

**UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA**

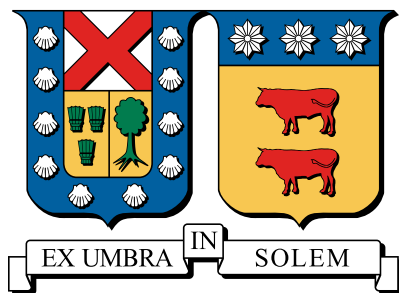
**DEPARTAMENTO DE INGENIERÍA MECÁNICA**

**VALPARAISO- CHILE**

**MODELACIÓN NUMÉRICA DE UNA PROTEÍNA  
INSERTA EN UNA MEMBRANA CELULAR  
USANDO EL MÉTODO DE ELEMENTOS DE  
BORDE.**

**Nicolás Espinoza Muñoz**

2019



**UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA**  
**DEPARTAMENTO DE INGENIERÍA MECÁNICA**  
**VALPARAISO- CHILE**

**MODELACIÓN NUMÉRICA DE UNA PROTEÍNA INSERTA EN UNA  
MEMBRANA CELULAR USANDO EL MÉTODO DE ELEMENTOS DE  
BORDE.**

**Nicolás Espinoza Muñoz**

Profesor Guía  
Christopher D. Cooper V., Ph.D.

Profesor Correferente  
Harvey A. Zambrano R., Ph.D.

---

# Resumen

Una proteína inmersa en una membrana que se modela de forma continua tiene al menos parte de su superficie encerrada entre lo que se puede considerar como dos interfaces, planas para efectos de modelación. Este trabajo busca modelar este tipo de moléculas para calcular su energía de solvatación, aplicando teoría de cargas imágenes mediante el método de elementos de borde (BEM). Se modifica una versión básica del programa PyGBe, que ya tiene la capacidad de resolver la ecuación de Laplace con BEM en sistemas proteína - solvente, para incluir el aporte de una membrana definida como el medio entre dos planos infinitos. Para esto se utiliza una función de Green que representa el efecto del solvente por fuera de la membrana de forma matemática mediante cargas imagen, evitando el problema de tener que discretizar los dos planos infinitos correspondientes. La función de Green que se utiliza no considera una distribución iónica en el solvente.

Los resultados obtenidos se contrastan con un software que se considera estándar actualmente - APBSmem. También comparamos el consumo de memoria y los tiempos de cálculo. Este trabajo es una prueba de concepto, y como tal la comparación con APBSmem es alentadora, pues el error porcentual es bajo y el tiempo de cómputo es normalmente inferior. El principal problema surge para mallas muy refinadas, o en moléculas con una gran superficie, pues el uso de memoria se dispara a causa del número de imágenes necesario para obtener una buena precisión. En síntesis, el método de cálculo es funcional, pero tiene falencias en cuanto a eficiencia que lo hacen poco atractivo como herramienta para casos más realistas.

**Palabras Clave** *Proteína, Membrana, Elementos de Borde, Función de Green, Carga Imagen, Energía de Solvatación*

---

# Abstract

A membrane-embedded protein that is modeled through continuum theory has at least part of its surface surrounded by what can be considered as two planar interfaces. This work presents a way of modelling these type of molecules to calculate their solvation energy by applying image charge theory with the Boundary Element Method (BEM). We modified a basic version of the `PyGBe` solver, which already has the capability to solve the Laplace equation via BEM for a protein - solvent system, to include the effect of a membrane defined as the medium between two infinite planes. To achieve this, we use a Green's function that takes into account the effect of the solvent outside the membrane in a mathematical fashion through the use of image charges, bypassing the problem of having to mesh the interfaces of the membrane. It is worth noting that said Green's function does not consider ion concentrations in the media.

The results are then compared with a standard in the field - `APBSmem`. A comparison in terms of memory consumption and calculation times is also done. This work, as a proof of concept, succeeds, because the difference in results with `APBSmem` is small and calculation times are usually shorter than those of the reference program. The main problem arises when using high density meshes, or when working with molecules with a big surface, since the memory consumption grows due to the number of image charges necessary to achieve a good level of precision. Therefore as a conclusion, the calculation method is functional, but its faults in terms of efficiency make it a less than attractive tool for realistic cases in its current state.

**Keywords** *Protein, Membrane, Solvation Energy, Image Charge, Boundary Element Method, Green's Function*

*“The greatest danger for most of us*

*is not that our aim is too high and we miss it,*

*but that it is too low and we reach it.”*

- Michelangelo

---

# Contenidos

<b>Resumen</b>	<b>iii</b>
<b>Abstract</b>	<b>iv</b>
<b>Índice de figuras</b>	<b>ix</b>
<b>Índice de tablas</b>	<b>xi</b>
<b>1 Introducción</b>	<b>1</b>
<b>2 Marco Teórico</b>	<b>5</b>
2.1 El campo eléctrico y el potencial eléctrico . . . . .	5
2.2 La permitividad de un medio material, la polarización y el campo eléctrico microscópico . . . . .	7
2.2.1 Campo eléctrico local y polarización . . . . .	8
2.2.2 Permitividad de un medio macroscópico . . . . .	12
2.3 Desplazamiento eléctrico y campo eléctrico en un medio dieléctrico . . . . .	13
2.4 Método de imágenes . . . . .	15
2.4.1 Campo de reacción en un sistema con una inter- faz y una carga puntual . . . . .	18
2.5 El Método de Elementos de Borde . . . . .	20
2.6 El contexto biológico de la membrana celular . . . . .	23
<b>3 Metodología, funcionamiento e implementación</b>	<b>27</b>

3.1	Representación del modelo de solvente implícito mediante BEM . . . . .	28
3.1.1	Solvent Excluded y Solvent Accesible Surface . . . . .	29
3.1.2	La ecuación de Poisson - Boltzmann (PB) y la Boundary Integral Equation (BIE) . . . . .	31
3.2	Descripción del software utilizado . . . . .	35
3.2.1	MSMS . . . . .	35
3.2.2	APBS . . . . .	36
3.2.3	APBSmem . . . . .	38
3.2.4	PyGBe . . . . .	39
3.3	Implementación del método de imágenes en el programa PyGBe . . . . .	44
3.3.1	Modelo de un sistema de tres medios dieléctricos con dos interfaces . . . . .	45
3.3.2	El método BEM y PyGBe con inclusión del efecto de la membrana . . . . .	48
3.3.3	El ciclo termodinámico en solvatación . . . . .	55
<b>4</b>	<b>Resultados</b>	<b>57</b>
4.1	Cálculo numérico de las energías . . . . .	57
4.1.1	Resultados mediante PyGBe . . . . .	58
4.1.2	Resultados mediante APBS - APBSmem . . . . .	60
4.2	Extrapolación de Richardson y convergencia . . . . .	63
4.3	Análisis de Resultados . . . . .	65
<b>5</b>	<b>Conclusiones</b>	<b>69</b>
5.1	Continuación del trabajo . . . . .	70
<b>A</b>	<b>Archivo Bromoethane.pqr</b>	<b>72</b>

<b>B</b>	<b>Energía de solvatación</b>	<b>73</b>
<b>C</b>	<b>Funcionamiento de APBSmem</b>	<b>74</b>
	<b>Bibliografía</b>	<b>79</b>



---

# Índice de figuras

1.1	Esquema de la situación física que se quiere modelar: una proteína en una membrana. . . . .	2
2.1	Efecto de un campo sobre la forma de un átomo. El electrón (en realidad la nube electrónica) es “halado” por el campo, mientras que el protón (o núcleo atómico) es “empujado”, induciendo así un dipolo. . . . .	8
2.2	Se observan las líneas de desplazamiento eléctrico, que dan cuenta del efecto de la polarización. . . . .	17
2.3	Sistema de una carga real y dos imaginarias para repre- sentar el campo de reacción, con el cambio de medio en $z = 0$ . Se consideran dos espacios semi-infinitos a partir de este plano. . . . .	18
2.4	Tramo de la bicapa con proteína inserta. Esta última se orienta de tal forma que la parte hidrofílica (polar) quede en contacto con el agua en el exterior, de igual forma que las cabezas de los fosfolípidos. . . . .	25
2.5	Esquema de un fosfolípido, con detalle sobre la com- posición de la cabeza hidrófila [13]. . . . .	26
3.1	Solvent Excluded y Solvent Accesible Surface, presen- tadas de forma esquemática. . . . .	30
3.2	Forma 2D de dividir la frontera del dominio de inte- gración para aplicar el método BEM a cada elemento de borde. . . . .	34

3.3	La imagen superior es la situación real para el caso de una carga, mientras que la figura inferior es el sistema trabajado con el método de imágenes. Esto sigue de manera infinita, pero solo se muestran unas pocas cargas por motivos obvios. . . . .	46
3.4	Aplicación del método semianalítico sobre un panel triangular. . . . .	53
3.5	Ciclo termodinámico estudiado según lo presentado en los párrafos anteriores. Lo que se calcula con PyGBe y APBS es la diferencia $1 \rightarrow 4$ . . . . .	56
4.1	Resultados de PyGBe con membrana para el caso <code>centered_ion</code> , según se varía el número de cargas imagen del método. La línea roja es el resultado extrapolado para 41 cargas imagen. . . . .	67
4.2	Variación porcentual en el resultado, en función del número de imágenes empleado. La línea roja es el resultado extrapolado para 41 cargas imagen. . . . .	68
4.3	Variación porcentual, mostrada en escala logarítmica, de los resultados en función del número de cargas imagen. .	68

---

# Índice de tablas

3.1	Expresión de las diferencias energéticas, en función de los estados 1 a 4. $\Delta G$ es la diferencia en la energía de Gibbs entre los distintos estados del ciclo 3.5. . . . .	56
4.1	Permitividades en los estados de interés. . . . .	58
4.2	Energías de solvatación calculadas, tiempos de cálculo y memoria RAM utilizada para los cálculos según la densidad de malla probada, para el archivo <code>centered_ion</code> . . . . .	59
4.3	Energías de solvatación calculadas según la densidad de malla para el archivo <code>3q</code> . . . . .	60
4.4	Energías de solvatación calculadas, tiempos de cálculo y memoria RAM para cada una de las tres mallas de la molécula de <code>bromoethane</code> . . . . .	60
4.5	Resultados, tiempos de cálculo y RAM utilizada para el cálculo sobre el archivo <code>centered_ion</code> , mediante APBS. . . . .	61
4.6	Resultados, tiempos de cálculo y RAM utilizada para las tres mallas en el caso de estudio <code>3q</code> , mediante APBS. . . . .	62
4.7	Resultado, tiempo de cálculo y RAM utilizada para la malla más fina en el caso de estudio de <code>bromoethane</code> , mediante APBS. . . . .	62
4.8	Valores teóricos extrapolados mediante extrapolación de Richardson de los distintos casos de estudio. . . . .	64

4.9	Convergencia al valor teórico extrapolado de los resultados, para PyGBE y las dos condiciones importantes de APBS. . . . .	64
4.10	Diferencia de PyGBE con inclusión de membrana, con las dos condiciones de contorno de APBS. . . . .	65

---

# Capítulo 1

## Introducción

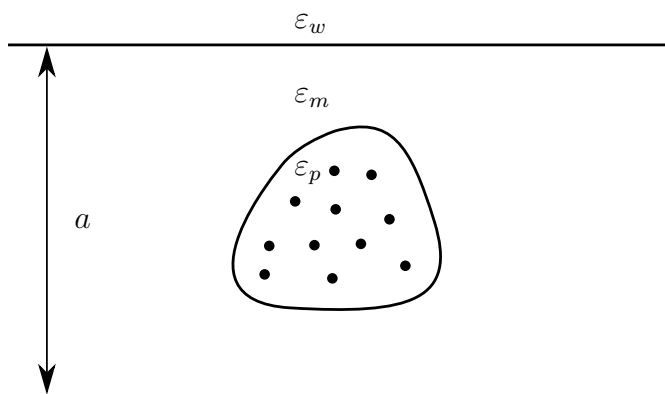
La complejidad de los fenómenos biológicos hace que sea imposible hasta el momento analizar de manera completa su naturaleza. Es por esto que actualmente se estudian de manera exhaustiva nuevas formas de representar los sistemas biológicos o sus elementos conformantes, recurriendo a múltiples disciplinas y enfoques. Por nombrar ejemplos de ramas de la ciencia relevantes se tienen la electrodinámica, la termodinámica, la mecánica de sólidos, y mecánica estadística, aunque de ninguna manera son las únicas. Además, al igual que en muchas otras áreas, la computación se tomó gran parte del escenario, volviéndose quizás la herramienta más útil y usada en las investigaciones y avances en la comprensión de la biología celular.

Para modelar la fenomenología de los sistemas biológicos según las diferentes disciplinas científicas mencionadas se utiliza una cantidad de métodos computacionales, de los cuales quizás los más conocidos son la dinámica molecular (MD) en casos explícitos y el método de elementos finitos (FEM) en casos implícitos. Por supuesto y como todo en la vida, ambos tienen sus ventajas y desventajas; un punto en contra que tienen en común ambos tipos de modelación es que en general son lentos (MD más que FEM). Por esto es que ha empezado a tomar cada vez más fuerza el método de elementos de borde/frontera (BEM), que permite resolver problemas físicos comparativamente rápido a expensas

de una formulación un tanto más compleja.

Por tanto y utilizando este método, el objetivo global de este trabajo consiste en obtener las ecuaciones de Green que permiten representar cargas en un medio acotado por dos interfaces paralelas, implementarlo en un programa de cálculo energético, y contrastar los resultados con aquellos de códigos o programas ya validados. Sumado a esto, la idea es que dicha implementación sea computacionalmente barata, conserve un buen nivel de precisión, y que la velocidad de cálculo sea competitiva frente a un programa de referencia en casos de estudio seleccionados.

La física de lo que buscamos representar consiste en una molécula totalmente embebida en una membrana de cierto espesor  $a$ , cuyos límites son superficies planas infinitas. Este sistema se modela a través de cargas imágenes mediante el método BEM, y esquemáticamente se muestra en la figura 1.1.



