics. This is illustrated in Fig. A.20, which

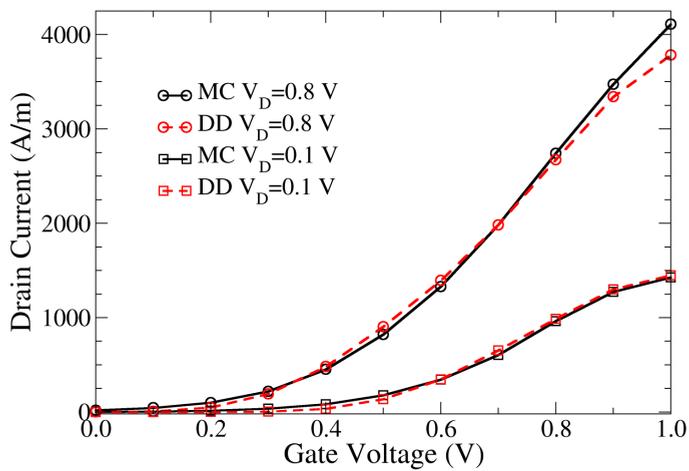


Figura A.19: Calibration of I_D - V_G characteristics at drain biases of 0,1 V and 0,8 V for the $\text{In}_{0,75}\text{Ga}_{0,25}\text{As}$ implant free MOSFET with a gate length of 15 nm against Monte Carlo device simulations.

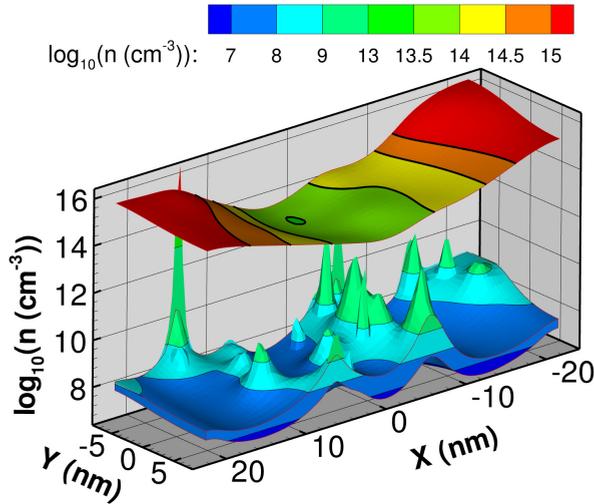


Figura A.20: Electron concentration, at equilibrium, in the δ -doping layer and at the bottom of the channel for a randomly generated pattern of dopants.

shows the effect of the random doping on the electron concentration at the bottom of the channel.

In this study, we have simulated more than 200 microscopically different transistors. Fig. A.21 shows the threshold voltage calculated with a constant current criterion [RBAL⁺06] and the corresponding number of dopants in the random region of the δ -doping layer for the whole ensemble. Although the number of simulations is not very high, a correlation between both variables can be identified. A lower number of dopants will result in a higher threshold voltage for the devices because of less available charge for the transport. Furthermore, a position dependence of the threshold voltage is also evident from the dispersion at a fixed dopant number. Fig. A.21 also shows the distribution of threshold voltage versus the number of dopants and a comparison with the continuous expected probability density function. The continuous doping values of the threshold voltage, the number of dopants, and the ensemble average of the random devices are shown for comparison. A threshold voltage shift towards higher values is observed originating from the artificial charge trapping in the neighbourhood of the dopant atoms which leads to a higher resistance. However, the obtained distribution is still qualitatively valid, giving a good indication of expected fluctuations associated with device electrosta-

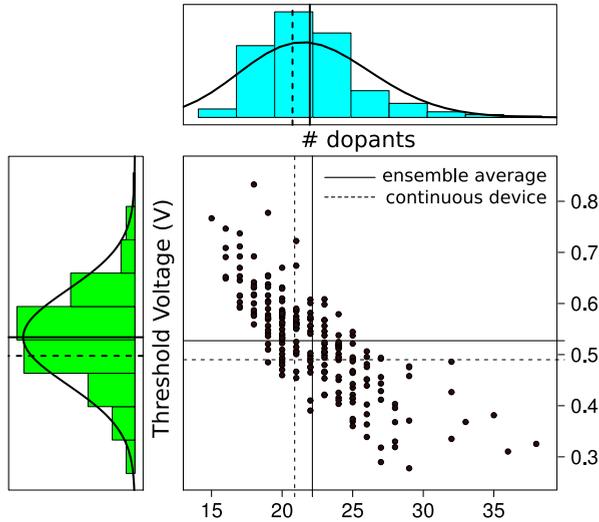


Figura A.21: Scatter plot showing the threshold voltage and the dopant number variability in the ensemble of simulated transistors. Histograms for the individual distributions are shown on the left and top sides of the figure. The expected probability density functions and the continuous device values are also indicated for comparison.

tics [ABD⁺03]. The spread in the threshold voltage values, $\sigma V_T = 94,5 \text{ mV}$, is similar compared to those published for equivalent gate length Si MOSFETs [RBAL⁺06] ($\sigma V_T = 100 \text{ mV}$).

A.6. Efficient parallel Monte Carlo simulations using finite element tetrahedral meshes applied to novel MOSFET architectures

With the scaling of semiconductor transistors to nanometre dimensions, bulk MOSFET architecture is reaching its limit. The bulk MOSFET is expected to be replaced with novel thin-body multigate architectures which may deliver the ITRS prescribed on-current [roa] in future scaling nodes. Therefore, an accurate physical modelling of these multigate transistors is of large interest especially using the ensemble Monte Carlo (MC) method [KWR03, RBA⁺07], one of the most widespread techniques for the modelling of carrier transport in nanoscale devices. However, it is computationally very expensive for complex

3D geometries of the thin-body architectures and its use in statistical studies (requiring hundreds of simulations) is often prohibitive, although a few studies have been done with constraints [RF02, RBA⁺07]. Therefore, the use of an optimal 3D mesh and parallel computers in order to save simulation time is desirable [RHLG94, BJLFD98].

For the initial solution, Poisson and current continuity equations are decoupled using Gummel methods and linearised using Newton's algorithm. All these algorithms are implemented fully in parallel manner as follows.

The linearised systems obtained from Poisson and current continuity equations are solved using a domain decomposition technique [BBC⁺94]. The solution domain Ω is partitioned in p subdomains Ω_i as

$$\Omega = \bigcup_{i=1}^p \Omega_i \tag{A.3}$$

and the domain decomposition methods attempt to solve the problem on the entire domain Ω by concurrent solutions on each subdomain Ω_i .

Each node belonging to a subdomain is an unknown of the whole problem. It is important to distinguish between three types of unknowns: (i) interior nodes, those that are coupled only with local nodes, (ii) local interface nodes, those coupled with external nodes as well as local nodes, and (iii) external interface nodes, which are those nodes in other subdomains coupled with local nodes. We label the nodes according to their subdomains, first the internal nodes and then both the local and external interface nodes. As a result, the linear system associated with the problem has the following structure:

$$\begin{pmatrix} B_1 & & & E_1 \\ & B_2 & & E_2 \\ & & \ddots & \\ & & & B_p & E_p \\ F_1 & F_2 & & F_p & C \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_s \\ y \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_s \\ g \end{pmatrix} \tag{A.4}$$

where, for the subdomain Ω_i , y represents the vector of all interface unknowns, B_i represents the equations of internal nodes, C the equations of interface nodes, E_i the subdomain to the interface coupling seen from the subdomains and F_i represents the interface to the subdomain coupling seen from the interface nodes.

We have used the program METIS to partition the mesh into subdomains. The same program was subsequently used to relabel the nodes in the subdomains in order to obtain a more suitable rearrangement to reduce the bandwidth of the matrix. The PPARSLIB [SLK97] parallel sparse iterative solvers library modified to take advantage of the reordering of the matrices has been used to solve the linear system (A.4). A great advantage of this library is its optimisation for various powerful multicomputers. Among many domain decomposition techniques supported within this library the best result was obtained when using the additive Schwarz technique [SGL06]. Finally, the resulting local linear systems are solved using standard ILUT preconditioner combined with FGMRES [Saa96].

However, in general, this is not the best way to share the load of the MC engine. We have to perform a load balancing since the regions with the highest electron density will have the highest number of particles to simulate. Furthermore, they will have more scattering events (ionised impurity scattering in regions of high doping and interface roughness scattering in the inversion layer) and, as a consequence, more computational weight, although a partitioning algorithm weighted using the electron density profile could help to overcome this difficulty. With this in mind, we have chosen a different strategy to parallelise the MC engine in the simulation. The main idea is to replicate the whole mesh and simulate an equal number of particles in each processor. Although this requires the replication of the mesh a number of times, this will not increase the use of the total memory required by the program significantly since even for a relatively big mesh it requires only a few MB of memory. The other drawback of the strategy is the higher number of communications, but we think that this is compensated by their regularity and the balancing of the computations.

The investigations of parallel performance of the 3D MC device code have been carried out by simulating the DG MOSFET over a small period of 20 fs. All the results presented here were obtained on an HP Superdome cluster with 128 Itanium 2 1.5 GHz processors. The scalability of the program can be characterised by the speedup (S) or the parallel efficiency (PE), defined for p processors as:

$$S(p) = \frac{t_1}{t_p} \quad PE(p) = \frac{t_1}{t_p p} \quad (\text{A.5})$$

where t_1 and t_p are the times employed running the simulation on one and p processors, respectively. Fig. A.22 shows their dependence with the number of processors. We can see that their values are over the linear scaling limit.