**Figura 1.1:** Esquema de la situación física que se quiere modelar: una proteína en una membrana.

Hay que tener presente que existen otros fenómenos además de los electrostáticos que no se consideran por la gran complejidad que conllevan; por ejemplo, la inversión de carga de la membrana, la onda mecánica

consecuente (o causal), etc, ya que en algunos casos ni siquiera se entiende bien su naturaleza; este trabajo se remite a electrostática de proteínas totalmente embebidas en la membrana.

El presente texto busca entonces *presentar la física que hay tras una interacción electrostática en medios dieléctricos, trabajando un método que permita modelar una membrana celular a partir de un sistema de tres medios separados por dos interfaces, con cargas distribuidas en el medio encerrado. Teniendo este modelo, se lo implementa en un código y se procede a la comparación con programas ya reconocidos.* En primer lugar se presenta una introducción a la teoría que permite trabajar los sistemas dieléctricos con interfaz de forma macroscópica y mediante el método de imágenes. También se echa un vistazo a la formulación que da pie al método BEM, la denominada Boundary Integral Equation de un sistema de ecuaciones diferencial, además de contextualizar sobre aspectos básicos de la membrana para entender qué es lo que se busca. A continuación se presenta un ejemplo de formulación integral de frontera con el caso de la ecuación de Poisson - Boltzmann, que no se trabaja directamente en esta investigación, pero que va de la mano con la metodología de solvente implícito; además, en el caso de no haber una distribución de iones ( $\kappa = 0$ ), se recupera una ecuación de Laplace, la que si se utiliza en este trabajo. Junto a esto, se describen los programas relevantes para lograr los objetivos planteados. Entrando a la parte fundamental de esta investigación, se desarrolla el método de imágenes para un sistema con dos interfaces paralelas, y cómo se incluye en la formulación, además de su implementación al programa PyGBe. Por último, se presentan los resultados obtenidos y cómo se comparan con los de programas como APBS – APBSmem;

esto considera también la forma en que influye la variación del número de cargas del método de imágenes en el resultado. Tras tener estos resultados se termina presentando las conclusiones extraídas de este trabajo.



---

## Capítulo 2

### Marco Teórico

#### 2.1 El campo eléctrico y el potencial eléctrico

El campo eléctrico correspondiente a una carga puntual  $q$  que se encuentra en el vacío en la posición  $\mathbf{r}_0$  viene dado por la conocida ecuación

$$\vec{E}(\mathbf{r}) = \frac{q}{4\pi\epsilon_0} \frac{(\mathbf{r} - \mathbf{r}_0)}{|\mathbf{r} - \mathbf{r}_0|^3} \quad (2.1.1)$$

siendo  $\epsilon_0$  la permitividad macroscópica del vacío, una medida de cómo fluye el campo eléctrico en el vacío. Por supuesto, en el caso ejemplificado en la ecuación (2.1.1), no hay un medio que reaccione, por lo que  $\epsilon_0$  se toma como referencia. De ahora en adelante se trabaja con  $\epsilon$  propio de un dieléctrico; además se elabora más sobre la permitividad en la sección siguiente.

Cuando se cuenta con  $N$  cargas puntuales, la ecuación (2.1.1) se convierte en una sumatoria sobre todos los elementos del sistema de cargas a considerar

$$\vec{E}(\mathbf{r}) = \sum_{j=1}^{N_q} \frac{q_j}{4\pi\epsilon} \frac{(\mathbf{r} - \mathbf{r}_j)}{|\mathbf{r} - \mathbf{r}_j|^3} \quad (2.1.2)$$

Para un caso con una distribución continua de carga, se define el flujo de campo eléctrico a través de una superficie que encierra toda la dis-

tribución de carga como

$$\int_{\partial\Omega} \vec{E}(\mathbf{r}) \cdot \hat{\mathbf{n}} d\Omega = \int_{\Omega} \frac{\rho}{\varepsilon} d\Omega \quad (2.1.3)$$

Esta es la ley de Gauss en forma integral. Con el teorema de la divergencia aplicado a la ecuación (2.1.3) se llega a la ley de Gauss en forma diferencial:

$$\int_{\Omega} \nabla \cdot \vec{E} d\Omega = \int_{\Omega} \frac{\rho}{\varepsilon} d\Omega \implies \nabla \cdot \vec{E} = \frac{\rho}{\varepsilon} \quad (2.1.4)$$

Por otro lado, como el campo eléctrico es un campo vectorial irrotacional, se puede aplicar el teorema fundamental del cálculo para obtener su función potencial. Como  $\vec{E}(\mathbf{r})$  es una variable solo de la diferencia de posiciones entre la de la carga y el punto de evaluación  $|\mathbf{r} - \mathbf{r}_0|$  (función solo del radio), la integración es directa si la referencia para el potencial eléctrico se toma como 0 al infinito, quedando

$$\begin{aligned} \vec{E} = \nabla\phi \quad \rightarrow \quad \phi(\mathbf{r}) &= - \int_{\infty}^{\mathbf{r}} \vec{E}(\mathbf{r}^*) d\mathbf{r}^* \\ &= - \int_{\infty}^{\mathbf{r}} \frac{q}{4\pi\varepsilon} \frac{(\mathbf{r}^* - \mathbf{r}_0)}{|\mathbf{r}^* - \mathbf{r}_0|^3} d\mathbf{r}^* = \frac{q}{4\pi\varepsilon} \frac{1}{|\mathbf{r} - \mathbf{r}_0|} \end{aligned} \quad (2.1.5)$$

Esto es en los casos en que se tenga una carga puntual, haciendo el paralelo con la ecuación (2.1.1). Para una carga distribuida se debe considerar la ecuación (2.1.3), y siguiendo la misma lógica se llega a

$$\phi(\mathbf{r}) = \int_{\Omega} \frac{\rho(\mathbf{r}_0)}{4\pi\varepsilon} \frac{1}{|\mathbf{r} - \mathbf{r}_0|} d\Omega \quad (2.1.6)$$

En los casos en que el medio en el que está la carga  $q$  no sea el vacío, se debe considerar que la materia que lo compone si reaccionará al campo eléctrico asociado a  $q$ ; en estos casos,  $\varepsilon \neq \varepsilon_0$ . Es decir, hay que aclarar que las ecuaciones (2.1.3) y (2.1.4) no están completas, pues falta considerar el efecto de la *polarización*. Además resulta interesante entender qué es realmente la permitividad, que está relacionada directamente con la polarización del medio; a continuación se explican estos puntos.

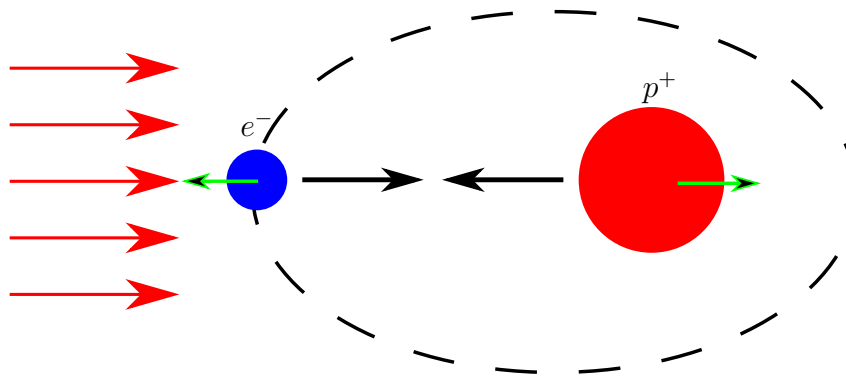
## 2.2 La permitividad de un medio material, la polarización y el campo eléctrico microscópico

Se puede entender la constante dieléctrica de un material (o permitividad) como una medida del reordenamiento de las partículas de un medio a nivel molecular frente a la presencia de un campo eléctrico externo, y provocando así que dichas partículas generen un *campo eléctrico inducido*, que es opuesto en sentido al original, pero no necesariamente de la misma intensidad; este disminuye el flujo neto del campo eléctrico externo, por lo que impide en cierta medida su paso, dando su nombre a la propiedad - en qué grado los átomos del material **permiten** el flujo de campo eléctrico.

La permitividad que se utiliza en las ecuaciones (2.1.1 - 2.1.6) es llamada también *constante dieléctrica* del medio, y en realidad es un concepto derivado de la susceptibilidad eléctrica y de lo que ocurre a nivel molecular en el medio dieléctrico afectado por un campo eléctrico. En el caso del vacío se menciona que no existe un medio que reaccione a la presencia de una carga (y por ende no “siente” el campo eléctrico), por lo que la constante dieléctrica del vacío es considerada como

referencia. Por el contrario, en un dieléctrico si se tiene una reacción al campo eléctrico; en este caso es importante explicar qué pasa con el campo eléctrico a nivel microscópico y cómo eso se relaciona con la susceptibilidad eléctrica. Los términos microscópico y local se utilizan indistintamente [18].

### 2.2.1 Campo eléctrico local y polarización



**Figura 2.1:** Efecto de un campo sobre la forma de un átomo. El electrón (en realidad la nube electrónica) es “halado” por el campo, mientras que el protón (o núcleo atómico) es “empujado”, induciendo así un dipolo.

Cuando se aplica un campo eléctrico externo a un cuerpo o medio (macroscópico) dieléctrico, las partículas constituyentes de los átomos se comportan de forma especial. Por ejemplo, en la figura 2.1 se muestra un átomo simple de hidrógeno que se reordena bajo la acción de un campo externo que rompe el equilibrio eléctrico del átomo, generando así un campo eléctrico opuesto con líneas de campo (inducido) que van del protón hacia el electrón (se representa la nube electrónica con un solo electrón por simplicidad). Este efecto es la polarización, e induce un momento dipolar  $\mathbf{p}$  en los átomos afectados por el campo eléctrico a nivel local.

$$\mathbf{p} = \alpha \vec{E}^{loc} \quad (2.2.1)$$

$\alpha$  se conoce como la polarizabilidad microscópica, un coeficiente de proporcionalidad que representa qué tanto se polariza la especie analizada - la magnitud del momento dipolar inducido - en presencia de un cierto campo eléctrico local  $\vec{E}^{loc}$ . Por supuesto, esta es una relación constitutiva, porque  $\alpha$  depende del material o elemento estudiado. Ahora se hace la distinción entre los distintos campos eléctricos de interés para este apartado:

$$\begin{aligned} \vec{E}^{ext} &\longrightarrow \text{Campo eléctrico externo} \\ \vec{E}^{loc} &\longrightarrow \text{Campo eléctrico local} \\ \vec{E}^{ind} &\longrightarrow \text{Campo eléctrico inducido (microscópico)} \\ \vec{E} &\longrightarrow \text{Campo eléctrico macroscópico} \end{aligned}$$

y se define la relación entre los primeros tres como

$$\vec{E}^{loc} = \vec{E}^{ind} + \vec{E}^{ext} \quad (2.2.2)$$

Es conveniente explicar que si bien se define uno de los campos como microscópico o local, esto no es el campo debido a un único átomo o molécula, sino el propio de un volumen macroscópicamente pequeño pero microscópicamente grande; por lo tanto, los términos local y microscópico son en realidad promedios sobre volúmenes más pequeños que el campo de las ecuaciones de Maxwell. Hay que notar también que si el campo eléctrico se transmite por el vacío,  $\vec{E}^{ind} = 0$ , y por ende  $\vec{E}^{ext} = \vec{E}^{loc} = \vec{E}$ . No se entra en detalle respecto al álgebra pues existen referencias sobre el tema, como por ejemplo el texto de Kan-

torovich [18], pero después de algo de trabajo se llega a la expresión para el campo eléctrico local, también llamado campo de Lorentz, en función del campo macroscópico

$$\vec{E}^{loc} = \vec{E}(1 + \kappa) \quad (2.2.3)$$

donde  $\kappa$  es un factor de corrección que depende del ordenamiento microscópico. El campo eléctrico local es el campo que siente cada uno de los muchos volúmenes pequeños que, en conjunto, componen el medio macroscópico que se está estudiando. Por último se presenta la definición de densidad de polarización  $\vec{P}$  para cerrar el círculo de relaciones entre el dieléctrico estudiado a nivel macroscópico y microscópico. La densidad de polarización por unidad de volumen se define de dos maneras; esto sale de diversas fuentes, pero W. Cai [7] lo resume todo de forma simple:

$$\vec{P}(\mathbf{r}) = \sum_i n_i \alpha_i \left[ \vec{E}(\mathbf{r})^{loc} \right]_i = \sum_i n_i \mathbf{p}_i \quad (2.2.4)$$

$$\vec{P}(\mathbf{r}) = \varepsilon_0 \chi \vec{E}(\mathbf{r}) \quad (2.2.5)$$

con  $n_i$  la densidad numérica (*number density*) del  $i$ -ésimo átomo por unidad de volumen, y  $\chi$  la susceptibilidad del medio dieléctrico (que en el caso del vacío es 0). Antes de continuar el desarrollo, se procede a definir qué son la densidad de polarización y la susceptibilidad.

Como concepto, la densidad de polarización es la polarización a nivel macroscópico, por unidad de volumen, causada por todos los momentos dipolares inducidos en las moléculas de dicho volumen asociados al

campo eléctrico que los induce. Por otra parte, la susceptibilidad permite relacionar un campo eléctrico macroscópico con los efectos que este tiene en el momento dipolar de las partículas en un volumen a través de la densidad de polarización, como se indica en la ecuación 2.2.5; más allá de ser una constante de proporcionalidad, la susceptibilidad indica de forma promediada cómo influye el campo eléctrico en la orientación de los dipolos, y cómo varía el momento dipolar del medio a nivel macroscópico en presencia del campo eléctrico. Hay que resaltar que esto no implica traslación o *migración* electrónica, sino solo rotación del átomo o molécula para la reorientación del dipolo, intentando alinearlos con el campo.