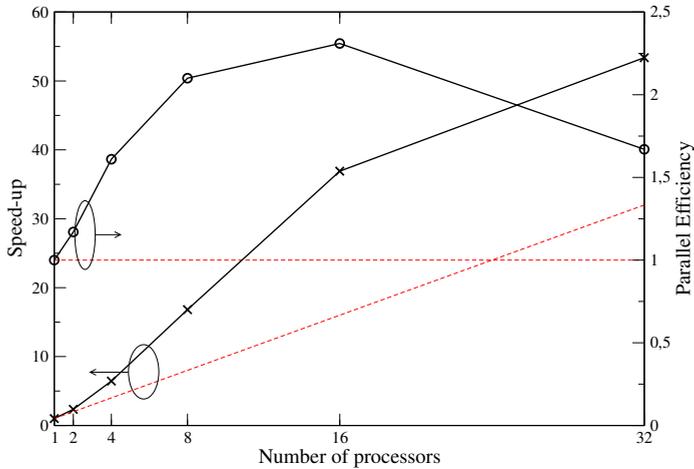


Figure A.22: Speedup (crosses) and parallel efficiency (circles) of one time step. We also show the ideal values (dashed lines) for comparison.

This effect is due to the efficiency of Poisson solver and it starts to vanish for a high number of processors when operations corresponding to non scaling stages (e.g., the electric field update) begin to be of the same order of magnitude than the scaling operations (the linear system solving stage or the particle flights). This limit is intrinsic to our parallelisation strategy and cannot be avoided. Its impact for a given number of processors will mainly depend on the size of the employed mesh.

Table A.3 shows the execution times of different parts of the main iteration in the Monte Carlo loop. We obtain a very good scalability for the particle flight times being limited by the contacts stage, where communications and replicated code are required in order to maintain the equilibrium. The other limiting stage is the update of the electric field, which requires a constant time independently of the number of processors. These two stages remain unimportant for a low number of processors, but they limit the scalability for a high number of processors, when these times are of the same order of magnitude or even larger than those of the MC free flights or the solution of Poisson equation.

To reflect quantum mechanical confinement effects, we have implemented the density gradient equation for electrons discretised using the finite element method [GLKA07]. The quantum potential V_Q is then used to correct the force

Table A.3: Mean simulation times (s) for one time step ($t_{iteration}$), linear Poisson solver ($t_{Poisson}$), particle flights (t_{flight}), contacts stage ($t_{contacts}$), reduction of electron concentration ($t_{reduction}$), communication of the potentials ($t_{comun.poten.}$) and electric field update (\vec{E}_{update}) for different number of processors (n_p)

| n_p | $t_{iteration}$ | $t_{Poisson}$ | t_{flight} | $t_{contacts}$ | $t_{reduction}$ | $t_{comun.poten.}$ | \vec{E}_{update} |
|-------|-----------------|---------------|--------------|----------------|-----------------|--------------------|--------------------|
| 1 | 27.50 | 24.9 | 0.68 | 0.35 | 0.013 | 0.019 | 0.207 |
| 2 | 11.74 | 11.1 | 0.33 | 0.20 | 0.010 | 0.007 | 0.204 |
| 4 | 4.27 | 3.87 | 0.173 | 0.146 | 0.0085 | 0.0054 | 0.203 |
| 8 | 1.64 | 1.08 | 0.102 | 0.212 | 0.0188 | 0.0050 | 0.198 |
| 16 | 0.745 | 0.386 | 0.0420 | 0.101 | 0.005 | 0.0042 | 0.200 |
| 32 | 0.5152 | 0.186 | 0.0209 | 0.079 | 0.009 | 0.006 | 0.206 |

on the particles solving the following equation:

$$2b_n \frac{\nabla^2 \exp(u)}{\exp(u)} = 2u - V + V_n, \quad (\text{A.6})$$

where $b_n = \frac{\hbar^2}{12m^*q}$ and the new variable $u = \frac{1}{2}(V - V_n + V_Q)$ has been introduced. A finite element discretisation is then applied resulting in the following system of equations:

$$\begin{aligned} & \int_{\partial\Omega} 2b_n \theta_i \nabla(u) \vec{n} dS - \\ & \int_{\Omega} 2b_n (\nabla(\theta_i) - \theta_i \nabla(u)) \nabla(u) d\Omega = \\ & \int_{\Omega} (2u - V + V_n) \theta_i d\Omega, \end{aligned} \quad (\text{A.7})$$

where θ_i are the finite element basis functions. The use of the discretisation based on potentials instead of that based on carrier densities,

$$V_Q = 2b \frac{\nabla^2 \sqrt{n}}{\sqrt{n}}, \quad (\text{A.8})$$

allows a much smoother behaviour for electron density, especially, when it is used self-consistently [wTW04, SMGGR06]. However, initially, we calculate

the correction only once from the solution of the drift-diffusion model used for the initialisation of the particles. Then, during the Monte Carlo simulation, we calculate the electric field used to move the carriers as

$$\mathbf{E} = -\nabla(V + V_Q^{FF}). \quad (\text{A.9})$$

where V_Q^{FF} stands for a quantum potential in the frozen field approximation. Although not self-consistent, it has been shown that this approximation is very close to the fully self-consistent approximation at low drain biases [RA08].

One extra advantage of the quantum corrections in particle simulations is that the displacement of the peak charge density from the interface brings numerical advantages, since collision detections with the interface are greatly diminished. This is especially important for non-cartesian domains because they require a high number of operations.

A.7. Reduction of the self-forces for Monte Carlo simulation of semiconductor devices on general unstructured meshes

In particle simulations where the field calculation on a mesh is self-consistently coupled to the movement of the particles, unphysical forces may appear. For example, in the Monte Carlo simulation of semiconductor devices, the particle charge is assigned to the nodes of the mesh, the Poisson equation is solved with this charge and the electric field obtained from the electrostatic potential is interpolated in the position of the particles. In this process, a particle can feel a force even if it is the only charge in an infinite discretised domain. This unphysical force is called self-force [Lau96, HE88], and can lead to spurious results if it is not taken into account in the simulation process.

In [HE88], two conditions are used to prove that the self-force is zero:

1. The charge assignment and the force interpolation schemes must be the same. Let $\phi(\mathbf{r} - \mathbf{r}_p)$ be the interpolation function associated with the node p of the considered element in the point \mathbf{r} . The contribution to the charge density in the node \mathbf{r}_p from a charge q in \mathbf{r} will be

$$\rho(\mathbf{r}_p) = \frac{q}{V} \phi(\mathbf{r} - \mathbf{r}_p), \quad (\text{A.10})$$

where V is a volume. This volume has to satisfy particular conditions which will be derived later. The electric field at the point \mathbf{r} will be

$$\mathbf{E}(\mathbf{r}) = \sum_{p \in \text{nodes}} \phi(\mathbf{r} - \mathbf{r}_p) \mathbf{E}(\mathbf{r}_p). \quad (\text{A.11})$$

2. The electric field defined on the nodes of the mesh must be spherically symmetric. The electric field in a node \mathbf{r}_i generated by a charge in the node \mathbf{r}_j must have the same magnitude and opposite direction than the electric field in the node \mathbf{r}_j generated by the same charge in the node \mathbf{r}_i . Assuming that there is a linear dependence between the electric field and the charge density,

$$\mathbf{E}(\mathbf{r}_p) = \sum_{p' \in \text{nodes}} G(\mathbf{r}_p; \mathbf{r}_{p'}) \rho(\mathbf{r}_{p'}), \quad (\text{A.12})$$

where the proportionality matrix must be anti-symmetric:

$$G(\mathbf{r}_p; \mathbf{r}_{p'}) = -G(\mathbf{r}_{p'}; \mathbf{r}_p). \quad (\text{A.13})$$

For tetrahedral meshes, condition 1 can easily be verified if the values of the electric field in the nodes of the mesh are known, for example, using the finite element interpolation functions. For tetrahedral elements, we have the following linear interpolation functions [BCO81]:

$$\begin{cases} \phi(\mathbf{r} - \mathbf{r}_p) = \frac{v_p}{V_{\text{element}}} & \mathbf{r} \in \Omega_{\text{element}} \\ \phi(\mathbf{r} - \mathbf{r}_p) = 0 & \text{otherwise,} \end{cases} \quad (\text{A.14})$$

where V_{element} is the volume of the element containing the point and v_p is the volume of the sub-tetrahedron formed by the face opposite to the node \mathbf{r}_p and the interior point \mathbf{r} .

Condition 2 holds if the value of the electric field in the nodes is exact. However, the values obtained for the electric field will be approximations and will not completely satisfy this condition. This will lead to unwanted self-forces in the device, with a magnitude dependent on the accuracy of the value of the electric field.

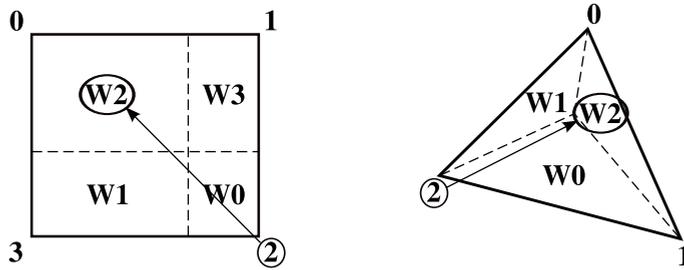


Figura A.23: Geometrical meaning of charge assignment schemes for a Cartesian element (left) and a simplex element (right) in 2D. w_i is the weight associated to node i and corresponds to the area of the sub-element the label is on. Sub-elements are defined by the point x , which would be the position of the charge.

A.7.1. Charge assignment

The use of (A.14) for the charge assignment appears naturally in the discretisation of the Poisson equation with the finite element method using point charges. In this case the electron density can be written as:

$$n = \sum_{k \in p} q_k \delta(\mathbf{r}_k). \tag{A.15}$$

The weak form of this equation is:

$$\int_{\Omega} n v \, d\Omega = \int_{\Omega} \sum_{k \in p} q_k \delta(\mathbf{r}_k) v \, d\Omega. \tag{A.16}$$

Using the Galerkin approximation for v , $v = \sum v_i \theta_i$, and transforming the expression for the shape functions, we obtain the following contribution for a given node i :

$$\int_{\Omega} \sum_{k \in p} q_k \delta(\mathbf{r}_k) \theta_i \, d\Omega = \int_{\Omega} \sum_{k \in p} q_k \delta(\mathbf{r}_k) \sum_{e \ni i} \phi_i^e \, d\Omega = \sum_{k \in p} \sum_{e \ni i} q_k \phi_i^e(\mathbf{r}_k). \tag{A.17}$$

Fig. A.23 shows an example in two dimensions of this charge assignment and compares it with the standard cloud-in-cell scheme used in finite difference meshes.

A.7.2. Electric field interpolation

When we solve the Poisson equation, what we obtain are the values of the electrostatic potential. From these values we can compute the gradient in finite element meshes and obtain the electric field in the elements. However, to apply the previously described interpolation scheme, we must have the electric field defined at the nodes.

The most consistent way of computing the electric field is the use of the finite element formulation on the equation:

$$\mathbf{E} = -\nabla\psi. \quad (\text{A.18})$$

Formulating the problem in variational form and using the Galerkin approximation we get to the following system of vector equations:

$$\sum_{j=1}^N \left[\mathbf{E}_j \int_{\Omega} \theta_j \theta_i d\Omega + \psi_j \int_{\Omega} \nabla \theta_j \theta_i d\Omega \right] = 0 \quad \forall i = 1, \dots, N. \quad (\text{A.19})$$

A more economic way of computing the electric field is the interpolation from the values calculated on the elements as $\mathbf{E} = -\nabla\psi$. For an element e we have:

$$\mathbf{E}_e = -\nabla\psi_e = -\nabla \left(\sum_{i \in e} \psi_i \theta_i \right) = - \sum_{i \in e} \psi_i \nabla \theta_i. \quad (\text{A.20})$$

From these values we can obtain the electric field on the nodes as:

$$\mathbf{E}(\mathbf{r}_i) = \sum_{e \ni i} w_e \mathbf{E}_e. \quad (\text{A.21})$$

We found that the best results were obtained with the following interpolation:

$$\mathbf{E}(\mathbf{r}_i) = \sum_j \frac{A}{d_j} \mathbf{E}_j \quad (\text{A.22})$$

where the sum is over all neighbouring tetrahedra, \mathbf{E}_j is the electric field in the element j , and d_j is the distance from the node \mathbf{r}_i to the barycentre of the neighbouring tetrahedron j . The constant A is chosen to obtain $\sum \frac{A}{d_j} = 1$ for each node, so it is $A = \left(\sum \frac{1}{d_j} \right)^{-1}$.

Firstly, for an structured tetrahedral mesh, we calculated the normalised proportionality matrix G between the components of the electric field and

the electron density in the four nodes of an element, $E_i = \sum_j G_{ij}q_j$. As an example, using the simplest interpolation, the following normalised G_x matrix was obtained for the x component of the electric field,

$$G_x \simeq \begin{pmatrix} 4,1 \cdot 10^{-8} & -0,22826 & -0,32240 & -0,99933 \\ 0,22827 & 1,1 \cdot 10^{-5} & -0,16331 & -0,51641 \\ 0,32241 & 0,16333 & 1,1 \cdot 10^{-5} & -0,52093 \\ 1,00000 & 0,51709 & 0,52161 & 6,8 \cdot 10^{-4} \end{pmatrix}$$

The property of anti-symmetry, one requirement for the zero self-force, is satisfied by this approximation with an accuracy of almost 0.1 %.

After this we obtained this proportionality matrix on an unstructured mesh. With the simple interpolation of the electric field we find the following matrix:

$$G^x \simeq \begin{pmatrix} 0,01719 & -0,70076 & -0,14044 & 0,05078 \\ 0,70327 & -0,11414 & 0,72234 & 1,00000 \\ 0,20840 & -0,77500 & 0,03204 & 0,31582 \\ -0,01774 & -0,99745 & -0,32233 & 0,03744 \end{pmatrix}$$

In this case, the matrix is further from antisymmetry, with a deviation of almost 100 % in one of the elements (G_{41}^x , G_{14}^x). To see if the finite element calculation of the electric field would improve this behaviour, we obtained the matrix for this scheme:

$$G^x \simeq \begin{pmatrix} -0,00785 & -0,69734 & -0,04820 & 0,06307 \\ 0,67289 & -0,06546 & 0,79161 & 0,94406 \\ 0,19568 & -0,84617 & 0,09457 & 0,36391 \\ -0,06996 & -1,00000 & -0,26227 & -0,01267 \end{pmatrix}$$

We can see that the behaviour is as irregular as in the previous case and the extra cost of the calculation does not compensate the possible improvement.

A.7.3. Double gate MOSFET simulation

To test our approach and to measure the self-forces, we have chosen a double gate (DG) metal-oxide-semiconductor field-effect transistor (MOSFET) structure because of its high symmetry and also because this multigate transistor architecture is expected to replace the conventional bulk MOSFET beyond 32 nm technology generation [roa]. We know that in the centre of this structure the effect of the boundaries should compensate one another, so that

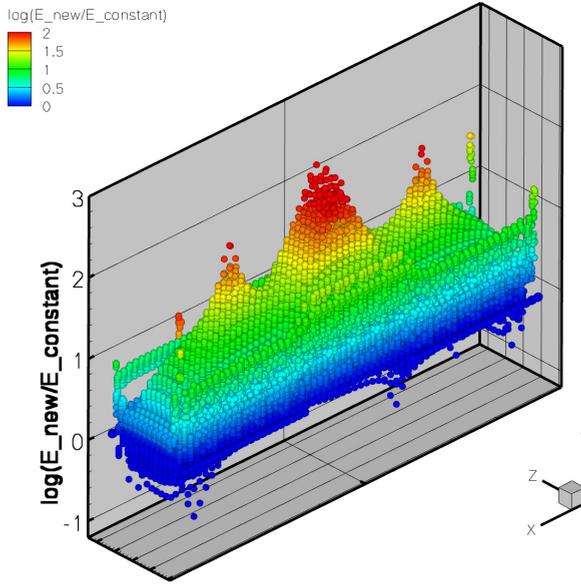


Figura A.24: Reduction of the electric field in orders of magnitude at the positions of a particle in the 10 nm gate length, 6,1 nm body thick, DG Si MOSFET.

the self-force in the centre should be the smallest or close to zero if our scheme works. The data were obtained by solving the Poisson equation in the device populated by only one electron, once for every element of the discretised domain. The electric field in the position of the particle is obtained using two different methods. One of them employs the previously described simple interpolation using linear functions. The other one is based on the approximation of constant electric field in the elements, which is the one usually described and used [BS94, HE88, Lau96].

Fig. A.24 shows the reduction in the self-force for the 10 nm gate length DG Si MOSFET. The most important improvement is far from the boundaries, in the centre of the device, since the contribution to the electric field here is mainly due to the charge assignment. However, near the boundaries, the main contribution comes from the break of symmetry in the computed electrostatic potential, so the reduction in the electric field is lower. Figs. A.25-A.26 show the magnitude of the electric field on a particle in different regions of the device. These figures demonstrate a correlation between the magnitude of

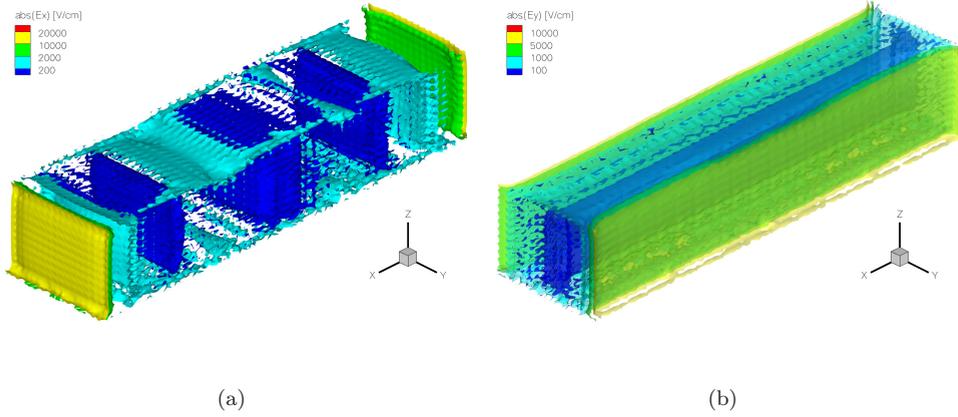


Figura A.25: Isosurfaces of the (a) x component and (b) y component of the electric field on a particle positioned in different places inside the device.

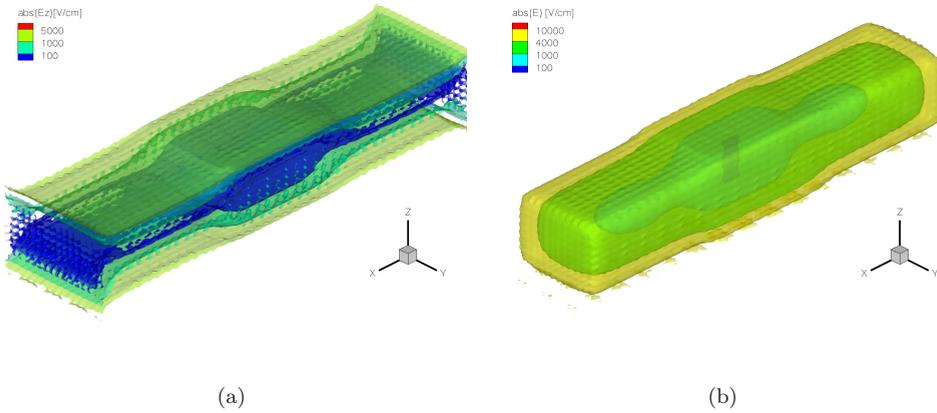


Figura A.26: Isosurfaces of the (a) x component and (b) absolute value of the electric field on a particle positioned in different places inside the device.