Para terminar esta parte del desarrollo teórico, de las ecuaciones (2.2.3 - 2.2.5) y sabiendo que el campo macroscópico es el mismo, se puede deducir que

$$\begin{aligned}\vec{P}(\mathbf{r}) &= \varepsilon_0 \chi \vec{E}(\mathbf{r}) = \vec{E}(\mathbf{r}) \sum_i n_i \alpha_i (1 + \kappa_i) \\ \Rightarrow \quad \varepsilon_0 \chi &= \sum_i n_i \alpha_i (1 + \kappa_i)\end{aligned}\tag{2.2.6}$$

Un caso particular es la teoría de Clausius-Mossotti. Aquí el valor de  $\kappa_i = \kappa$  es  $\chi/3$ ; uno de los supuestos es que la distribución molecular es una malla cúbica regular.

Con esto se puede entender de manera general cómo se relacionan el campo eléctrico macros-cópico y microscópico. El propósito de este apartado es simplemente aclarar que ambos análisis no son inde-

pendientes y que el tratamiento de continuo que se ve en lo sucesivo está fundamentado en teoría del comportamiento eléctrico molecular; es por esto que se omiten pasos y solo se presentan resultados que se consideran importantes.

### 2.2.2 Permitividad de un medio macroscópico

Ya se explicó qué son la susceptibilidad y densidad de polarización, por lo que se procede a definir un poco mejor qué es la permitividad, profundizando en lo que se mencionó al principio de la sección 2.2. Primero corresponde aclarar que la permitividad es netamente una propiedad macroscópica del medio, pero se prefiere caer en la redundancia un par de veces, como en el caso del título de esta sección, para reafirmar este punto. Cuando un campo eléctrico afecta un medio dieléctrico, los dipolos no solo se intentan alinear con las líneas de campo, sino que además hay un *movimiento de cargas libres* dentro del cuerpo. Sumado a esto está el hecho de que los átomos conformantes del medio rompen el estado de equilibrio, pues los protones son repelidos por el campo, mientras que los electrones son atraídos por este (figura 2.1). Dicho efecto genera que se formen nuevos dipolos y en consecuencia que haya un momento dipolar distinto al original en el medio; es por esta razón que la susceptibilidad se relaciona con la permitividad. Teniendo presentes los fenómenos recién nombrados, se reitera la definición de permitividad o constante dieléctrica como una cuantificación macroscópica promedio de la tendencia de un medio (en este caso un dieléctrico) a oponerse al flujo de campo eléctrico por alguno de los mecanismos nombrados anteriormente.

Existen tres *permitividades* de interés para este desarrollo: la per-



mitividad del vacío  $\varepsilon_0$ , la permitividad propia de un material  $\varepsilon$ , y la permitividad relativa o constante dieléctrica, a la que en este caso se da más importancia

$$\varepsilon_r = \frac{\varepsilon}{\varepsilon_0} = \varepsilon_0(1 + \chi) \quad (2.2.7)$$

(llamada  $k$  en algunas fuentes). De ahora en adelante,  $\varepsilon$  a secas pasa a ser la permitividad relativa y se denomina simplemente *permitividad* o *constante dieléctrica* de un medio.<sup>1</sup>

## 2.3 Desplazamiento eléctrico y campo eléctrico en un medio dieléctrico

En lo sucesivo, todo el trabajo es relativo a un medio o campo macroscópico. En este informe no se trabaja directamente con los momentos dipolares, la susceptibilidad, o incluso con la densidad de polarización, sino que se presenta una sola ecuación que engloba todos los efectos de un campo eléctrico sobre un dieléctrico; para llegar a dicha ecuación se utilizan las relaciones ya introducidas y un par de definiciones nuevas.

La ecuación (2.2.5) es una expresión que asocia el campo eléctrico y la densidad de polarización, establecida a través de la susceptibilidad y la permitividad. Además, la ecuación (2.1.3) es la ley de Gauss para una distribución de carga, que tiene que ver únicamente con la permitividad.

Si se asume que el flujo de campo eléctrico se debe a contribuciones

---

<sup>1</sup>En estricto rigor, la constante dieléctrica es  $\varepsilon_r \cdot \varepsilon_0$ , pero por costumbre se utiliza este término para referirse a la propiedad adimensional en este informe.

de la distribución de carga y de la polarización del medio únicamente, entonces se puede expresar dicho flujo de campo eléctrico como

$$\nabla \cdot \vec{E} = \frac{\rho - \nabla \cdot \vec{P}}{\varepsilon_0} \quad (2.3.1)$$

Se define entonces el *desplazamiento eléctrico* como

$$\vec{D} = \vec{E} + \vec{P} = \varepsilon_0 \vec{E} (1 + \chi) \quad (2.3.2)$$

lo que permite escribir la ecuación (2.3.1) como

$$\nabla \cdot \vec{D} = \rho \quad (2.3.3)$$

Esto corresponde al efecto completo del flujo de campo eléctrico sobre el medio. En general, en la ecuación (2.3.2) se asocia el campo eléctrico  $\vec{E}$  a las cargas libres (free charges) y la densidad de polarización  $\vec{P}$  a las cargas fijas (bound charges), pero para el objetivo de este texto eso no es relevante.

Siguiendo con la idea, recordando que ahora  $\varepsilon_r = \varepsilon$  y suponiendo que la permitividad es una propiedad isotrópica, al usar la ecuación (2.2.7) en (2.3.2), se puede obtener

$$\begin{aligned} \vec{D} &= \varepsilon \vec{E} \quad \Big/ \quad \nabla \cdot \\ \rightarrow \quad \nabla \cdot \vec{D} &= \nabla \cdot (\varepsilon \vec{E}) = \nabla(\varepsilon) \cdot \vec{E} + \varepsilon \nabla \cdot \vec{E} = \rho \end{aligned} \quad (2.3.4)$$

Aún más, si la permitividad del medio es constante sobre el mismo, entonces el término  $\nabla(\varepsilon) \cdot \vec{E}$  se anula. Es así que se retorna a la ecuación

de la que prácticamente parte el desarrollo de la teoría, la ley de Gauss.

$$\nabla \cdot \vec{E} = \frac{\rho}{\varepsilon} \quad \Longleftrightarrow \quad \nabla^2 \phi = \frac{\rho}{\varepsilon} \quad (2.3.5)$$

El gran progreso es haber demostrado que (2.1.4) no solo sirve para calcular el campo eléctrico en el vacío, sino que se puede aplicar a los dieléctricos en general y que incluso se consideran los efectos de polarización inducidos.

## 2.4 Método de imágenes

En la teoría electrostática hay ocasiones en que hay que trabajar con interfaces que separan distintos medios dieléctricos. El método de imágenes destaca por su simpleza en este tipo de escenarios. La idea es que cargas en un medio inducirán un campo opuesto en el otro, el denominado campo de reacción. A su vez este campo inducido afectará al campo en el primer medio, por lo que hay que encontrar una expresión para el comportamiento de los campos eléctricos en los dos medios; por supuesto, este problema no se soluciona simplemente con aplicar la ecuación (2.1.1) una vez.

El método de imágenes facilita la resolución del problema de una forma simple: En vez de considerar la polarización y el consecuente campo inducido por el segundo medio como un todo, se hace más fácil poner una carga o distribución de cargas imágenes (puede ser incluso una distribución continua) de tal forma que el efecto de dicha carga sobre el primer medio sea el mismo que si hubiera un campo de reacción. Para esto uno analiza los distintos medios por separado en vez de como un sistema.

Como la interfaz es un límite para el espacio de análisis, es aquí donde se establecen las condiciones de frontera. Para un sistema de coordenadas adecuado al problema, existen condiciones para la componente normal del campo y para su componente tangencial justo en la interfaz. Matemáticamente esto es

$$\varepsilon_1 \vec{E}_1 \cdot \hat{n} = \varepsilon_2 \vec{E}_2 \cdot \hat{n} \quad (2.4.1)$$

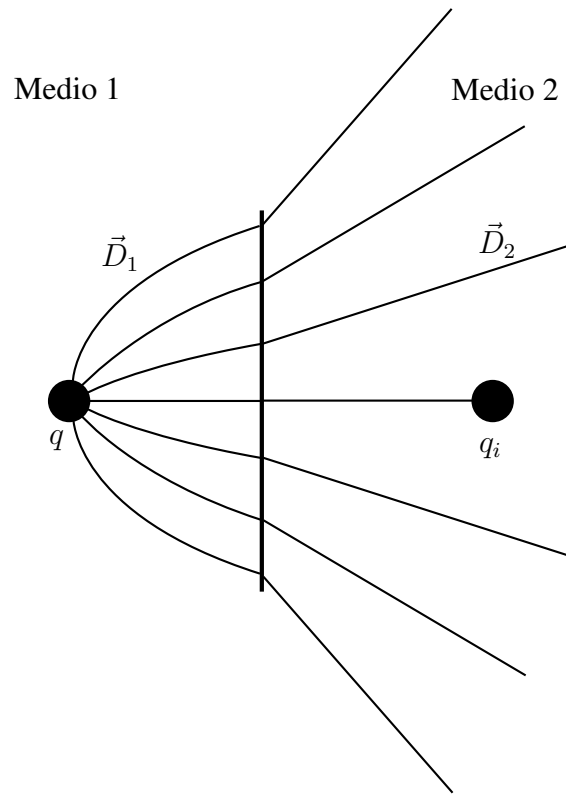
$$\vec{E}_1 \cdot \hat{t} = \vec{E}_2 \cdot \hat{t} \quad (2.4.2)$$

En este caso,  $\hat{n}$  es un vector normal a la superficie interfaz, y  $\hat{t}$  un vector tangencial a la misma. En palabras, las ecuaciones (2.4.1) y (2.4.2) indican que hay un cambio en la componente normal del campo igual a  $\varepsilon_1/\varepsilon_2$ ; esto es, hay un salto en la componente normal. Como ejemplo se tiene el caso más simple, con dos dieléctricos homogéneos isotrópicos ( $\varepsilon = cte$ ) en la figura 2.2. Hay que notar que, como se dijo, al trabajar de esta forma se analizan los medios por separado, y se debe considerar el efecto de la carga imagen que representa el campo de reacción solo en el medio 1. La idea es determinar qué valor tiene que tener la carga o conjunto de cargas necesarias, que pueden ser continuas o discretas, en función de los parámetros que si se conocen en el sistema. En el caso particular de que la interfaz sea una placa cargada de espesor infinitesimal y los medios a analizar tengan la misma constante dieléctrica, se llega al resultado elemental de un salto (discontinuidad) en la componente normal del campo eléctrico dado por

$$(\vec{E}_2 - \vec{E}_1) \cdot \hat{n} = \frac{\sigma}{\varepsilon_m} \quad (2.4.3)$$

con  $\sigma$  la densidad superficial de carga y  $\varepsilon_m$  la constante dieléctrica de los medios que rodean al plano.

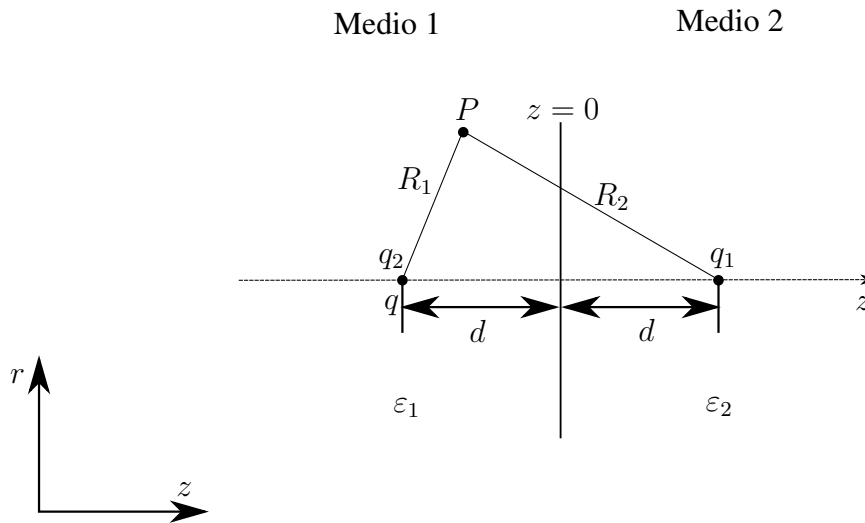
Se desarrolla más a cabalidad el tema en secciones posteriores, ya que forma parte del problema clave que motiva este escrito. Se presenta un desarrollo del caso planteado con el sistema de ecuaciones, y se analiza un sistema con dos interfaces paralelas, todo mediante el método de imágenes.



**Figura 2.2:** Se observan las líneas de desplazamiento eléctrico, que dan cuenta del efecto de la polarización.

### 2.4.1 Campo de reacción en un sistema con una interfaz y una carga puntual

En primer lugar se tiene un caso simple, presente en la mayoría de los libros de electrodinámica, que consiste en una carga en un medio 1 y una interfaz que separa a este último de un medio 2; el caso en cuestión se presenta en la figura 2.3. El campo eléctrico viene dado por la ecuación (2.1.1), y tiene como condiciones de contorno las presentadas en las Ecs. (2.4.1) y (2.4.2). El potencial en algún punto  $P$  en el medio 1 viene dado por el potencial de una carga  $q$  real y la carga imagen  $q_1$ , que representa el campo de reacción del medio 2 a causa de  $q$ . Además el potencial en el medio 2 ya no es debido únicamente a la carga  $q$ , pues el medio dieléctrico reacciona; esto se representa por una nueva carga imagen  $q_2$ . En este caso,  $R_{1(2)} = \sqrt{r^2 + (z \mp d)^2}$ , el signo dependiendo de si la carga, ya sea imaginaria o real, esta hacia la izquierda o derecha del plano  $z = 0$ .



**Figura 2.3:** Sistema de una carga real y dos imaginarias para representar el campo de reacción, con el cambio de medio en  $z = 0$ . Se consideran dos espacios semi-infinitos a partir de este plano.

No es complejo realizar el desarrollo, remplazando la expresión para el campo en las condiciones de contorno. Con un poco de álgebra se puede llegar a una forma de escribir las cargas  $q_1$  y  $q_2$  en función de la carga  $q$ , así como de las permitividades de los medios. Se muestra a continuación un paso a paso breve de cómo se resuelve el sistema. Si  $\phi_1$  es el potencial para algún punto en el medio 1, y  $\phi_2$  es el potencial en el medio 2, se tiene

$$\phi_1 = \frac{1}{4\pi\epsilon_1} \left( \frac{q}{\sqrt{r^2 + (z-d)^2}} + \frac{q_1}{\sqrt{r^2 + (z+d)^2}} \right) \quad (2.4.4)$$

$$\phi_2 = \frac{1}{4\pi\epsilon_2} \left( \frac{q_2}{\sqrt{r^2 + (z-d)^2}} \right) \quad (2.4.5)$$

Derivando estas expresiones y remplazándolas en la condición de contorno (2.4.1) se obtiene el siguiente sistema de ecuaciones  $2 \times 2$

$$\begin{aligned} & \frac{\partial}{\partial z} \left[ \epsilon_1 \cdot \left( \frac{q}{4\pi\epsilon_1} \cdot \frac{1}{\sqrt{r^2 + (z-d)^2}} + \frac{q_1}{4\pi\epsilon_1} \frac{1}{\sqrt{r^2 + (z+d)^2}} \right) \right]_{z=0} = \\ & \frac{\partial}{\partial z} \left[ \epsilon_2 \cdot \left( \frac{q_2}{4\pi\epsilon_2} \frac{1}{\sqrt{r^2 + (z-d)^2}} \right) \right]_{z=0} \\ \Rightarrow & d \cdot \left( \frac{q}{\sqrt{(r^2 + (-d)^2)^3}} - \frac{q_1}{\sqrt{(r^2 + d^2)^3}} \right) = d \cdot \left( \frac{q_2}{\sqrt{(r^2 + (-d)^2)^3}} \right) \\ \Rightarrow & q - q_1 = q_2 \end{aligned} \quad (2.4.6)$$

Para el caso de la condición de contorno (2.4.2) se remplazan también las expresiones de  $\partial_r \phi_1$  y  $\partial_r \phi_2$ , pero en este caso no se simplifican las

permitividades, por lo que se llega a una expresión que las incluye:

$$\begin{aligned}
 & \frac{\partial}{\partial r} \left( \frac{q}{4\pi\epsilon_1} \frac{1}{\sqrt{r^2 + (z-d)^2}} + \frac{q_1}{4\pi\epsilon_1} \frac{1}{\sqrt{r^2 + (z+d)^2}} \right)_{z=0} = \\
 & \frac{\partial}{\partial r} \left( \frac{q_2}{4\pi\epsilon_2} \frac{1}{\sqrt{r^2 + (z-d)^2}} \right)_{z=0} \\
 \Rightarrow & \frac{r}{\epsilon_1} \cdot \left( \frac{q}{\sqrt{(r^2 + (-d)^2)^3}} + \frac{q_1}{\sqrt{(r^2 + d^2)}} \right) = \frac{r}{\epsilon_2} \cdot \left( \frac{q_2}{\sqrt{(r^2 + (-d)^2)^3}} \right) \\
 \Rightarrow & \frac{1}{\epsilon_1} (q + q_1) = \frac{q_2}{\epsilon_2} \tag{2.4.7}
 \end{aligned}$$

Ahora, del sistema que conforman las ecuaciones (2.4.6) y (2.4.7), se obtiene la expresión para las cargas imagen en función de los parámetros conocidos del sistema:

$$q_1 = - \left( \frac{\epsilon_2 - \epsilon_1}{\epsilon_2 + \epsilon_1} \right) \cdot q \tag{2.4.8}$$

$$q_2 = \left( \frac{2\epsilon_2}{\epsilon_1 + \epsilon_2} \right) \cdot q \tag{2.4.9}$$

Luego, el potencial en cualquier punto en el espacio se trabaja dependiendo de en qué medio se ubica dicho punto, y para calcular su valor se utilizan las ecuaciones (2.4.4) y (2.4.5) dependiendo de dónde se encuentra; por supuesto, esto remplazando  $q_1$  y  $q_2$  por sus valores de las ecuaciones (2.4.8) y (2.4.9).

## 2.5 El Método de Elementos de Borde

Previamente ya se mencionó el uso del método de elementos de borde (BEM) para aplicar el modelo derivado del método de imágenes para membranas. El método BEM actúa partiendo del mismo principio de



debilitar una ecuación diferencial que el método de elementos finitos FEM, pero en este caso la discretización no ocurre sobre el dominio de integración de la ecuación diferencial, sino sobre su contorno.

El método comienza tomando la formulación débil de una ecuación diferencial. En este caso se utiliza la ecuación de Laplace, multiplicada por una función  $\psi$  tal que

$$\int \nabla^2 \phi \cdot \psi = 0$$

Con esto como punto de partida, se procede a debilitar la ecuación de Laplace, integrando por partes

$$\int_{\Omega} \nabla^2 \phi \cdot \psi \, d\Omega = \int_{\Omega} \nabla(\nabla \phi \cdot \psi) - \nabla \phi \cdot \nabla \psi \, d\Omega = 0$$

Respecto al integrando del lado derecho, si se aplica el teorema de la divergencia al primer término y se integra por partes el segundo, se llega a

$$\int_{\partial\Omega} (\nabla \phi \cdot \psi) \cdot \hat{\mathbf{n}} \, d\partial\Omega - \int_{\Omega} [\nabla \cdot (\nabla \psi \cdot \phi) - \nabla^2 \psi \cdot \phi] \, d\Omega = 0$$

A continuación, al aplicar nuevamente el teorema de la divergencia al primer término de la segunda integral, se obtienen dos integrales de superficie y una de volumen.

$$\int_{\partial\Omega} (\nabla \phi \cdot \psi - \nabla \psi \cdot \phi) \cdot \hat{\mathbf{n}} \, d\partial\Omega = - \int_{\Omega} \phi \cdot \nabla^2 \psi \, d\Omega \quad (2.5.1)$$

Si en particular  $\nabla^2 \psi$  corresponde a una función delta de Dirac ( $\delta(\mathbf{r})$ ), entonces la integral del lado derecho es  $\phi$  evaluado en los puntos seleccionados por la función  $\delta$ . Para que se cumpla  $\nabla^2 \psi = -\delta(\mathbf{r}^*)$ ,  $\psi$  debe

ser una función de Green

$$\psi = G(\mathbf{r}, \mathbf{r}^*) = \frac{k}{|\mathbf{r} - \mathbf{r}^*|} \quad (2.5.2)$$

donde  $k$  es una constante que no afecta la proporcionalidad inversa respecto del radio. Aquí  $\mathbf{r} = \mathbf{r}_\Omega$  y  $\mathbf{r}^* = \mathbf{r}_\Omega^*$ , significando que ambos corren sobre el mismo dominio de integración, pero son independientes uno del otro. De esta manera, utilizando una función como la (2.5.2), la ecuación (2.5.1) queda con la forma final

$$\begin{aligned} \int_{\Omega} \phi(\mathbf{r}_\Omega) \delta(\mathbf{r}_\Omega^*) d\Omega = \\ \phi(\mathbf{r}_\Omega^*) = \int_{\partial\Omega} \left[ \frac{\partial}{\partial \hat{\mathbf{n}}} \phi(\mathbf{r}_{\partial\Omega}) \cdot G(\mathbf{r}_\Omega, \mathbf{r}_{\partial\Omega}) - \right. \\ \left. \frac{\partial}{\partial \hat{\mathbf{n}}} G(\mathbf{r}_\Omega, \mathbf{r}_{\partial\Omega}) \cdot \phi(\mathbf{r}_{\partial\Omega}) \right] d\partial\Omega \end{aligned} \quad (2.5.3)$$

En los casos en que se produzca una singularidad debido a que  $\mathbf{r}_\Omega \rightarrow \mathbf{r}_{\partial\Omega}$ , los valores de las integrales de superficie son

$$\int_{\partial\Omega} \frac{\partial}{\partial \hat{\mathbf{n}}} \phi \cdot G = 0 \quad (2.5.4)$$

$$\int_{\partial\Omega} \frac{\partial}{\partial \hat{\mathbf{n}}} G \cdot \phi = -\frac{\phi}{2} \quad (2.5.5)$$

En este escrito se omite la demostración correspondiente debido a que no se considera relevante más allá del resultado, pero un desarrollo metódico se puede encontrar en textos de la materia, como por ejemplo el libro de Brebbia [6].

En síntesis, la ecuación (2.5.3) permite obtener  $\phi$  en el dominio  $\Omega$  contenido por una superficie  $\partial\Omega$ ; por otro lado, para calcular  $\phi$  sobre el

contorno  $\partial\Omega$ , se aplican los resultados (2.5.4) y (2.5.5) en la ecuación (2.5.3) y se obtiene

$$\frac{\phi(\mathbf{r}_{\partial\Omega})}{2} = \int_{\partial\Omega} \frac{\partial}{\partial \hat{\mathbf{n}}} \phi(\mathbf{r}_{\partial\Omega}) \cdot G(\mathbf{r}_{\partial\Omega}, \mathbf{r}_{\partial\Omega}^*) - \frac{\partial}{\partial \hat{\mathbf{n}}} G(\mathbf{r}_{\partial\Omega}, \mathbf{r}_{\partial\Omega}^*) \cdot \phi(\mathbf{r}_{\Omega}) d\partial\Omega \quad (2.5.6)$$

En ambos casos ( $\mathbf{r}_{\partial\Omega}$  y  $\mathbf{r}_{\partial\Omega}^*$ ) la aplicación es sobre la misma superficie  $\partial\Omega$ . Es importante considerar que la formulación de la Boundary Integral Equation (BIE) que se ha hecho hasta ahora es para la ecuación de Laplace en 3D. En caso de que la ecuación gobernante del fenómeno bajo estudio sea otra, corresponde hacer una derivación para dicha situación. Afortunadamente, para lo desarrollado en el presente texto no hace falta más.

## 2.6 El contexto biológico de la membrana celular

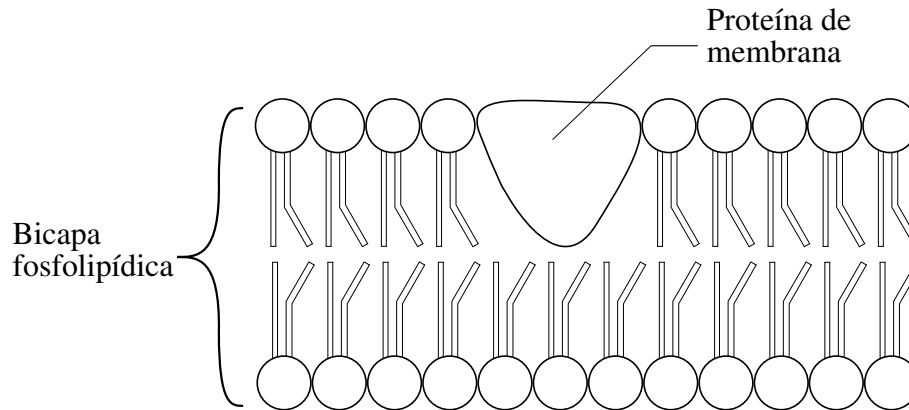
A pesar de haber mencionado la teoría sobre el comportamiento de cargas en un medio, el método de imágenes, y el método BEM que permite llevar todo esto a aplicaciones numéricas, estos pueden aún parecer tópicos un tanto aislados unos de otros, e incluso abstractos en cierta medida. Como consecuencia se presenta el contexto biológico que motiva la búsqueda de un modelo para integrar el efecto de la membrana celular en el que se encuentran insertas algunas proteínas.

Una célula se compone de distintas estructuras funcionales que permiten transformar moléculas en energía, enviar y recibir señales de tipo químico y eléctrico, permitir el paso de compuestos desde y hacia el exterior de la célula, entre otros. Las proteínas son probablemente el tipo de macromolécula con la mayor diversidad de funciones en el

entorno celular, y su importancia en el quehacer de la membrana, sea este el flujo de elementos o la conducción de señales, por ejemplo, es fundamental. La membrana plasmática se compone de una cadena de fosfolípidos, uno al lado del otro, que rodean el citoplasma (fluido intracelular) y separan el interior de la célula, con estructuras y todo, del fluido extracelular. Más aún, la membrana celular, también llamada bicapa fosfolipídica debido a su composición, es la encargada de definir la frontera de ciertos organelos de la célula; las lisosomas, mitocondrias y el mismo núcleo celular son algunos.

Los fosfolípidos son un tipo de molécula conformado por dos “colas” de ácidos grasos que se unen a un glicerol. Para completar la molécula, también hay un fosfato enlazado al glicerol, que lleva por el lado opuesto una unión con un grupo de carácter polar. Para efectos de este trabajo, lo que hay que rescatar es el hecho de que las colas son hidrófobas, mientras que la “cabeza” conformada por el glicerol, el fosfato y el compuesto polar - generalmente una colina - son la parte hidrófila del fosfolípido. La figura 2.4 muestra una sección de la membrana, mientras que la figura 2.5 corresponde al detalle de un fosfolípido.