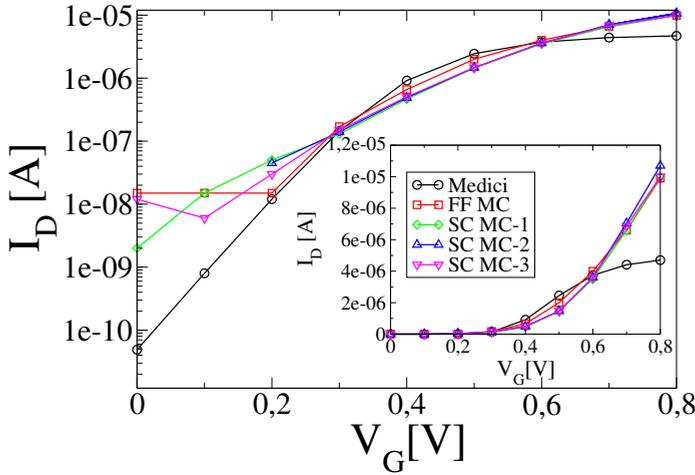


Figura A.27: I_D - V_G characteristics at a drain voltage of $V_D = 0,05$ V for different charge assignment/electric field interpolation schemes

a component of the electric field and the distance to the boundaries perpendicular to it. This is because the strongest break in the spherical symmetry takes place in that direction. The resulting module of the electric field is also shown. There is a clear correlation between the region of highest improvement of the electric field and the region with the lowest electric field. This suggests that the reduction comes from the electric field associated to the interpolation (self-force), confirming that the combined scheme for the charge assignment/electric field interpolation does reduce the self-forces coming from the decomposition of the domain into tetrahedra.

Finally, we evaluated the impact of the self-forces on the current characteristics of the device. Fig. A.27 shows a comparison between the I_D - V_G characteristics of the device obtained from the simulation using different charge assignment schemes. In one of them, which does not lead to zero self-forces (SC MC-1 in Fig. A.27), the charge of the particle is split among the nodes of the element enclosing it using the interpolation function (A.14). Then, to obtain the density required by the Poisson equation, this value is divided by the volume associated to the node, i.e., the sum of one fourth of the volume of every tetrahedron containing that node. The other method is the one proposed in this work (section A.7.1, SC MC-3 in Fig. A.27). Finally, the third set of results is obtained using a constant electric field in the elements calculated

according to equation (A.20) (SC MC-2 in Fig. A.27). The self-forces are the most pronounced when the background electric field is low. Therefore, we just show the curves corresponding to a low drain voltage of $V_D = 0,05 V$ and the values of the gate voltage in the range $0,1 V - 0,8 V$, since very low polarisation current values are also very difficult to obtain accurately within the MC method. We can see a better behaviour of the current with the second charge assignment scheme attributable to its better behaviour regarding to minimal self-forces. However, since the mesh does not have large gradients in the size of the elements and the electric fields inside the device are higher, the current obtained with both assignment schemes are very similar at large gate voltages.

Bibliografía

- [ABD⁺03] A. Asenov, A.R. Brown, J.H. Davies, S. Kaya, and G. Slavcheva. Simulation of intrinsic parameter fluctuations in decananometer and nanometer-scale MOSFETs. *IEEE Trans. Electron Devices*, 50(9):1837–1852, 2003.
- [ACC⁺08] A. Asenov, A. Cathignol, B. Cheng, K.P. McKenna, A.R. Brown, A.L. Shluger, D. Chanemougame, K. Rochereau, and G. Ghibaud. Origin of the asymmetry in the magnitude of the statistical variability of n- and p-channel poly-Si gate bulk MOSFETs. *IEEE Electron Device Lett.*, 29(8):913–915, 2008.
- [AI89] M. G. Ancona and G. J. Iafrate. Quantum correction to the equation of state of an electron gas in a semiconductor. *Phys. Rev. B*, 39(13):9536–9540, May 1989.
- [ALD96] N. R. Aluru, K. H. Law, and R. W. Dutton. Simulation of the hydrodynamic device model on distributed memory parallel computers. *IEEE Trans. Comput-Aided Des. Integr. Circuits Syst.*, 15(9):1029–1047, 1996.
- [AMP00] D. N. Arnold, A. Mukherjee, and L. Pouly. Locally adapted tetrahedral meshes using bisection. *SIAM J. Sci. Comput.*, 22(2):431–448, 2000.
- [APGL06] M. Aldegunde, J. J. Pombo, and A. J. García-Loureiro. Modified octree mesh generation for Manhattan type structures with narrow layers applied to semiconductor devices. *Int. J. Numer. Model.-Electron. Netw. Device Fields*, 19(6):473–489, 2006.

- [aps] APSYS, Advanced Physical Models of Semiconductor Devices. http://www.crosslight.com/Product_Overview/prod_overv.html#APSYS.
- [ARA08] C. Alexander, G. Roy, and A. Asenov. Random-dopant-induced drain current variation in nano-MOSFETs: A three-dimensional self-consistent Monte Carlo simulation study using “ab initio” ionized impurity scattering. *IEEE Trans. Electron Devices*, 55(11):3251–3258, 2008.
- [AS00] A. Asenov and S. Saini. Polysilicon gate enhancement of the random dopant induced threshold voltage fluctuations in sub-100 nm MOSFETs with ultrathin gate oxide. *IEEE Trans. Electron Devices*, 47(4):805–812, 2000.
- [Ase98] A. Asenov. Random dopant induced threshold voltage lowering and fluctuations in sub-0.1 μm MOSFET’s: A 3-D “atomistic” simulation study. *IEEE Trans. Electron Devices*, 45(12):2505–2513, 1998.
- [AT87] M. G. Ancona and H. F. Tiersten. Macroscopic physics of the silicon inversion layer. *Phys. Rev. B*, 35(15):7959–7965, May 1987.
- [atl] ATLAS: Device Simulation Framework. http://www.silvaco.com/products/device_simulation/atlas.html.
- [AW01] W. Fichtner, A. Wettstein, A. Schenk. Quantum device-simulation with the density-gradient model on unstructured grids. *IEEE Trans. Electron Devices*, 48(2):279–284, 2001.
- [Bar73] J. R. Barker. Quantum transport theory of high-field conduction in semiconductors. *J. Phys. C: Solid State Phys.*, 6(17):2663–2684, 1973.
- [BAW02] A. R. Brown, A. Asenov, and J. Watling. Intrinsic fluctuations in sub 10-nm double-gate MOSFETs introduced by discreteness of charge and matter. *IEEE Trans. Nanotech.*, 1(4):195–200, 2002.
- [BAY⁺03] F.M. Bufler, Y. Asahi, H. Yoshimura, C. Zechner, A. Schenk, and W. Fichtner. Monte Carlo simulation and measurement of nanos-

- cale n-MOSFETs. *IEEE Trans. Electron Devices*, 50(2):418–424, 2003.
- [BBC⁺94] R. Barrett, M. Berry, T. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. van der Vorst. *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*. SIAM, 1994.
- [BCO81] E. B. Becker, G. F. Carey, and J. T. Oden. *Finite Elements*. Prentice-Hall, Englewood Cliffs, New Jersey 07632, 1981.
- [BGG⁺08] Giorgio Baccarani, Elena Gnani, Antonio Gnudi, Susanna Reggiani, and Massimo Rudan. Theoretical foundations of the quantum drift-diffusion and density-gradient models. *Solid-State Electron.*, 52(4):526–532, April 2008.
- [BJLFD98] F. Banse, J.-L. Dekeyser, R. Fauquenbergue, and F. Dessenne. Implementation of a bi-parallel Monte Carlo device simulation on two architectures. *Lect. Notes Comput. Sc.*, 1401:193–202, 1998.
- [BOC02] S. G. Badcock, A. G. O’Neill, and E. G. Chester. Device and circuit performance of SiGe/Si MOSFETs. *Solid-State Electron.*, 46:1925–1932, 2002.
- [Bow81] A. Bowyer. Computing Dirichlet tessellations. *The Computer Journal*, 24:162–166, 1981.
- [BP03] A. Bongiorno and A. Pasquarello. Atomistic structure of the Si(100)-SiO₂ interface: A synthesis of experimental data. *Appl. Phys. Lett.*, 83(7):1417–1419, 2003.
- [BPP00] R. Buczko, S. J. Pennycook, and S. T. Pantelides. Bonding arrangements at the Si-SiO₂ and SiC-SiO₂ interfaces and a possible origin of their contrasting properties. *Phys. Rev. Lett.*, 84(5):943–946, Jan 2000.
- [BR80] R.E. Bank and D. J. Rose. Parameter selection for Newton-like methods applicable to nonlinear partial differential equations. *SIAM J. Numer. Anal.*, 17(6):806–822, 1980.

- [BR81] R. E. Bank and D. J. Rose. Global approximate Newton methods. *Numer. Math.*, 37(2):279–295, 1981.
- [BS50] J. Bardeen and W. Shockley. Deformation potentials and mobilities in non-polar crystals. *Phys. Rev.*, 80(1):72–80, 1950.
- [BS94] J. Bude and R. K. Smith. Phase-space simplex Monte Carlo for semiconductor transport. *Semicond. Sci. Technol.*, 9:840–843, 1994.
- [BWD84] G. Baccarani, M.R. Wordeman, and R.H. Dennard. Generalized scaling theory and its application to a 1/4 micrometer MOSFET design. *IEEE Trans. Electron Devices*, 31(4):452–462, 1984.
- [CBBG96] J. P. Colinge, X. Baie, V. Bayot, and E. Grivei. A silicon-on-insulator quantum wire. *Solid-State Electron.*, 39(1):49 – 51, 1996.
- [CC03] G. K. Celler and S. Cristoloveanu. Frontiers of silicon-on-insulator. *J. Appl. Phys.*, 93(9):4955–4978, 2003.
- [CDD⁺05] R. Chau, S. Datta, M. Doczy, B. Doyle, B. Jin, J. Kavalieros, A. Majumdar, M. Metz, and M. Radosavljevic. Benchmarking nanotechnology for high-performance and low-power logic transistor applications. *IEEE Trans. Nanotechnol.*, 4(2):153–158, 2005.
- [CQ81] D. Chattopadhyay and H. J. Queisser. Electron scattering by ionized impurities in semiconductors. *Rev. Mod. Phys.*, 53(4):745–768, 1981.
- [CSY⁺07] K. H. Cho, S. D. Suk, Y. Y. Yeoh, M. Li, K. H. Yeo, D.-W. Kim, D. Park, W.-S. Lee, Y. C. Jung, B. H. Hong, and S. W. Hwang. Temperature-dependent characteristics of cylindrical gate-all-around twin silicon nanowire MOSFETs (TSNWFETs). *IEEE Electron Device Lett.*, 28(12):1129–1131, 2007.
- [CT67] D.M. Caughey and R.E. Thomas. Carrier mobilities in silicon empirically related to doping and field. *Proc. IEEE*, 55(12):2192–2193, 1967.