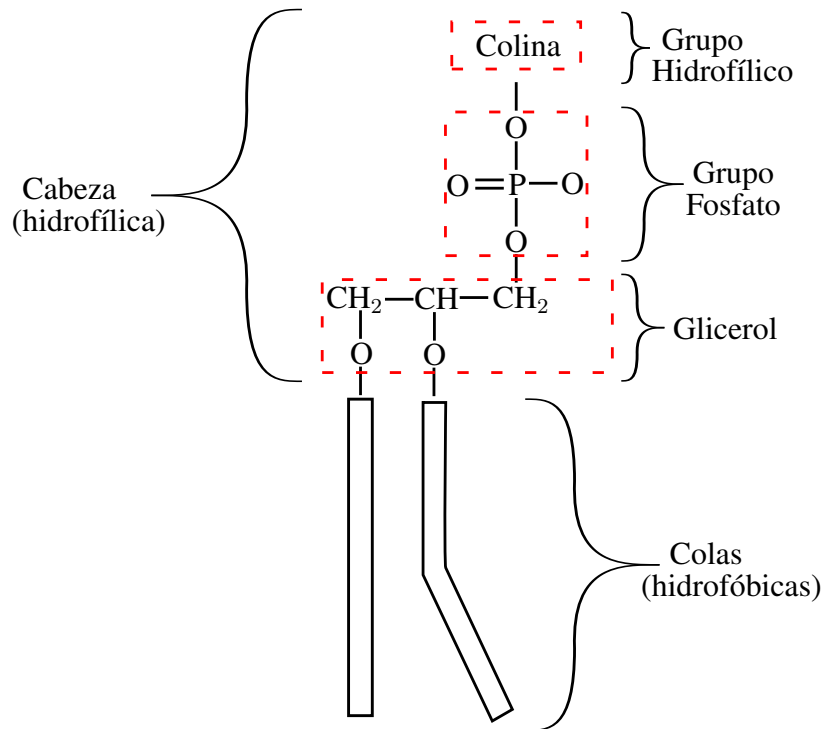
La importancia de la hidrofobicidad e hidrofiliidad de las colas y cabezas respectivamente, es la orientación que toman los fosfolípidos al conformar la membrana. Esto es lo mismo que sucede con las proteínas insertas en la membrana, ocupando lugares en la pared celular que permiten minimizar la exposición al agua de los aminoácidos conformantes que tengan una pobre afinidad al agua y maximizando la de los que tengan una afinidad alta.



**Figura 2.4:** Tramo de la bicapa con proteína inserta. Esta última se orienta de tal forma que la parte hidrofílica (polar) quede en contacto con el agua en el exterior, de igual forma que las cabezas de los fosfolípidos.

La pared celular no tiene un espesor definido, sino que varía en función de la sección que se está analizando y de las proteínas de membrana que haya insertas, y de forma más considerable según qué célula se está estudiando. Sin embargo, actualmente existe un valor aceptado y definido para *células estándar* de 8 [nm] [20]. Al trabajar con las proteínas embebidas y la mayoría de los eventos que ocurren y tienen relación con la membrana celular, el espesor es la única dimensión de importancia y para efectos prácticos la membrana puede considerarse un mar infinito de fosfolípidos y proteínas.

Como última propiedad de interés, se menciona que la membrana no es en ningún caso plana, ni tampoco rígida. Por el contrario, los fosfolípidos están en constante movimiento, provocando una ondulación en la bicapa, permitiendo entre otras cosas la formación de vacuolas y el desplazamiento de la célula. Entre los movimientos en que incurren los fosfolípidos están el desplazamiento “horizontal” (siguiendo la referencia de la figura 2.4), que se efectúa en el plano de la membrana y es muy común, y el denominado “*flip - flop*”, que corresponde a un



**Figura 2.5:** Esquema de un fosfolípido, con detalle sobre la composición de la cabeza hidrófila [13].

cambio de ubicación entre dos fosfolípidos de forma transversal a la membrana; se puede decir que es un intercambio de lugar.

La membrana es la estructura fundamental de este estudio, pero es importante no dejar fuera el medio en el que está inmerso. Ya se nombraron los fluidos intra y extracelular, por lo que corresponde caracterizarlos un poco. El primer factor a considerar es que el ambiente en el que está la membrana es principalmente salino o, dicho más formalmente, una disolución con una cierta concentración de especies iónicas. En segundo lugar, dicha solución no es homogénea, sino que sigue una distribución que, junto a ciertas propiedades como la diferencia de potencial, propicia el intercambio iónico a través de la membrana.

---

## Capítulo 3

# Metodología, funcionamiento e implementación

Actualmente existen dos modelos ampliamente utilizados para representar un solvente. Uno se basa en una consideración de cada átomo que conforma el solvente de forma explícita, modelando el problema con dinámica molecular, y la otra en un modelo continuo, en el que se trabaja el solvente como un medio dieléctrico con propiedades representativas ponderadas.

El primer modelo tiene la gran ventaja de que permite una precisión enorme, pues es una consideración de todas las partículas de manera explícita, una por una, y los fenómenos que ocurren en todas (o muchas de) las interacciones. El gran nivel de detalle de este modelo resulta también ser su gran debilidad, pues la cantidad de cálculos que hay que hacer depende del número de partículas del sistema, significando un gran costo computacional.

El segundo modelo permite omitir la consideración de partículas individuales al seguir un enfoque continuo, pero al no ser una representación detallada del sistema, el modelo cae en impresiones; un ejemplo de esto es no tomar en cuenta las fuerzas de van der Waals u otros fenómenos que ocurren a distancias muy pequeñas. A continuación

se empieza a ensamblar la teoría presentada hasta ahora, explicando cómo el método BEM permite modelar una proteína inmersa en una solución a través de una discretización de la interfaz que los separa, cómo se genera dicha interfaz y qué es lo que se busca a partir de estos cálculos.

### 3.1 Representación del modelo de solvente implícito mediante BEM

Ya se caracterizó el modelo de solvente implícito y cómo funciona. Un resultado de esto es, en el caso de que haya una distribución de iones libres en el solvente, la ecuación de Poisson - Boltzmann, que en resumidas cuentas toma la ecuación de Poisson y aplica una distribución de Boltzmann para incluir el efecto de los iones en una solución, por ejemplo salina.

$$-\nabla \cdot (\varepsilon(\mathbf{r}) \cdot \nabla \phi(\mathbf{r})) = \rho(\mathbf{r}) + \sum_i e z_i n_i^0 \cdot \exp\left(-\frac{e z_i \phi}{k_B T}\right) \quad (3.1.1)$$

donde  $e$  es el valor de la carga del electrón,  $z_i$  la carga de la  $i$ -ésima especie iónica,  $n_i^0$  es el number density de dicha especie de ion,  $k_B$  es la constante de Boltzmann y  $T$  la temperatura del sistema. Como esta ecuación varía de forma exponencial, se busca linealizarla para facilitar su aplicación en cálculos

$$-\nabla \cdot (\varepsilon(\mathbf{r}) \cdot \nabla \phi(\mathbf{r})) = \rho(\mathbf{r}) + \sum_i e z_i n_i^0 \cdot \left(1 - \frac{e z_i \phi}{k_B T}\right) \quad (3.1.2)$$

La suma es sobre todas las especies iónicas en la solución. Tras el supuesto de que esta última no tiene una carga neta, sino que es neu-



tra ( $\sum_i e z_i n_i^0 = 0$ ) y si además no hay una dependencia espacial de  $\varepsilon$ , se llega a la ecuación de Poisson - Boltzmann utilizada comúnmente en los modelos implícitos

$$\begin{aligned} \nabla^2 \phi(\mathbf{r}) - \kappa^2 \phi(\mathbf{r}) &= -\frac{1}{\varepsilon} \rho(\mathbf{r}) \\ \kappa^2 &= \frac{e}{\varepsilon k_B T} \sum_i e n_i^0 z_i^2 \end{aligned} \quad (3.1.3)$$

Siendo  $\kappa$ , llamado el inverso del largo de Debye, una medida del apantallamiento eléctrico a causa de las cargas móviles de los iones en solución.

Ahora, esto sirve más que nada de contexto para introducir la forma de trabajo del método BEM, así como los conceptos de Solvent Accesible Surface, Solvent Excluded Surface y energía de solvatación. Además es interesante comprender cómo atacar una distribución iónica en electrostática continua.

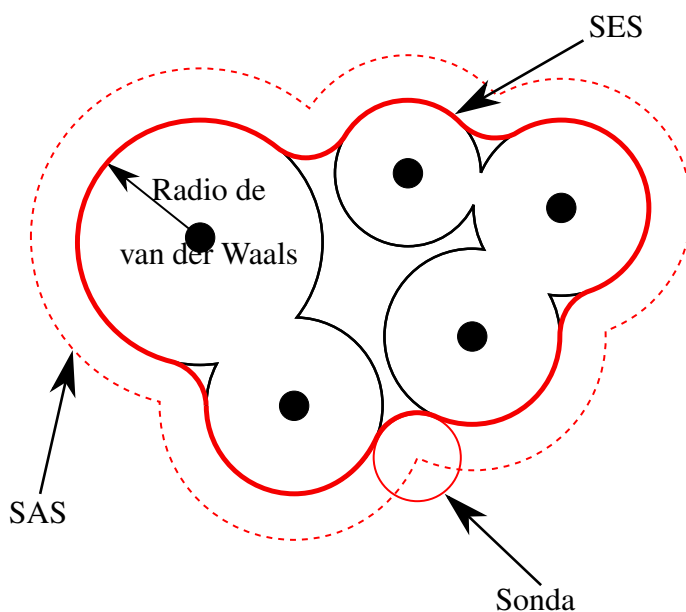
El método BEM requiere una superficie que permita diferenciar dos dominios. Es esta interfaz la que se discretiza y sobre la cual se integran las ecuaciones que modelan la propiedad a trabajar. En el caso de una proteína en un solvente existen por lo menos dos, pues la interfaz es una y hay un medio a cada lado. En primer lugar es importante saber cómo se establece esta superficie.

### 3.1.1 Solvent Excluded y Solvent Accesible Surface

De lo que se ha percibido a lo largo de este estudio, los dos métodos que se emplean usualmente para representar una superficie característica de una proteína son los que se indican en el título de esta parte. Ambos

consisten en utilizar una *sonda*, por lo general una molécula de agua, que interactúa con los radios de van der Waals de los átomos de la proteína. Si por ejemplo se quiere mapear la superficie exterior de la proteína o macromolécula, se hace “rodar” una molécula de agua por sobre la superficie de van der Waals de los átomos exteriores, de tal modo que la distancia entre los centros de un átomo de la macromolécula y el de la sonda de agua sea la suma de los radios de cada uno.

En el caso de la Solvent Accesible Surface (SAS), el centro de la molécula de agua es lo que define la superficie de trabajo. Esto genera puntas como se puede apreciar en la figura 3.1 . La distancia entre el centro del átomo a sondear y la SAS es entonces la suma del radio de van der Waals de dicho átomo y el radio de la molécula de sondeo.



**Figura 3.1:** Solvent Excluded y Solvent Accesible Surface, presentadas de forma esquemática.

Por otra parte, la Solvent Excluded Surface (SES) considera el borde de

la sonda como la definición de la superficie de la macromolécula. La superficie generada es por tanto suave, pues siempre consta al menos de una sección de arco de la sonda. La distancia entre el centro de un átomo de la macromolécula y la SES ya no está definida de forma tan clara como en el caso de la SAS, y a pesar de que en algunos casos coincide con ser simplemente el radio de van der Waals de un átomo de la proteína, esto no es una regla.

A pesar de las diferencias, no existe una razón para preferir una sobre la otra, y depende simplemente de lo que se busque lograr en cada situación. En el caso de este trabajo, y principalmente por heredar la metodología que se venía aplicando de antes, se utiliza la Solvent Excluded Surface para definir la superficie de las moléculas de estudio, puesto que, como se dijo, la superficie resultante es suave.

### 3.1.2 La ecuación de Poisson - Boltzmann (PB) y la Boundary Integral Equation (BIE)

Teniendo ya la ecuación PB, se presenta rápidamente el caso del sistema para una proteína en un solvente con iones; el objetivo de esto es mostrar cómo se usa el método BEM, aplicado específicamente a una discretización de superficie proteínica. Para esto hay que considerar que hay dos medios en el sistema, y una interfaz: el sistema exterior, que como es usual consiste en agua salina  $\Omega_1$ , el ensamblaje de  $N_q$  átomos con  $q_j$  cargas netas que conforman la proteína y que será el sistema interior  $\Omega_2$ , y la Solvent Excluded Surface que se asume separa ambos dominios recién nombrados, encerrando al segundo de estos.

Las ecuaciones corresponden entonces a Poisson por dentro, pues

existe una distribución de cargas puntuales en un medio dieléctrico, y Poisson - Boltzmann por fuera, lo que considera la teoría explicada anteriormente.

$$\begin{aligned}
 \text{En el soluto:} \quad & \nabla^2 \phi_1(\mathbf{r}) = -\frac{\rho(\mathbf{r})}{\varepsilon} \\
 \text{En el solvente} \quad & \nabla^2 \phi_2(\mathbf{r}) = \kappa^2(\mathbf{r}) \phi_2(\mathbf{r}) \\
 \text{Con condiciones de borde:} \quad & \frac{\partial \phi_1}{\partial \hat{\mathbf{n}}} = \frac{\varepsilon_2}{\varepsilon_1} \frac{\partial \phi_2}{\partial \hat{\mathbf{n}}} \quad ; \quad \phi_1 = \phi_2
 \end{aligned}$$

De lo presentado sobre el método BEM en la sección 2.5, se sabe que el sistema cumple con la ecuación (2.5.3). Esta es válida para ambas direcciones hacia la interfaz. Es decir, si es que la aproximación a la superficie de la proteína ocurre por fuera, la ecuación es igualmente válida que si ocurre por dentro. Ambos dominios, y por ende ambas ecuaciones gobernantes, se encuentran en la SES, por lo que es ahí donde se aplican las condiciones de contorno de electrostática presentadas en la sección 2.4, específicamente (2.4.1) y (2.4.2). Llevando ambas ecuaciones a una formulación integral sobre la frontera, el sistema  $2 \times 2$  para una proteína en solución con iones queda [11]

$$\begin{aligned}
 \frac{\phi_1(\mathbf{r}^*)}{2} - \int_{\partial\Omega} \frac{\partial}{\partial \hat{\mathbf{n}}} G_Y(\mathbf{r}, \mathbf{r}^*) \cdot \phi(\mathbf{r}) d\partial\Omega + \\
 \int_{\partial\Omega} \frac{\partial}{\partial \hat{\mathbf{n}}} \phi(\mathbf{r}) \cdot G_Y(\mathbf{r}, \mathbf{r}^*) d\partial\Omega = 0
 \end{aligned} \tag{3.1.4}$$

$$\begin{aligned}
 \frac{\phi_2(\mathbf{r}^*)}{2} + \int_{\partial\Omega} \frac{\partial}{\partial \hat{\mathbf{n}}} G_L(\mathbf{r}, \mathbf{r}^*) \cdot \phi(\mathbf{r}) d\partial\Omega - \\
 \int_{\partial\Omega} \frac{\partial}{\partial \hat{\mathbf{n}}} \phi(\mathbf{r}) \cdot G_L(\mathbf{r}, \mathbf{r}^*) d\partial\Omega = \frac{1}{4\pi\varepsilon} \sum_{j=0}^{N_q} \frac{q_j}{\mathbf{r} - \mathbf{r}_j}
 \end{aligned} \tag{3.1.5}$$

Se puede reconocer, con algo de imaginación, que (3.1.5) es la ecuación de Poisson de electrostática (2.3.5), con una distribución de cargas puntuales, siendo todo el lado izquierdo de la igualdad el potencial sobre la superficie. Este último hecho también es válido para (3.1.4); el lado izquierdo es el potencial sobre la superficie. Las variables de  $\mathbf{r}$  y  $\mathbf{r}^*$  corresponden a una evaluación sobre la superficie  $\partial\Omega$ , mientras que  $\mathbf{r}_j$  son las posiciones de las mencionadas cargas puntuales. Los subíndices  $L$  e  $Y$  son propios de *Laplace* y *Yukawa*. La función de Green para la ecuación de Laplace ya se presentó en la sección 2.5, mientras que la función de Green desarrollada por Yukawa, que corresponde a la ecuación de Poisson - Boltzmann, es

$$G_Y(\mathbf{r}, \mathbf{r}^*) = \frac{\exp(-\kappa \cdot |\mathbf{r} - \mathbf{r}^*|)}{4\pi|\mathbf{r} - \mathbf{r}^*|} \quad (3.1.6)$$

Para simplificar la notación, se presentan los operadores Single y Double-Layer Potential.

$$\text{Single-Layer Potential: } V^r(\phi(\mathbf{r})) = \int_{\partial\Omega} \phi(\mathbf{r}^*) \cdot G(\mathbf{r}, \mathbf{r}^*) d\partial\Omega^* \quad (3.1.7)$$

$$\text{Double-Layer Potential: } K^r(\phi(\mathbf{r})) = \int_{\partial\Omega} \phi(\mathbf{r}^*) \cdot \frac{\partial}{\partial \hat{\mathbf{n}}} G(\mathbf{r}, \mathbf{r}^*) d\partial\Omega^* \quad (3.1.8)$$

En este trabajo no se utilizan los operadores de Yukawa, pero si los de Laplace, por lo que es conveniente presentarlos de forma explícita.

$$V_L(\phi(\mathbf{r})) = \int_{\partial\Omega} \phi(\mathbf{r}^*) \cdot \frac{1}{4\pi|\mathbf{r} - \mathbf{r}^*|} d\partial\Omega^* \quad (3.1.9)$$

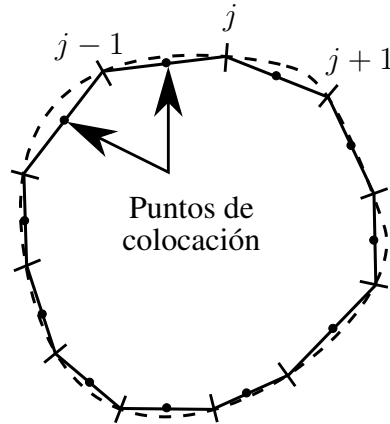
$$K_L(\phi(\mathbf{r})) = \int_{\partial\Omega} \phi(\mathbf{r}^*) \cdot \frac{1}{4\pi} \frac{\partial}{\partial \hat{\mathbf{n}}} \frac{1}{|\mathbf{r} - \mathbf{r}^*|} d\partial\Omega^* \quad (3.1.10)$$

Al aplicar las condiciones de contorno, el sistema de ecuaciones deriva en

$$\frac{\phi_2(\mathbf{r})}{2} + K_L(\phi_2(\mathbf{r})) - V_L \left( \frac{\partial}{\partial \hat{\mathbf{n}}} \phi_2(\mathbf{r}) \right) = \frac{1}{4\pi\epsilon_2} \sum_{j=0}^{N_q} \frac{q_j}{|\mathbf{r} - \mathbf{r}_j|} \quad (3.1.11)$$

$$\frac{\phi_2(\mathbf{r})}{2} - K_Y(\phi_2(\mathbf{r})) + \frac{\epsilon_2}{\epsilon_1} V_Y \left( \frac{\partial}{\partial \hat{\mathbf{n}}} \phi_2(\mathbf{r}) \right) = 0 \quad (3.1.12)$$

Ahora bien, si se subdivide la superficie de la proteína en  $N$  paneles, en este caso triangulares, las ecuaciones siguen siendo válidas para cada panel; estas se trabajan *colocándolas* en los centroides de cada panel (llamado a partir de ahora *punto de colocación*), permitiendo así su discretización de forma representativa.



**Figura 3.2:** Forma 2D de dividir la frontera del dominio de integración para aplicar el método BEM a cada elemento de borde.

Las propiedades que se buscan - potencial y su derivada respecto a la normal - son incógnitas. Se asume entonces que dichas propiedades son constantes sobre un mismo panel, por lo que salen de la integral respectiva en ambas ecuaciones y se genera un sistema de la forma  $\underline{A}\mathbf{x} = \mathbf{b}$

$$\begin{pmatrix} \frac{1}{2} + K_L & -V_L \\ \frac{1}{2} - K_Y & V_Y \end{pmatrix} \begin{pmatrix} \phi \\ \frac{\partial}{\partial \mathbf{n}} \phi \end{pmatrix} = \begin{pmatrix} \frac{1}{4\pi\epsilon} \sum_{j=0}^{N_q} \frac{q_j}{|\mathbf{r}-\mathbf{r}_j|} \\ 0 \end{pmatrix} \quad (3.1.13)$$

Resolviendo (3.1.13) para los  $N$  paneles se calcula la energía potencial del sistema, y multiplicando por una serie de factores de transformación de unidades, se obtiene la energía sobre la superficie de la proteína. Con estos valores se puede entrar a calcular la reacción del medio sobre dicha superficie, lo que corresponde a la energía de solvatación. Se adjunta una definición de *energía de solvatación* en el Apéndice B de este escrito. Se profundiza más sobre el cálculo energético en el siguiente apartado, particularmente en la descripción del programa PyGBE.

## 3.2 Descripción del software utilizado

En el transcurso de este trabajo se hizo uso de una serie de programas para ayudar a implementar el modelo encontrado, probarlo, o comparar los resultados obtenidos. Se mencionan y describen en esta sección, en mayor o menor profundidad, los cuatro softwares sin los cuales esta memoria no podría haberse realizado.

### 3.2.1 MSMS

Este resulta el programa menos utilizado directamente en el trabajo, pero que sin embargo es fundamental y de hecho es empleado de una u otra forma por los otros en esta sección. MSMS permite mallar la superficie de un átomo o conjunto de átomos en paneles triangulares, entregando la Solvent Excluded Surface (SES) en función de un “radio de sondeo”,

habilitando además al usuario para definir una variedad de parámetros que facilitan obtener modelos para realizar diferentes pruebas. El programa está escrito en lenguaje C y arroja un par de archivos, uno con una numeración para identificar los triángulos, y el otro con las coordenadas de los vértices de dichos triángulos. Más allá de esto, no es necesario ahondar en el funcionamiento del programa, pues su funcionalidad es lo que aquí importa.

### 3.2.2 APBS

Adaptive Poisson-Boltzmann Solver, más conocido por su sigla APBS, es un código escrito en C que permite calcular distintos tipos de energías en un sistema molécula o macromolécula - solvente. De particular interés resulta en este caso la energía de solvatación de una molécula o macromolécula. El programa emplea el método de diferencias finitas por defecto, o si se desea también está disponible FEM, para parcelar un dominio de integración en  $l \times m \times n$  elementos en los ejes  $x$ ,  $y$  y  $z$ , resolviendo el sistema de ecuaciones con teoría electrostática clásica, es decir considerando una descripción macroscópica del problema. Ahora bien, la gracia de este software es la capacidad de resolver la ecuación de Poisson-Boltzmann para una concentración, carga y radio de iones solicitados en el solvente. El programa realiza el cálculo energético de la molécula o macromolécula en dos situaciones: en el solvente, ya en su situación final (con o sin distribución de Boltzmann, según se requiera), y en un punto de referencia que corresponde a los átomos que conforman a la (macro)molécula en un dieléctrico correspondiente a la permitividad de dicha (macro)molécula; en definitiva, esto es calcular las energías debido a interacciones de Coulomb. Luego, las energías se restan, lo que corresponde a la energía de solvatación, siguiendo su



definición, del cuerpo bajo estudio.

APBS tiene gran fama de ser confiable y con una rapidez de cómputo relativamente alta, esto último en especial en comparación con programas explícitos (MD) como CHARMM o LAMMPS, pero debido al método de modelación que utiliza, puede ser problemático representar secciones de una molécula que tengan interacciones que requieran una alta refinación de la malla, como por ejemplo en el caso en que un ion pase por un canal iónico, pues las energías dependen de la distancia ion - proteína y la geometría de la proteína. Esto no quita que el programa sea altamente confiable, y es por eso y por la formulación en electrostática en continuos que emplea que se utiliza como referencia para comparar resultados.

Es importante explicar que, como se utiliza el método de diferencias finitas, se debe establecer un dominio de integración que luego se subdivide en puntos o nodos. Este dominio es finito y en las superficies límites se establecen condiciones de contorno. Es importante para este estudio ahondar un poco en cómo funcionan los tres tipos de condiciones que se pueden utilizar. En primer lugar, se puede imponer la condición que tiene el potencial al infinito, que es igual a cero - esta condición es llamada `zero`. Otra alternativa es ubicar una esfera con una carga puntual, un dipolo y un cuadrupolo, cuyo radio es el de la molécula modelada; la idea es que los momentos (de la carga puntual, dipolo y cuadrupolo) sean representativos de los de la macromolécula, y es con esta información que se evalúa el potencial en el contorno - se denomina condición `sdh`. La última opción que interesa en el contexto de este trabajo consiste en tomar los átomos componentes de la molécula y

formar esferas que no interactúan entre ellas, con las cargas y radios correspondientes, y evaluar el potencial en la frontera del dominio - recibe el nombre `mdh`. Todo esto se puede encontrar en el sitio dedicado al programa [2]

### 3.2.3 APBSmem

APBSmem es un programa de cálculos energéticos en sistemas proteína - membrana con teoría electrostática en continuos. Funciona de la misma forma que APBS (APBSmem es de hecho una GUI que solo llama a APBS con dos grupos de archivos distintos, uno pre y uno pos consideración de la membrana) y facilita el trabajo a usuarios con poco conocimiento computacional a través de una interfaz gráfica.

Este software escrito en `Java` funciona siguiendo una serie de pasos, en función de parámetros indicados por el usuario:

1. Generar mapas de permitividades  $\epsilon$ , radios de Debye o accesibilidad iónica  $\kappa$ , y densidad de carga  $\rho$  para una proteína en disolución.
2. Utilizar esos mapas con el programa APBS base para calcular la energía de solvatación de una proteína en disolución.
3. Generar un nuevo set de mapas de las propiedades en 1., pero esta vez incluyendo la membrana según parámetros indicados por el usuario (espesor y permitividad principalmente).
4. Correr nuevamente APBS para obtener la energía de solvatación de la proteína inserta en la membrana.
5. Restar las energías de 4. y 2., entregando esta resta para indicar la afinidad de la proteína con la membrana y el resto del entorno.

APBSmem entrega la energía total de la proteína embebida en la membrana menos la del soluto en solvente, que en caso de ser un estado de mínima energía constituye el estado más probable del sistema. En resumen, si el resultado entregado por APBSmem es  $< 0$ , la proteína tiene más probabilidades de estar inserta en la membrana que en la solución salina original, y visceversa.

Para poder comparar los resultados presentados más adelante con los de APBSmem y APBS, es necesario sumar las energías de solvatación entregadas por estos últimos dos programas a modo de obtener la energía de solvatación del sistema membrana - proteína. Otro camino es modificar el archivo de entrada `.solv.in` para que calcule directamente lo mismo que PyGBe.

### 3.2.4 PyGBe

Python, GPUs and Boundary elements for biomolecular electrostatics (PyGBe) es una biblioteca de Python que permite, entre otras cosas, calcular energías de solvatación en macromoléculas. El programa utiliza un algoritmo de tipo treecode a modo de implementación del Fast Multipole Method (FMM) para desarrollar cálculos, y además si estos son muy demandantes, pueden llevarse a cabo en GPU en vez de en CPU. Este trabajo se realizó en una versión básica de PyGBe que no incluye el uso de GPUs ni del FMM. Sin embargo, el resultado es el mismo, pues las características de las que carece son métodos de aceleración y no modifican la solución. La versión de PyGBe utilizada forma y resuelve un sistema de ecuaciones mediante la generación de una matriz  $\underline{A}$ , y es esto lo que se modifica para incluir el aporte de membrana de forma implícita, como se explica más adelante.

PyGBe funciona leyendo una serie de archivos. Los inputs del programa son un `.config` y un `.param`. Además, el primero de estos dos archivos hace referencia a otro par de archivos, `.face` y `.vert`, así como a un `.pqr` o `.crd`. En primera instancia se profundiza en qué son estos archivos, y luego se presenta y desarrolla la estructura funcional de PyGBe (`pygbe_matrix` específicamente).