- [CWAA⁺06] J. Cervenka, W. Wessner, E. Al-Ani, T. Grasser, and S. Selberherr. Generation of unstructured meshes for process and device simulation by means of partial differential equations. *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, 25(10):2118–2128, 2006.
- [CZW⁺03] Yi Cui, Z. Zhong, D. Wang, W. U. Wang, and C. M. Lieber. High performance silicon nanowire field effect transistors. *Nano Lett.*, 3(2):149–152, 2003.
- [Dat00] S. Datta. Nanoscale device modeling: the Green’s function method. *Superlattices Microstruct.*, 28:253–278, 2000.
- [Dat07] S. Datta. III-V field-effect transistors for low power digital logic applications. *Microelectron. Eng.*, 84(9-10):2133–2137, 2007.
- [Del34] B. Delaunay. Sur la sphère vide. *Bulletin of the Academy of Sciences of the USSR VII, Class Sci.Mat. Nat.*, 6:793–800, 1934.
- [DeM68a] A. DeMari. An accurate numerical steady-state one-dimensional solution of the P-N junction. *Solid-State Electron.*, 11:33–58, 1968.
- [DeM68b] A. DeMari. An accurate numerical steady-state one-dimensional solution of the P-N junction under arbitrary transient conditions. *Solid-State Electron.*, 11:1021–1053, 1968.
- [DGKY72] R. H. Dennard, F. H. Gaensslen, L. Kuhn, and H. N. Yu. Design of micron MOS switching devices. In *1972 International Electron Devices Meeting*, pages 168–170, 1972.
- [DGR⁺74] R.H. Dennard, F.H. Gaensslen, V.L. Rideout, E. Bassous, and A.R. LeBlanc. Design of ion-implanted MOSFET’s with very small physical dimensions. *IEEE J. Solid-State Circuit*, 9(5):256–268, 1974.
- [Dir50] G. L. Dirichlet. Über die reduction der positiven quadratischen formen mit drei unterstimmten ganzen zahlen. *Z. Angew. Math. Mech.*, 40:209–227, 1850.

- [DLGC98] J. Dompierre, P. Labbé, F. Guibault, and R. Camarero. Proposal of benchmarks for 3D unstructured tetrahedral mesh optimization. In *Proceedings of the 7th International Meshing Roundtable*, pages 459–478, 1998.
- [dRW86] T. G. Van de Roer and F. P. Widdershoven. Ionized impurity scattering in Monte Carlo calculations. *J. Appl. Phys.*, 59(3):813–815, 1986.
- [DW06] Qiang Du and Desheng Wang. Recent progress in robust and quality delaunay mesh generation. *J. Comput. Appl. Math.*, 195(1-2):8–23, 2006.
- [FAV00] D.K. Ferry, R. Akis, and D. Vasileska. Quantum effects in MOSFETs: use of an effective potential in 3D Monte Carlo simulation of ultra-short channel devices. In *2000 International Electron Devices Meeting Technical Digest.*, pages 287–290, 2000.
- [FCBA06] D. Fischer, A. Curioni, S. Billeter, and W. Andreoni. The structure of the SiO₂/Si(100) interface from a restraint-free search using computer simulations. *Appl. Phys. Lett.*, 88:012101, 2006.
- [FDE⁺01] D. Frank, R. Dennard, E. Nowak, P. Solomon, Y. Taur, and H. S. Wong. Device scaling limits of Si MOSFETs and their application dependencies. *Proc. IEEE*, 89:227–239, 2001.
- [FG00] P. J. Frey and P.-L. George. *Mesh Generation*. Hermes Science, 2000.
- [Fis03] M. V. Fischetti. Scaling MOSFETs to the limit: A physicists’s perspective. *J. Comput. Electron.*, 2:73–79, 2003.
- [Fra02] D. Frank. Power-constrained CMOS scaling limits. *IBM J. Res. & Dev.*, 46(2/3):235–244, 2002.
- [FSVF02] G. F. Formicone, M. Saraniti, D. Z. Vasileska, and D.K. Ferry. Study of a 50 nm nMOSFET by ensemble Monte Carlo simulation including a new approach to surface roughness and impurity scattering in the Si inversion layer. *IEEE Trans. Electron Devices*, 49:125–132, 2002.

- [Gar94] C. L. Gardner. The quantum hydrodynamic model for semiconductor devices. *SIAM J. Appl. Math.*, 54:409–427, 1994.
- [GHFY06] J. Goldberger, A. I. Hochbaum, R. Fan, and P. Yang. Silicon vertically integrated nanowire field effect transistors. *Nano Lett.*, 6(5):973–977, 2006.
- [GHS91] P. L. George, F. Hecht, and E. Saltel. Automatic mesh generator with specified boundary. *Computer Methods in Applied Mechanics and Engineering*, 92:269–288, 1991.
- [GLKA05] A. J. García-Loureiro, K. Kalna, and A. Asenov. Efficient three-dimensional parallel simulations of PHEMTs. *Int. J. Numer. Model.-Electron. Netw. Device Fields*, 18(5):327–340, 2005.
- [GLKA07] A. J. García-Loureiro, K. Kalna, and A. Asenov. Implementation of a quantum corrections in a 3D parallel drift-diffusion simulator. In *Proceedings of the 2007 Spanish Conference on Electron Devices*, pages 60–63, 2007.
- [GP96] T. González and D. Pardo. Physical models of ohmic contact for Monte Carlo device simulation. *Solid-State Electron.*, 39:555–562, 1996.
- [GP05] F. Giustino and A. Pasquarello. Theory of atomic-scale dielectric permittivity at insulator interfaces. *Phys. Rev. B*, 71(14):144104, 2005.
- [GR88] Lester Guttman and Shafiqur M. Rahman. Simulation of the structure of amorphous silicon dioxide. *Phys. Rev. B*, 37(5):2657–2668, 1988.
- [GRCC⁺01] F. Gámiz, J. B. Roldán, P. Cartujo-Cassinello, J. A. López-Villanueva, and P. Cartujo. Role of surface-roughness scattering in double gate silicon-on-insulator inversion layers. *J. Appl. Phys.*, 89(3):1764–1770, 2001.
- [GS78] P. J. Green and R. Sibson. Computing Dirichlet tessellations in the plane. *The Computer Journal*, 21:168–173, 1978.
- [GS07] J. L. Gavartin and A. L. Shluger. Modeling HfO₂/SiO₂/Si interface. *Microelectron. Eng.*, 84(9-10):2412–2415, 2007.

- [GSPS07] L. Giordano, P. V. Sushko, G. Pacchioni, and A. L. Shluger. Optical and EPR properties of point defects at a crystalline silica surface: Ab initio embedded-cluster calculations. *Phys. Rev. B*, 75(2):024109, 2007.
- [Gum64] H. K. Gummel. A self-consistent iterative scheme for one-dimensional steady state transistor calculations. *IEEE Trans. on Electron Devices*, 11:455–465, 1964.
- [HE88] R. W. Hockney and J. W. Eastwood. *Computer Simulation Using Particles*. IOP Publishing, 1988.
- [Hes91] K. Hess, editor. *Monte Carlo Device Simulation: Full Band and Beyond*. Kluwer Academic Publishers, 1991.
- [HK05] N. Hitschfeld-Kahler. Generation of 3D mixed element meshes using a flexible refinement approach. *Eng. Comput.*, 21(2):101–114, 2005.
- [HKKT89] D. Hisamoto, T. Kaga, Y. Kawamoto, and E. Takeda. A fully depleted lean-channel transistor (DELTA)-a novel vertical ultra thin SOI MOSFET. In *1989 International Electron Devices Meeting Technical Digest.*, pages 833–836, 1989.
- [HLK⁺00] D. Hisamoto, Wen-Chin Lee, J. Kedzierski, H. Takeuchi, K. Asano, C. Kuo, E. Anderson, Tsu-Jae King, J. Bokor, and Chenming Hu. FinFET-a self-aligned double-gate MOSFET scalable to 20 nm. *IEEE Trans. Electron Devices*, 47(12):2320–2325, 2000.
- [HVKR03] N. Hitschfeld, L. Villablanca, J. Krause, and M. C. Rivara. Improving the quality of meshes for the simulation of semiconductor devices using Lepp-based algorithms. *Int. J. Numer. Methods Eng.*, 58:333–347, 2003.
- [IKW02] M. Ida, K. Kurishima, and N. Watanabe. Over 300 GHz f_T and f_{max} InP/InGaAs double heterojunction bipolar transistors with a thin pseudomorphic base. *IEEE Electron Device Lett.*, 23:694–696, 2002.