### Archivo `.config`

El archivo `.config` que sirve como entrada directa al programa consta de una serie de parámetros que atañen principalmente a las características físicas del modelo. Se trabaja en una estructura de campos, siendo cada uno representativo de uno de los dominios de integración. Siempre habrá un `FIELD` que representa el solvente exterior a la proteína, designado con el número 0, y los campos consecutivos listan las propiedades de los otros medios a considerar. Esto permite modelar casos en los que la proteína se plegó sobre sí misma dejando solvente encerrado en pequeños huecos o cavidades. Además este set de parámetros sirve de puntero para los archivos que entregan información sobre las cargas (`.pqr` o `.crd`) y sobre la geometría del problema (`.face` y `.vert`).

El archivo `.config` tiene en cada una de sus líneas, que son los campos (`FIELD`), información sobre si el dominio debe ser tratado mediante la ecuación de Poisson o la de Poisson - Boltzmann, si se requiere el cómputo de energías en el medio, cuál es la permitividad  $\epsilon$  de dicho medio y cuál es el inverso de la longitud de Debye. También se incluye información sobre si hay cargas en el medio y en qué archivo está la

data al respecto, qué campo(s) hay al interior de cierto(s) FIELD(s) y cuál(es) al exterior<sup>1</sup>, y cuáles son sus designaciones.

Los archivos `.face` y `.vert` son información sobre la superficie de la proteína. El primero de ambos documentos numera y ordena los paneles en los que se subdivide la interfaz molécula - solvente, mientras que el segundo entrega las posiciones espaciales de los vértices de dichos paneles triangulares. Los archivos `.pqr` y `.crd` son información sobre la ubicación, radios de van der Waals y carga neta de los átomos constituyentes de la molécula que se busca modelar.

#### Archivo `.param`

El archivo `.param` por otro lado tiene información propia del método de modelación matemática del problema, aunque solamente se describen a continuación los que se utilizan en `pygbe_matrix`. Aquí van la precisión computacional (`double` o `float`), el número de puntos de cuadratura Gaussiana para tres tipos de resolución que se explicarán más a fondo en lo sucesivo, un parámetro que indica cuándo utilizar el método de resolución semianalítica (también se explica más adelante), el error relativo de máquina, y parámetros para el método de mínimos residuos general, como número máximo de iteraciones, iteraciones para el *reinicio* del método, y la tolerancia entre iteraciones tras la cual se acepta el resultado calculado.

Ahora se procede a explicar el funcionamiento del programa, pues este trabajo consiste en la modificación del mismo para integrar la

---

<sup>1</sup>Por ejemplo, un único ion (que forma una esfera como superficie de v. d. Waals) tendría agua por fuera y nada por dentro.

capacidad de cálculo de interfaces por sobre y bajo la macromolécula.

### Forma de operación del programa

PyGBe funciona siguiendo una serie de pasos para el cálculo de energías. En primer lugar, el programa lee los archivos `.config` y `.param` y guarda los parámetros correspondientes. Con esta información y la geometría importada (existe una función que se encarga de eliminar los paneles con áreas muy pequeñas, tendiendo a 0), la meta es crear un sistema como 3.1.13 que se cumpla para todos los paneles de la discretización. Se generan entonces dos sets de arreglos que corresponden a la clases definida como *superficie*, uno que guarda las fuentes (`src`) y otro que guarda los objetivos (`tar`). El programa calcula las interacciones electrostáticas entre todos los objetivos y todas las fuentes: para un panel arbitrario, todos los paneles del ensamblaje tienen influencia, incluyéndose a si mismo. Para generar la matriz del sistema, se requieren los single y double layer operators de cada panel para la ecuación por fuera y por dentro, y como son operadores integrales, se utiliza cuadratura Gaussiana para discretizarlos. Las denominadas fuentes son los puntos de cuadratura de Gauss, que representan el efecto del panel. Esto es, si los paneles objetivo están más allá de cierto rango con respecto a las fuentes. La información sobre qué operador se debe utilizar (Laplace o Yukawa) para cada `FIELD` viene del archivo `.config`, así como los parámetros físicos requeridos en cada ecuación ( $\kappa$  y  $\varepsilon$ , por ejemplo).

Para lograr esta discretización y dado un panel arbitrario, los paneles lejanos se trabajan con pocos puntos de cuadratura, en vista de que su distancia hasta el panel de referencia es grande respecto del parámetro

*threshold*. Para los paneles que están a una distancia inferior a este valor, pero aún no están suficientemente cerca, se utiliza simplemente una mayor densidad de puntos de cuadratura Gaussiana sobre la superficie y se procede a discretizar las integrales de la misma forma. Por último, para los casos en que los paneles objetivo están muy cerca de las fuentes, o el panel fuente y el objetivo son el mismo, la cuadratura Gaussiana se reemplaza por un método semianalítico que reduce la integral de área a una integral de línea, siendo esta última la que se discretiza, y solo sobre los bordes del triángulo; este método surge de la publicación de Hess y Smith [16] para el caso de Laplace, y Zhu *et al.* [25] para el caso de Yukawa, y se explica a fondo en la sección 3.3.2.

Con los valores de los operadores ya calculados, se procede a generar la matriz  $\underline{A}$  del sistema, ubicando cada elemento  $V_L$ ,  $K_L$ ,  $V_Y$  y/o  $K_Y$  donde corresponda<sup>2</sup>. El ordenamiento de la matriz incide en el lado derecho de la ecuación, por lo que también hay que organizar la ubicación de los elementos del vector solución para que corresponda al sistema físico que se busca modelar. Luego de generar un preconditionador para el sistema de ecuaciones, el código utiliza el método Generalized Minimal Residues (GMRES) para calcular el vector de incógnitas que tiene los valores de  $\phi$  y  $\partial/\partial\hat{n}(\phi)$  para todos los paneles. Este vector se le entrega a una función que toma los valores y calcula la energía de solvatación del sistema, aplicando una discretización con cuadratura Gaussiana al campo de reacción causado por la molécula sobre la superficie de esta última.

---

<sup>2</sup>Se escribe y/o porque es el sistema físico/real el que se compone de una proteína rodeada de una solución, pero se podría realizar cualquier combinación de casos, como por ejemplo agua por dentro y por fuera.

### 3.3 Implementación del método de imágenes en el programa PyGBe

El objetivo principal del trabajo aquí presentado es definir una forma de calcular la energía de solvatación de un átomo o molécula inserta entre dos interfaces, lo que tiene un simil biológico conocido en una proteína embebida en una membrana. Hay que recalcar que esta manera de representar las interfaces correspondientes a la membrana tiene dos restricciones importantes. En primer lugar, solo se puede aplicar a las cargas que se encuentran dentro de la membrana; para poder considerar proteínas transmembrana o similares hay que buscar otra forma de modelar. En segundo lugar, el modelo presentado y aplicado no permite incluir el efecto de iones en la disolución. Dicho de otra forma, se utiliza el método BEM para trabajar una ecuación de Poisson en vez de una ecuación de Poisson-Boltzmann. Además de dichas restricciones, surge otra que en vista del enfoque continuo que se da a la formulación, no incide mucho, pero aún así es importante mencionar: la bicapa fosfolipídica, tal como se muestra en el apartado 2.6, no es exactamente continua, sino que hay separación entre los fosfolípidos y, más importante, entre las cabezas polares de éstos. Sin perjuicio de lo anterior, la implementación exitosa del método de imágenes supone un avance, pues permite comprobar la viabilidad de esta parte de la teoría de campos electromagnéticos como representación de una membrana.

Como se adelanta en la sección de descripción de los programas utilizados, PyGBe es modificado, añadiendo una opción que permite calcular la energía de solvatación de una molécula en un sistema conformado por un medio 1 encerrado por un medio 2 (figura 3.3). Antes



de elaborar sobre la inclusión de esta opción al software, es imperativo ahondar en la teoría electrostática para este tipo de situaciones.

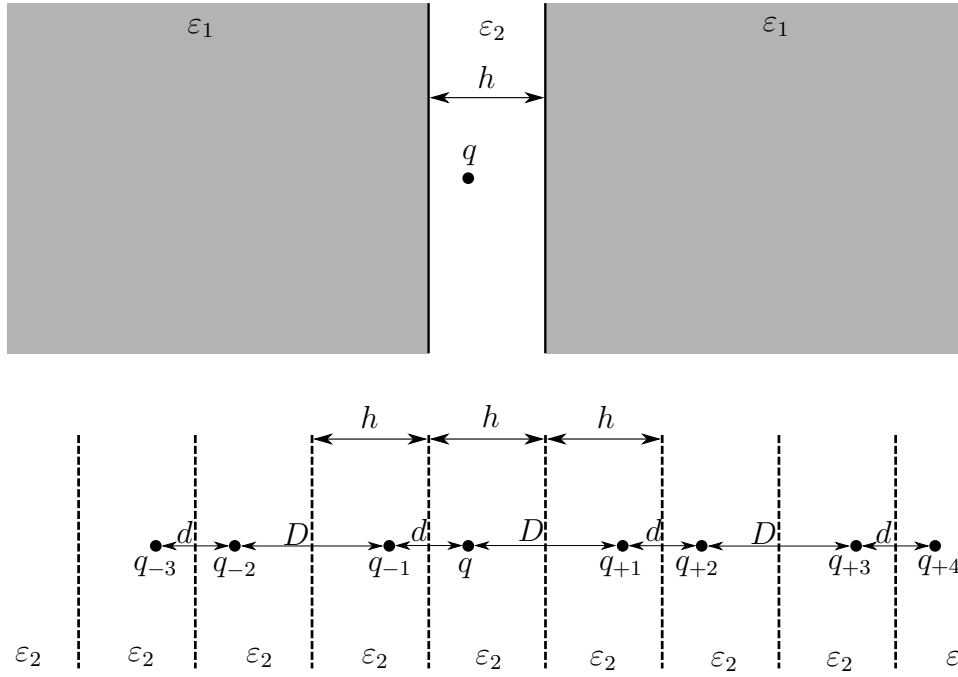
### 3.3.1 Modelo de un sistema de tres medios dieléctricos con dos interfaces

La presente sección se basa fundamentalmente en el desarrollo visto en la publicación de Yang, Liaw y Lim [23]. La teoría electrostática puede ser encontrada en el libro de Emile Durand [12], pero por lo que se sabe, el grupo de Yang es el único que ha utilizado esta teoría para representar una membrana con el método de imágenes.

En este paper se presenta una herramienta basada en el método de cargas imagen aplicado a sistemas rectangulares con más de una interfaz, denominado método RIC - Rectangular Image Charge. Un sistema de una interfaz planar infinita tiene una solución analítica para el caso de una carga puntual (sección 2.4.1), pero si se considera el caso de un sistema con dos interfaces planares paralelas y una carga puntual en el medio dieléctrico encerrado, el problema se complica.

Si se tienen tres medios dieléctricos 1, 2 y 3, tales que las constantes dieléctricas de los medios 1 y 3 son iguales entre sí  $\varepsilon_1 = \varepsilon_3$  (figura 3.3), entonces una carga en el medio 2 genera un campo de reacción en los medios 1 y 3, y cada uno de estos induce una reacción en los otros medios. Se genera así un efecto dominó, lo que se traduce en una sumatoria infinita [23] como se presenta a continuación

$$\phi(\mathbf{r}) = \sum_{i=-\infty}^{\infty} \frac{q_i}{4\pi\varepsilon_2|\mathbf{r} - \mathbf{r}_i|} \quad (3.3.1)$$



**Figura 3.3:** La imagen superior es la situación real para el caso de una carga, mientras que la figura inferior es el sistema trabajado con el método de imágenes. Esto sigue de manera infinita, pero solo se muestran unas pocas cargas por motivos obvios.

donde

$$q_i = q \left( \frac{\epsilon_2 - \epsilon_1}{\epsilon_1 + \epsilon_2} \right)^{|i|} \quad x_i = (-1)^i x_0 + i \cdot a \quad \mathbf{r}_i = (x_i, y, z) \quad (3.3.2)$$

Cada término de la sumatoria es menor al anterior en magnitud, tendiendo a converger de forma relativamente rápida.

Una analogía de este fenómeno y que también tiene que ver con electromagnetismo es lo que ocurre al mirar hacia uno de dos espejos paralelos enfrentados. La imagen de uno se refleja infinitamente en el otro, disminuyendo en tamaño con cada reflejo.

Considerar un número mayor de cargas ( $N_q$ ) simplemente implica sumar sobre estas, considerando el potencial debido a la interacción electrostática entre las partículas componentes del sistema, pues la sumatoria entre  $-\infty$  e  $\infty$  da cuenta del campo de reacción, haciendo que la ecuación (3.3.1) quede

$$\phi(\mathbf{r}) = \sum_{\substack{j=1 \\ j \neq i}}^{N_q} \frac{q_j}{4\pi\epsilon_2|\mathbf{r} - \mathbf{r}_j|} + \sum_{j=1}^{N_q} \sum_{\substack{i=-\infty \\ i \neq 0}}^{\infty} \frac{q_{j,i}}{4\pi\epsilon_2|\mathbf{r} - \mathbf{r}_{j,i}|} \quad ; \quad \mathbf{r} \neq \mathbf{r}_j \quad ; \quad \mathbf{r} \neq \mathbf{r}_{j,i} \quad (3.3.3)$$

En este caso,  $\mathbf{r}_{j,i}$  corresponde a la posición de la  $i$ -ésima imagen de la  $j$ -ésima carga del arreglo de cargas. Así también se define  $q_{j,i}$  como la  $i$ -ésima imagen de la  $j$ -ésima carga del arreglo. El primer término es el potencial de Coulomb en algún punto debido a las cargas. El segundo término corresponde al efecto del campo de reacción a causa del conjunto de cargas.