- [IN07] S. S. Iyer and E. J. Nowak. 45 nm SOI and beyond - getting to a general purpose technology. In *Proceedings of the 2007 IEEE International SOI Conference*, pages 1–4, 2007.
- [IWE⁺02] M. Jeong, H.-S. P. Wong, E. Nowak, J. Kedzierski, and E. C. Jones. High performance double-gate device technology challenges and opportunities. In *Proceedings fo the International Symposium on Quality Electronic Design*, pages 492–495, 2002.
- [JBSM⁺08] M.-A. Jaud, S. Barraud, J. Saint-Martin, A. Bournel, P. Dollfus, and H. Jaouen. A Pearson effective potential for Monte Carlo simulation of quantum confinement effects in nMOSFETs. *IEEE Trans. Electron Devices*, 55(12):3450–3458, 2008.
- [JFT07a] S. Jin, M.V. Fischetti, and T.-W. Tang. Modeling of electron mobility in gated silicon nanowires at room temperature: Surface roughness scattering, dielectric screening, and band nonparabolicity. *J. Appl. Phys.*, 102(8):083715, 2007.
- [JFT07b] S. Jin, M.V. Fischetti, and T.-W. Tang. Modeling of surface-roughness scattering in ultrathin-body SOI MOSFETs. *IEEE Trans. Electron Devices*, 54(9):2191–2203, 2007.
- [JL89] C. Jacoboni and P. Lugli. *The Monte Carlo Method for Semiconductor Device Simulation*. Springer-Verlag, Wien-New York, 1989.
- [JLFS91] G. U. Jensen, B. Lund, T. A. Fjeldly, and M. Shur. Monte Carlo simulation of semiconductor devices. *Comput. Phys. Commun.*, 67:1–61, 1991.
- [JPM04] S. Jin, Y. J. Park, and H. S. Min. Simulation of quantum effects in the nano-scale semiconductor device. *Journal of Semiconductor Technology and Science*, 4(1):32–40, March 2004.
- [JR83a] C. Jacoboni and L. Reggiani. The Monte Carlo method for the solution of charge transport in semiconductors with applications to covalent materials. *Rev. Mod. Phys.*, 55:645–705, 1983.
- [JR83b] C. Jacoboni and L. Reggiani. The Monte Carlo method for the solution of charge transport in semiconductors with applications to covalent materials. *Rev. Mod. Phys.*, 55:645–705, 1983.

- [KA01] K. Kalna and A. Asenov. Scaling of pHEMTs to decanano dimensions. *VLSI Design*, 13:435–439, 2001.
- [KA04] K. Kalna and A. Asenov. Role of multiple delta doping in PHEMTs scaled to sub-100 nm dimensions. *Solid-State Electron.*, 48:1223–1232, 2004.
- [KAAM⁺08] K. Kalna, A. Asenov, J. S. Ayubi-Moak, A. J. Craven, R. Droopad, R. Hill, M. C. Holland, X. Li, A. R. Long, P. Longo, D. MacIntyre, M. Passlack, G. Paterson, C. R. Stanley, S. Thoms, H. Zhou, and I. G. Thayne. III-V MOSFETs for digital applications with silicon co-integration. In *Proceedings of the 2008 International Conference on Advanced Semiconductor Devices and Microsystems*, pages 39–46, 2008.
- [KDD⁺06] J. Kavalieros, B. Doyle, S. Datta, G. Dewey, M. Doczy, B. Jin, D. Lionberger, M. Metz, W. Rachmady, M. Radosavljevic, U. Shah, N. Zelik, and R. Chau. Tri-gate transistor architecture with high-k gate dielectrics, metal gates, and strain engineering. In *2006 Symposium on VLSI Technology, Digest of Technical Papers*, pages 50–51, 2006.
- [KDPA07] K. Kalna, R. Droopad, M. Passlack, and A. Asenov. Monte Carlo simulations of InGaAs nano-MOSFETs. *Microelectron. Eng.*, 84(9-10):2150 – 2153, 2007.
- [Key01] R. W. Keyes. Fundamental limits of silicon technology. *Proc. IEEE*, 89:259–288, 2001.
- [KK97] G. Karypis and V. Kumar. METIS: A software package for partitioning unstructured graphs, partitioning meshes, and computing fill-reducing orderings of sparse matrices. Technical report, Univ. of Minnesota, 1997.
- [KL57] W. Kohn and J. M. Luttinger. Quantum theory of electrical transport phenomena. *Phys. Rev.*, 108(3):590–611, 1957.
- [Knu75] D. E. Knuth. *The Art of Computer Programming*. Addison-Wesley, 1975.

- [KS65] W. Kohn and L. J. Sham. Self-consistent equations including exchange and correlation effects. *Phys. Rev.*, 140(4A):A1133–A1138, 1965.
- [KSGL⁺08] K. Kalna, Natalia Seoane, A. J. Garcia-Loureiro, I. G. Thayne, and A. Asenov. Benchmarking of scaled InGaAs implant-free nanoMOSFETs. *IEEE Trans. Electron Devices*, 55(9):2297–2306, 2008.
- [KWR03] G.A. Kathawala, B. Winstead, and U. Ravaioli. Monte Carlo simulations of double-gate MOSFETs. *IEEE Trans. Electron Devices*, 50(12):2467–2473, 2003.
- [KYU⁺99] C. Kaneta, T. Yamasaki, T. Uchiyama, T. Uda, and K. Terakura. Structure and electronic property of Si(100)/SiO₂ interface. *Microelectron. Eng.*, 48(1-4):117–120, 1999.
- [Lau96] S. E. Laux. On particle-mesh coupling in Monte Carlo semiconductor device simulation. *IEEE Trans. Comput-Aided Des. Integr. Circuits Syst.*, 15:1266–1277, 1996.
- [Law77] C. L. Lawson. Software for C1 Surface Interpolation. In J. R. Rice, editor, *Mathematical Software III*, pages 161–194, New York, USA, 1977. Academic Press.
- [LL86] D. T. Lee and A. K. Lin. Generalized Delaunay triangulation for planar graphs. *Discrete & Computational Geometry*, 1:201–217, 1986.
- [Lo85] S. H. Lo. A new mesh generation scheme for arbitrary planar domains. *Int. J. Numer. Methods Eng.*, 21:1403–1426, 1985.
- [LP88] R. Lohner and P. Parikh. Three-dimensional grid generation by the advancing front method. *International Journal for Numerical Methods in Fluids*, 8:1135–1149, 1988.
- [LP94] M. Levy and J. P. Perdew. Density functionals for exchange and correlation energies: Exact conditions and comparison of approximations. *Int. J. Quantum Chem.*, 49(4):539–548, 1994.

- [LSF06] M. Luisier, A. Schenk, and W. Fichtner. Three-dimensional full-band simulation of Si nanowire transistors. In *2006 International Electron Devices Meeting Technical Digest.*, pages 811–814, 2006.
- [LSF07] M. Luisier, A. Schenk, and W. Fichtner. Atomistic treatment of interface roughness in Si nanowire transistors with different channel orientations. *Appl. Phys. Lett.*, 90:102103, 2007.
- [MAA⁺07] K. Mistry, C. Allen, C. Auth, B. Beattie, D. Bergstrom, M. Bost, M. Brazier, M. Buehler, A. Cappellani, R. Chau, C.-H. Choi, G. Ding, K. Fischer, T. Ghani, R. Grover, W. Han, D. Hanken, M. Hattendorf, J. He, J. Hicks, R. Huessner, D. Ingerly, P. Jain, R. James, L. Jong, S. Joshi, C. Kenyon, K. Kuhn, K. Lee, H. Liu, J. Maiz, B. McIntyre, P. Moon, J. Neiryneck, S. Pae, C. Parker, D. Parsons, C. Prasad, L. Pipes, M. Prince, P. Ranade, T. Reynolds, J. Sandford, L. Shifren, J. Sebastian, J. Seiple, D. Simon, S. Sivakumar, P. Smith, C. Thomas, T. Troeger, P. Vandervoorn, S. Williams, and K. Zawadzki. A 45nm logic technology with high-k+metal gate transistors, strained silicon, 9 Cu interconnect layers, 193nm dry patterning, and 100% Pb-free packaging. In *2007 International Electron Devices Meeting Technical Digest.*, pages 247–250, 2007.
- [Mar86] P. A. Markowich. *The Stationary Semiconductor Device Equations*. Springer-Verlag, Wien-New York, 1986.
- [med] Taurus Medici: A 2D device simulator. <http://www.synopsys.com/Tools/TCAD/DeviceSimulation/Pages/TaurusMedici.aspx>.
- [MHM92] S. Miyano, M. Hirose, and F. Masuoka. Numerical analysis of a cylindrical thin-pillar transistor (CYNTHIA). *IEEE Trans. Electron Devices*, 39(8):1876–1881, 1992.
- [Moo65] G. E. Moore. Cramming more components onto integrated circuits. *Electronics*, 38:114–117, 1965.
- [MSA⁺05] A. Martinez, A. Svizhenko, M. P. Anantram, J. R. Barker, A. R. Brown, and A. Asenov. A study of the interface roughness on a

- DG-MOSFET using a full 2D NEGF technique. In *2005 International Electron Devices Meeting Technical Digest.*, pages 627–630, 2005.
- [MSR⁺08] S. Markov, P. V. Sushko, S. Roy, C. Fiegna, E. Sangiorgi, A. L. Shluger, and A. Asenov. Si-SiO₂ interface band-gap transition - effects on MOS inversion layer. *Phys. Status Solidi A-Appl. Mat.*, 205(6):1290–1295, 2008.
- [MSSS04] S. Mukhopadhyay, P. V. Sushko, A. M. Stoneham, and A. L. Shluger. Modeling of the structure and properties of oxygen vacancies in amorphous silica. *Phys. Rev. B*, 70(19):195203, 2004.
- [MU49] N. Metropolis and S. M. Ulam. The Monte Carlo method. *J. Amer. Stat. Assoc.*, 44:335–341, 1949.
- [MV00] S. A. Mitchell and S. A. Vavasis. Quality mesh generation in higher dimensions. *SIAM Journal on Computing*, 29(2):1334–1370, 2000.
- [NCA⁺06] K. Nehari, N. Cavassilas, J.L. Autran, M. Bescond, D. Munteanu, and M. Lannoo. Influence of band structure on electron ballistic transport in silicon nanowire MOSFET's: An atomistic study. *Solid-State Electronics*, 50(4):716 – 721, 2006.
- [NTO⁺91] A. Nitayama, H. Takato, N. Okabe, K. Sunouchi, K. Hieda, F. Horiguchi, and F. Masuoka. Multi-pillar surrounding gate transistor (M-SGT) for compact and high-speed circuits. *IEEE Trans. Electron Devices*, 38(3):579–583, 1991.
- [Pas05] M. Passlack. Development methodology for high- κ gate dielectrics on III-V semiconductors: Gd_xGa_{0,4-x}O_{0,6}/Ga₂O₃ dielectric stacks on GaAs. *J. Vacuum Sci. Technol. B*, 23(4):1773–1781, 2005.
- [PBE96] J. P. Perdew, K. Burke, and M. Ernzerhof. Generalized gradient approximation made simple. *Phys. Rev. Lett.*, 77(18):3865–3868, 1996.
- [PC02] J. T. Park and J. P. Colinge. Multiple-gate SOI MOSFETs: Device desing guidelines. *IEEE Trans. Electron Devices*, 49:2222–2229, 2002.

- [PCD01] Jong-Tae Park, J.-P. Colinge, and C.H. Diaz. Pi-gate SOI MOSFET. *IEEE Electron Device Lett.*, 22(8):405–406, 2001.
- [PDG04] E. Pop, R. W. Dutton, and K. E. Goodson. Analytic band Monte Carlo model for electron transport in Si including acoustic and optical phonon dispersion. *J. Appl. Phys.*, 96(9):4998–5005, 2004.
- [PHC96] A. Pasquarello, M. S. Hybertsen, and R. Car. Theory of Si 2p core-level shifts at the Si(001)-SiO₂ interface. *Phys. Rev. B*, 53(16):10942–10950, Apr 1996.
- [PHC98] A. Pasquarello, M. S. Hybertsen, and R. Car. Interface structure between silicon and its oxide by first-principles molecular dynamics. *Nature*, 396(6706):58–60, 1998.
- [PKK⁺06] A. Pethe, T. Krishnamohan, Donghyun Kim, Saeroonter Oh, H-S.P. Wong, and K. Saraswat. Investigation of the performance limits of III-V double-gate n-MOSFETs. In *2006 16th Biennial University/Government/Industry Microelectronics Symposium*, pages 47–50, 2006.
- [PVMZ87] J. Peraire, M. Vahdati, K. Morgan, and O. C. Zienkiewicz. Adaptive remeshing for compressible flow computations. *Journal of Computational Physics*, 72:449–466, 1987.
- [RA08] C. Riddet and A. Asenov. Convergence properties of density gradient quantum corrections in 3D ensemble Monte Carlo simulations. In *2008 International Conference on Simulation of Semiconductor Processes and Devices*, pages 261–264, 2008.
- [RAF04] S. Richard, F. Aniel, and G. Fishman. Energy-band structure of Ge, Si, and GaAs: A thirty-band $k \cdot p$ method. *Phys. Rev. B*, 70(23):235204, 2004.
- [Rav98] U. Ravaioli. Hierarchy of simulation approaches for hot carrier transport in deep submicron devices. *Semicond. Sci. Technol.*, 13:1–10, 1998.
- [RBA⁺07] C. Riddet, A. R. Brown, C. Alexander, J. R. Watling, S. Roy, and A. Asenov. 3-D Monte Carlo simulation of the impact of quantum confinement scattering on the magnitude of current fluctuations

- in double gate MOSFETs. *IEEE Trans. Nanotechnol.*, 6:48–55, 2007.
- [RBAL⁺06] G. Roy, A. R. Brown, F. Adamu-Lema, S. Roy, and A. Asenov. Simulation study of individual and combined sources of intrinsic parameter fluctuations in conventional nano-MOSFETs. *IEEE Trans. Electron Devices*, 53(12):3063–3070, 2006.
- [Ree68] H. D. Rees. Calculation of steady state distribution functions by exploiting stability. *Phys. Lett. A*, 26:416–417, 1968.
- [Ree69] H. D. Rees. Calculation of distribution functions by exploiting the stability of the steady state. *J. Phys. Chem. Solids*, 30:643–655, 1969.
- [Rei65] F. Reif. *Fundamentals of Statistical and Thermal Physics*. McGraw-Hill, 1965.
- [RF02] S. M. Ramey and D. K. Ferry. 3D Monte Carlo modeling of thin SOI MOSFETs including the effective potential and random dopant distribution. *J. Comp. Electron.*, 1:267–271, 2002.
- [RF03] S. M. Ramey and D. K. Ferry. Implementation of surface roughness scattering in Monte Carlo modeling of thin SOI MOSFETs using the effective potential. *IEEE Trans. Nanotechnol.*, 2(2):110–114, 2003.
- [RHLG94] U. A. Ranawake, C. Huster, P. M. Lenders, and S. M. Goodnick. PMC-3D: A parallel three-dimensional Monte Carlo semiconductor device simulator. *IEEE Trans. Comput-Aided Des. Integr. Circuits Syst.*, 13:712–724, 1994.
- [Rid77] B. K. Ridley. Reconciliation of the Conwell-Weisskopf and Brooks-Herring formulae for charged-impurity scattering in semiconductors: Third-body interference. *J. Phys. C: Solid State Phys.*, 10(10):1589–1593, 1977.
- [Riv84] M.-C. Rivara. Mesh refinement processes based on the generalized bisection of simplices. *SIAM J. Numer. Anal.*, 21(3):604–613, 1984.

- [RKH⁺01] K. Rim, S. Koester, M. Hargrove, J. Chu, P.M. Mooney, J. Ott, T. Kanarsky, P. Ronsheim, M. Jeong, A. Grill, and H.-S.P. Wong. Strained Si NMOSFETs for high performance CMOS technology. In *2001 Symposium on VLSI Technology, Digest of Technical Papers*, pages 59–60, 2001.
- [roa] The International Technology Roadmap for Semiconductors. <http://www.itrs.net>.
- [RPD85] C. S. Rafferty, M. R. Pinto, and R. W. Dutton. Iterative methods in semiconductor device simulation. *IEEE Trans. on Computer-Aided Design*, 4(4):462–471, 1985.
- [RS88] C. Ringhofer and C. Schmeiser. A modified Gummel method for the basic semiconductor device equations. *IEEE Trans. on Computer-Aided Design*, 7(2):251–253, 1988.
- [Saa96] Y. Saad. *Iterative Methods for Sparse Linear Systems*. PWS Publishing Co., 1996.
- [SCB⁺04] Huiling Shang, J.O. Chu, S. Bedell, E.P. Gusev, P. Jamison, Ying Zhang, J.A. Ott, M. Copel, D. Sadana, K.W. Guarini, and Meikei Jeong. Selectively formed high mobility strained Ge PMOSFETs for high performance CMOS. In *2004 International Electron Devices Meeting Technical Digest.*, pages 157–160, 2004.
- [SCB⁺08] A.A. Salman, Shuqing Cao, S.G. Beebe, M.M. Pelella, and R.W. Dutton. Double-well field effect diode vs. SCR behavior under CDM stress in 45nm SOI technology. In *Proceedings of the 2008 IEEE International SOI Conference*, pages 143–144, 2008.
- [Sch68] L. I. Schiff. *Quantum Mechanics*. McGraw-Hill, 3rd edition, 1968.
- [Sel84] S. Selberherr. *Analysis and Simulation of Semiconductor Devices*. Springer-Verlag, Wien-New York, 1984.
- [sen] Sentaurus Device: An advanced multidimensional (1D/2D/3D) device simulator. <http://www.synopsys.com/Tools/TCAD/DeviceSimulation/Pages/SentaurusDevice.aspx>.

- [SG69] D. L. Scharfetter and H. K. Gummel. Large-signal analysis of a silicon Read diode oscillator. *IEEE Trans. Electron Devices*, 16:64–77, 1969.
- [SG91] M. S. Shephard and M. K. Georges. Three-Dimensional Mesh Generation by Finite Octree Technique. *Int. J. Numer. Methods Eng.*, 32:709–749, 1991.
- [SGL06] N. Seoane and A. J. García-Loureiro. Study of parallel numerical methods for semiconductor device simulation. *Int. J. Numer. Model.-Electron. Netw. Device Fields*, 19:15–32, 2006.
- [SGLA08] N. Seoane, A. García-Loureiro, and M. Aldegunde. Optimization of linear systems for 3D parallel simulation of semiconductor devices: Application to statistical studies. *Int. J. Numer. Model.-Electron. Netw. Device Fields*, 2008.
- [SGZ⁺03] P.M. Solomon, K.W. Guarini, Y. Zhang, K. Chan, E.C. Jones, G.M. Cohen, A. Krasnoperova, M. Ronay, O. Dokumaci, H.J. Hovel, J.J. Bucchignano, Jr. Cabral, C., C. Lavoie, V. Ku, D.C. Boyd, K. Petrarca, J.H. Yoon, I.V. Babich, J. Treichler, P.M. Kozlowski, J.S. Newbury, C.P. D’Emic, R.M. Sicina, J. Benedict, and H.-S.P. Wong. Two gates are better than one [double-gate MOSFET process]. *IEEE Circuits and Devices Magazine*, 19(1):48–62, 2003.
- [SH84] T. Sekigawa and Y. Hayashi. Calculated threshold-voltage characteristics of an XMOS transistor having an additional bottom gate. *Solid-State Electron.*, 27(8-9):827–828, 1984.
- [She88] M. S. Shephard. Approaches to the automatic generation and control of finite element meshes. *Applied Mechanics Reviews*, 41:169–185, 1988.
- [SKBA08] D. Shiri, Yifan Kong, A. Buin, and M.P. Anantram. Investigation of strain effects on the band-structure of Si nanowires using TB and DFT methods. In *8th IEEE Conference on Nanotechnology*, pages 343–345, 2008.
- [SKK⁺07] Y. Sun, S. J. Koester, E. W. Kiewra, J. P. de Souza, N. Ruiz, J. J. Bucchignano, A. Callegari, K. E. Fogel, D. K. Sadana, J. Fompeyrine, D. J. Webb, J.-P. Locquet, M. Sousa, and R. Germann.

- Post-Si CMOS: III-V n-MOSFETs with high-k gate dielectrics. In *Proceedings of the 2007 Compound Semiconductor Integrated Circuit Symposium*, pages 231–234, 2007.
- [SLK97] Y. Saad, G.-C. Lo, and S. Kuznetsov. PPARSLIB users manual: A portable library of parallel sparse iterative solvers. Technical report, Univ. of Minnesota, 1997.
- [SLK⁺04] H. Shang, K.-L. Lee, P. Kozłowski, C. D’Emic, I. Babich, E. Sikorski, M. Jeong, H.-S.P. Wong, K. Guarini, and W. Haensch. Self-aligned n-channel germanium MOSFETs with a thin Ge oxynitride gate dielectric and tungsten gate. *IEEE Electron Device Lett.*, 25(3):135–137, 2004.
- [SMGGR06] C. Sampedro-Matarin, F. Gamiz, A. Godoy, and F.J.G. Ruiz. The multivalley effective conduction band-edge method for Monte Carlo simulation of nanoscale structures. *IEEE Trans. Electron Devices*, 53(11):2703–2710, 2006.
- [SRG⁺08] R. Shaltaf, G.-M. Rignanese, X. Gonze, F. Giustino, and A. Pasquarello. Band offsets at the Si/SiO₂ interface from many-body perturbation theory. *Phys. Rev. Lett.*, 100(18):186401, 2008.
- [SS83] T. Sota and K. Suzuki. Phonon attenuation in heavily doped many-valley semiconductors. II. the effect of mass anisotropy. *J. Phys. C: Solid State Physics*, 16(22):4347–4364, 1983.
- [Ste77] F. Stern. Effect of a thin transition layer at a Si-SiO₂ interface on electron mobility and energy levels. *Solid State Communications*, 21(2):163–166, 1977.
- [SVSP96] P. R. Suresh, P. Venugopal, S. T. Selvam, and S. Potla. Combined Effect of Grain Boundary Depletion and PolySi/Oxide Interface Depletion on Drain Characteristics of a p-MOSFET. In *Proceedings of the Ninth International Conference on VLSI Design*, pages 156–161, 1996.
- [TAA⁺04] S.E. Thompson, M. Armstrong, C. Auth, S. Cea, R. Chau, G. Glass, T. Hoffman, J. Klaus, Zhiyong Ma, B. McIntyre, A. Murthy, B. Obradovic, L. Shifren, S. Sivakumar, S. Tyagi,

- T. Ghani, K. Mistry, M. Bohr, and Y. El-Mansy. A logic nanotechnology featuring strained-silicon. *IEEE Electron Device Lett.*, 25(4):191–193, 2004.
- [TBCF97] Y. Taur, D. Buchanan, W. Chen, and D. Frank. CMOS scaling into nanometer regime. *Proc. IEEE*, 85:486–503, 1997.
- [TMI+04] B. H. V. Topping, J. Muylle, P. Iványi, R. Putanowicz, and B. Cheng. *Finite Element Mesh Generation*. Saxe-Coburg Publications, 2004.
- [TMSS97] H. P. Tuinhout, A. H. Montree, J. Schmitz, and P. A. Stolk. Effects of gate depletion and boron penetration on matching of deep submicron CMOS transistors. In *1997 International Electron Devices Meeting Technical Digest.*, pages 631–634, 1997.
- [Tom93] K. Tomizawa. *Numerical Simulation of Submicron Semiconductor Devices*. Artech House, 1993.
- [Vog07] E. M. Vogel. Technology and metrology of new electronic materials and devices. *Nat. Nanotechnol.*, 2:25–32, 2007.
- [Vor08] G. Voronoï. Nouvelles applications des paramètres continus à la théorie des formes quadratiques. recherches sur les paralléloèdres primitifs. *Journal Reine angew. Math.*, 134, 1908.
- [Wat81] D. F. Watson. Computing the n-dimensional Delaunay tessellation with application to Voronoi polytopes. *The Computer Journal*, 24:167–172, 1981.
- [Wat90] D. F. Watson. The integrity of geometrical boundaries in the 2-dimensional Delaunay triangulation. *Communication in Numerical Methods in Engineering*, 6:101–109, 1990.
- [WCH+06] W. Wessner, J. Cervenka, C. Heitzinger, A. Hossinger, and S. Selberherr. Anisotropic mesh refinement for the simulation of three-dimensional semiconductor manufacturing processes. *IEEE Trans. Comput-Aided Des. Integr. Circuits Syst.*, 25(10):2129–2139, 2006.
- [WFS98] H. S. Wong, D. Frank, and P. Solomon. Device design considerations for double-gate, ground-plane and single gated ultra-thin

- SOI MOSFETs at the 25nm channel length generation. In *1998 International Electron Devices Meeting Technical Digest.*, pages 407–410, 1998.
- [WFS⁺99] H.-S.P. Wong, D.J. Frank, P.M. Solomon, C.H.J. Wann, and J.J. Welser. Nanoscale CMOS. *Proc. IEEE*, 87(4):537–570, 1999.
- [WH94] N. P. Weatherill and O. Hassan. Efficient three-dimensional Delaunay triangulation with automatic point creation and imposed boundary constraints. *Int. J. Numer. Methods Eng.*, 37:2005–2039, 1994.
- [WLWD89] Ke-Chih Wu, R. F. Lucas, Z. Y. Wang, and R. W. Dutton. New approaches in a 3-D one-carrier device solver. *IEEE Trans. on Computer-Aided Design*, 8(5):528–537, 1989.
- [WMH00] C. S. Whelan, P. F. Marsh, and W. E. Hoke. GaAs metamorphic HEMT (MHEMT): an attractive alternative to InP HEMTs for high performance low noise and power applications. In *Proceedings of the 2000 Conference on Indium Phosphide and Related Materials*, pages 337–340, 2000.
- [WML⁺03] R. J. Welty, K. Mochizuki, C. R. Lutz, R. E. Welser, and P. M. Asbeck. Design and performance of tunnel collector HBTs for microwave power amplifiers. *IEEE Trans. Electron Devices*, 50:894–900, 2003.
- [WMM06] H. Watanabe, D. Matsushita, and K. Muraoka. Determination of tunnel mass and physical thickness of gate oxide including poly-Si/SiO₂ and Si/SiO₂ interfacial transition layers. *IEEE Trans. Electron Devices*, 53(6):1323–1330, 2006.
- [Won02] H. S. P. Wong. Beyond the conventional transistor. *IBM J. Res. & Dev.*, 46:133–165, 2002.
- [wTW04] Ting wei Tang and Bo Wu. Quantum correction for the Monte Carlo simulation via the effective conduction-band edge equation. *Semicond. Sci. Technol.*, 19(1):54–60, 2004.
- [Yan94] D. Yang. *Mesh generation and information model for device simulation*. PhD thesis, Stanford University, 1994.

- [YCA⁺02] Bin Yu, Leland Chang, S. Ahmed, Haihong Wang, S. Bell, Chih-Yuh Yang, C. Tabery, Chau Ho, Qi Xiang, Tsu-Jae King, J. Bokor, Chenming Hu, Ming-Ren Lin, and D. Kyser. FinFET scaling to 10 nm gate length. In *2002 International Electron Devices Meeting Technical Digest.*, pages 251–254, 2002.
- [YCC⁺02] Fu-Liang Yang, Hao-Yu Chen, Fang-Cheng Chen, Cheng-Chuan Huang, Chang-Yun Chang, Hsien-Kuang Chiu, Chi-Chuang Lee, Chi-Chun Chen, Huan-Tsung Huang, Chih-Jian Chen, Hun-Jan Tao, Yee-Chia Yeo, Mong-Song Liang, and Chenming Hu. 25 nm CMOS omega FETs. In *2002 International Electron Devices Meeting Technical Digest.*, pages 255–258, 2002.
- [YES02] Y. Yamashita, A. Endoh, and K. Shinohara. Pseudomorphic $In_{0.52}Al_{0.48}As/In_{0.7}Ga_{0.3}As$ HEMTs with an ultrahigh f_t of 562 GHz. *IEEE Electron Device Lett.*, 23:573–575, 2002.
- [YHH99] G.-Y. Yang, S.-H. Hur, and C.-H. Han. A physical-based analytical turn-on model of polysilicon thin-film transistors for circuit simulation. *IEEE Trans. Electron Devices*, 46(1):165–172, 1999.
- [YKU⁺01] T. Yamasaki, C. Kaneta, T. Uchiyama, T. Uda, and K. Terakura. Geometric and electronic structures of $SiO_2/Si(001)$ interfaces. *Phys. Rev. B*, 63(11):115314, 2001.
- [YLK⁺04] Eun-Jung Yoon, Sung-Young Lee, Sung-Min Kim, Min-Sang Kim, Sung Hwan Kim, Li Ming, Sungdae Suk, Kyoung-hawn Yeo, Chang Woo Oh, Jung dong Choe, Donguk Choi, Dong-Won Kim, Donggun Park, Kinam Kim, and Byung-Il Ryu. Sub 30 nm multi-bridge-channel MOSFET (MBCFET) with metal gate electrode for ultra high performance application. In *2004 International Electron Devices Meeting Technical Digest.*, pages 627–630, 2004.
- [YS83] M. A. Yerry and M. S. Shephard. A modified quadtree approach to finite element mesh generation. *IEEE Journal*, pages 39–46, 1983.
- [YS84] M. A. Yerry and M. S. Shephard. Three-dimensional mesh generation by modified octree technique. *Int. J. Numer. Methods Eng.*, 20:1965–1990, 1984.

- [Yua99] J. S. Yuan. *SiGe, GaAs, and InP Heterojunction Bipolar Transistors*. John Wiley & Sons, Inc., New York, NY, 1999.
- [YWK⁺03] P.D. Ye, G.D. Wilk, J. Kwo, B. Yang, H.-J.L. Gossmann, M. Frei, S.N.G. Chu, J.P. Mannaerts, M. Sergent, M. Hong, K.K. Ng, and J. Bude. GaAs MOSFET with oxide gate dielectric grown by atomic layer deposition. *IEEE Electron Device Lett.*, 24(4):209–211, 2003.
- [ZRL⁺05] Yun Zheng, C. Rivas, R. Lake, K. Alam, T.B. Boykin, and G. Klimeck. Electronic properties of silicon nanowires. *IEEE Trans. Electron Devices*, 52(6):1097–1103, 2005.
- [ZWK⁺00] Y. P. Zhao, J. R. Watling, S. Kaya, A. Asenov, and J. R. Barker. Drift diffusion and hydrodynamic simulations of Si/SiGe p-MOSFETs. *Mater. Sci. Eng. B-Solid State Mater. Adv. Technol.*, 72:180–183, 2000.