Lo que interesa es el segundo término, pues el primero ya ha sido implementado en el programa previamente. Con esto se puede modelar la reacción de los medios por fuera de las interfaces de la membrana o, mejor dicho, el comportamiento de estos debido a la presencia de las cargas de la proteína. Esto es importante, pues nunca se modela directamente una membrana, sino que queda todo implícito sobre la superficie de la proteína. Es más, el solvente correspondiente al agua se omite, quedando solo representado mediante la ecuación (3.3.1); la molécula esta ahora en un medio con las propiedades de la membrana, y es a través de la expresión para el campo de reacción que se incluye el efecto de las interfaces que por fuera tienen agua.

El método BEM exige, a través de sus operadores integrales Single y Double Layer Potential, que haya una función de Green que modele el comportamiento del potencial de una carga (por ende la carga  $q$  debe ser unitaria), así como la derivada de dicha función. Por lo tanto, a continuación se presentan estos dos operadores que se utilizan en la modelación de la membrana. Los operadores se denominan a continuación con el subíndice  $M$ , por ser propios de un sistema con membrana.

$$V_M(\phi(\mathbf{r})) = \int_{\partial\Omega} \phi(\mathbf{r}) \frac{1}{4\pi} \sum_{i=-\infty}^{\infty} \frac{q_i}{|\mathbf{r} - \mathbf{r}_i^*|} d\partial\Omega \quad (3.3.4)$$

$$K_M(\phi(\mathbf{r})) = \int_{\partial\Omega} \phi(\mathbf{r}) \frac{1}{4\pi} \sum_{i=-\infty}^{\infty} q_i \frac{\partial}{\partial \hat{\mathbf{n}}} \left( \frac{1}{|\mathbf{r} - \mathbf{r}_i^*|} \right) d\partial\Omega \quad (3.3.5)$$

Los términos  $q_i$  y  $\mathbf{r}_i^*$  retienen las definiciones de 3.3.2, y tanto  $V_M$  como  $K_M$  siguen siendo válidos para cada panel, tal como sucedía con los operadores de la sección 3.1.2. Se puede apreciar que la complejidad no está en la formulación, puesto que los operadores del método de imágenes son simplemente una suma lineal de operadores Single y Double Layer Potential de Laplace. El obstáculo reside en la sumatoria infinita, que demanda muchos recursos computacionales, pues como se explica en los siguientes párrafos, cada punto de cuadratura tiene  $n$  imágenes por arriba y por abajo, así como la imagen  $i = 0$ , lo que da un total de  $2n + 1$  imágenes.

### 3.3.2 El método BEM y PyGBe con inclusión del efecto de la membrana

El método de imágenes presentado permite dar cuenta del efecto de dos interfaces de forma implícita, y esa es su gran fortaleza, pues no se

requiere una discretización de las interfaces membrana - solvente. En vez de eso, se reproduce el efecto de los campos de reacción debido a los paneles de la proteína al modificar las funciones de Green de los operadores Single y Double Layer Potential. Ahora se ve más detalladamente cómo se trabaja este tipo de sistema de forma numérica.

Se debe considerar que el método de imágenes como se aplica en este caso considera que la molécula a modelar siempre está encerrada por dos interfaces paralelas ubicadas en planos  $z = \pm z_0$ . Con esto claro, se procede a explicar cómo se trabaja matemáticamente con BEM.

Se tienen las ecuaciones por fuera y por dentro de la interfaz de la molécula y dentro de la membrana; la de afuera lleva implícitamente el aporte del campo de reacción del solvente. El sistema de ecuaciones para una molécula embebida en una membrana de espesor  $a$ , permitividades de membrana  $\varepsilon_m$ , de proteína  $\varepsilon_p$  y de solvente (agua)  $\varepsilon_w$  es

$$\begin{aligned} \frac{\phi_1(\mathbf{r})}{2} + \int_{\partial\Omega} \frac{\partial}{\partial \hat{\mathbf{n}}} (\phi_1(\mathbf{r})) \cdot \sum_{i=-n}^n \frac{q_i}{4\pi |\mathbf{r} - \mathbf{r}_i^*|} d\partial\Omega \\ - \int_{\partial\Omega} \phi_1(\mathbf{r}) \cdot \sum_{i=-n}^n \frac{q_i}{4\pi} \nabla \left( \frac{1}{|\mathbf{r} - \mathbf{r}_i^*|} \right) \cdot \hat{\mathbf{n}} d\partial\Omega = 0 \end{aligned} \quad (3.3.6)$$

$$\begin{aligned} \frac{\phi_2(\mathbf{r})}{2} - \int_{\partial\Omega} \frac{\partial}{\partial \hat{\mathbf{n}}} (\phi_2(\mathbf{r})) \cdot \frac{1}{4\pi |\mathbf{r} - \mathbf{r}^*|} d\partial\Omega \\ + \int_{\partial\Omega} \phi_2(\mathbf{r}) \cdot \nabla \left( \frac{1}{4\pi |\mathbf{r} - \mathbf{r}^*|} \right) \cdot \hat{\mathbf{n}} d\partial\Omega = \frac{1}{4\pi \varepsilon_p} \sum_{j=1}^{N_q} \frac{q_j}{|\mathbf{r} - \mathbf{r}_j|} \end{aligned} \quad (3.3.7)$$

reteniendo su definición los términos 3.3.2 y definiendo el radio  $\mathbf{r}_i^*$

$$q_i = q \left( \frac{\varepsilon_p - \varepsilon_m}{\varepsilon_p + \varepsilon_m} \right)^{|i|} \quad z_i = (-1)^i z_0 + i \cdot a \quad \mathbf{r}_i^* = (x, y, z_i)$$

Aplicando las condiciones de contorno de electrostática - continuidad del potencial y el salto en la componente normal del campo eléctrico (ecuaciones (2.4.1, 2.4.2)) - el sistema de ecuaciones se puede llevar a uno con incógnitas en términos de un único  $\phi$  y su derivada  $\partial/\partial\hat{\mathbf{n}}(\phi)$ . Utilizando la notación de los operadores Single y Double Layer Potential para el método de imágenes (3.3.4, 3.3.5) y los de Laplace (3.1.9, 3.1.10), se obtiene la BIE para la situación estudiada

$$\frac{\phi_2(\mathbf{r})}{2} + K_L(\phi_2(\mathbf{r})) - V_L \left( \frac{\partial}{\partial\hat{\mathbf{n}}} \phi_2(\mathbf{r}) \right) = \frac{1}{4\pi\varepsilon_p} \sum_{j=1}^{N_q} \frac{q_j}{|\mathbf{r} - \mathbf{r}_j|} \quad (3.3.8)$$

$$\frac{\phi_2(\mathbf{r})}{2} - K_M(\phi_2(\mathbf{r})) + \frac{\varepsilon_2}{\varepsilon_1} V_M \left( \frac{\partial}{\partial\hat{\mathbf{n}}} \phi_2(\mathbf{r}) \right) = 0 \quad (3.3.9)$$

Estas ecuaciones están siendo aplicadas sobre la superficie de la molécula, en este trabajo de carácter Solvent Excluded. Para resolver de forma numérica, la SES se subdivide en  $N$  paneles triangulares y las ecuaciones 3.3.8 y 3.3.9 son colocadas en el centroide de cada panel. Las variables dependientes desconocidas están dentro de la integral, por lo que para resolver el sistema se asume que  $\phi(\mathbf{r})$  y  $\partial_{\hat{\mathbf{n}}}\phi(\mathbf{r})$  son constantes, permitiendo sacarlas de las integrales correspondientes; como toda la formulación se realiza sobre la interfaz, se necesita calcular la interacción de la superficie de la molécula consigo misma<sup>3</sup>. Omitiendo el subíndice 2, para un panel de referencia  $v$ , la interacción con cada uno de los  $N - 1$  paneles restantes (aquí con subíndice  $w$ ), y consigo mismo

<sup>3</sup>En casos en que haya más de una superficie, es necesario calcular las combinaciones entre todas ellas.

se muestra a continuación

$$\begin{aligned} & \frac{\phi(\mathbf{r}_v)}{2} + \phi(\mathbf{r}_v) \sum_{w=0}^N \int_{\partial\Omega_v} \left( \frac{\partial}{\partial \hat{\mathbf{n}}} G_L(\mathbf{r}_v, \mathbf{r}_w) \right) d\partial\Omega_v \\ & - \left( \frac{\partial}{\partial \hat{\mathbf{n}}} \phi(\mathbf{r}_v) \right) \sum_{w=0}^N \int_{\partial\Omega_v} G_L(\mathbf{r}_v, \mathbf{r}_w) d\partial\Omega_v = \frac{1}{4\pi\epsilon_p} \sum_{j=1}^{N_q} \frac{q_j}{|\mathbf{r}_v - \mathbf{r}_j|} \end{aligned} \quad (3.3.10)$$

$$\begin{aligned} & \frac{\phi(\mathbf{r}_v)}{2} + \phi(\mathbf{r}_v) \sum_{w=0}^N \int_{\partial\Omega_v} \left( \frac{\partial}{\partial \hat{\mathbf{n}}} G_R(\mathbf{r}_v, \mathbf{r}_w) \right) d\partial\Omega_v \\ & - \left( \frac{\partial}{\partial \hat{\mathbf{n}}} \phi(\mathbf{r}_v) \right) \sum_{w=0}^N \int_{\partial\Omega_v} G_R(\mathbf{r}_v, \mathbf{r}_w) d\partial\Omega_v = 0 \end{aligned} \quad (3.3.11)$$

Como este par de ecuaciones resuelve el sistema para un panel, y la superficie discretizada se compone de  $N$  paneles, la forma matricial del sistema de ecuaciones que rige para todo el borde queda

$$\begin{bmatrix} \frac{\mathbb{I}}{2} + K_L & -V_L \\ \frac{\mathbb{I}}{2} - K_M & V_M \end{bmatrix} \begin{bmatrix} \phi \\ \frac{\partial}{\partial \hat{\mathbf{n}}} \phi \end{bmatrix} = \begin{bmatrix} \frac{1}{4\pi\epsilon_p} \sum_{j=1}^{N_q} \frac{q_j}{|\mathbf{r} - \mathbf{r}_j|} \\ 0 \end{bmatrix} \quad (3.3.12)$$

El cálculo de los Single y Double Layer Potentials requiere a su vez la discretización del panel para el cómputo numérico de las integrales, lo que supone en este caso distribuir  $k_G$  puntos de Gauss sobre un panel para calcular los valores de los integrandos en dichos puntos;  $k_G$  varía según si se consideran paneles lejanos, cercanos, o cuando corresponde usar un método semianalítico. Como ejemplo, para los operadores que consideran la ecuación de Laplace por fuera de la molécula, se tiene

$$V_{v,w}^M = A_v \cdot \sum_{k=1}^{k_G} \sum_{i=-n}^n \frac{\omega_k \cdot q_i}{4\pi |\mathbf{r}_w - \mathbf{r}_{v,k,i}^*|} \quad (3.3.13)$$

$$\begin{aligned} K_{v,w}^M &= A_v \cdot \sum_{k=1}^{k_G} \sum_{i=-n}^n \nabla \left( \frac{\omega_k \cdot q_i}{4\pi |\mathbf{r}_w - \mathbf{r}_{v,k,i}^*|} \right) \cdot \hat{\mathbf{n}} \\ &= A_v \cdot \sum_{k=1}^{k_G} \sum_{i=-n}^n \frac{\omega_k \cdot q_i}{4\pi} \cdot \nabla \left( \frac{1}{|\mathbf{r}_w - \mathbf{r}_{v,k,i}^*|} \right) \cdot \hat{\mathbf{n}} \end{aligned} \quad (3.3.14)$$

Aquí,  $\omega_k$  son los pesos de la cuadratura Gaussiana,  $q_i$  mantiene su definición según 3.3.2 con carga  $q = 1$ ,  $\mathbf{r}_{v,k,i}^*$  es la posición de la  $i$ -ésima imagen del  $k$ -ésimo punto de cuadratura del panel  $v$ , y  $A_v$  es el área de dicho panel.

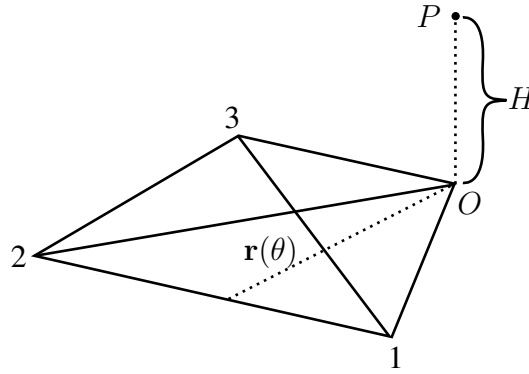
Lo anterior aplica para el panel  $v$  como `src` y  $w$  como `tar` (es decir, se esta calculando el potencial debido a  $v$  en el punto de colocación del panel  $w$ ), por lo que cada término en la matriz del sistema 3.3.12 tiene dimensiones de la cantidad de paneles objetivo  $\times$  cantidad de paneles fuente, y el sistema completo tiene una cantidad de  $2 \times (\text{paneles objetivo} \times \text{paneles fuente})$  elementos. El método recién explicado funciona bien cuando los paneles objetivo están a una cierta distancia del panel fuente; en general, estando a más de  $10^{-10}[\text{\AA}]$ , no se tienen problemas. Esto es, para paneles *lejanos* se utilizan pocos puntos de cuadratura, para paneles bajo cierto umbral pero que aún no están bajo los mencionados  $10^{-10}[\text{\AA}]$ , simplemente se aumenta la densidad de puntos.

Distinto es el caso del panel propio, que requiere la mencionada



integración semianalítica, pues las distancias serían muy pequeñas entre los puntos de cuadratura y el centro del panel (distancia  $\text{tar} - \text{src}$ ), y el denominador de las funciones de Green se volvería prácticamente cero. Esta forma de integración se utiliza, en el caso del método de imágenes, para el término 0 del contador, pues todos los demás están al menos a una distancia  $(a - z_0)$  con respecto al punto de cuadratura;  $z_0$  es la tercera coordenada del punto de cuadratura.

El método que se utiliza para integrar de forma semianalítica se inicia tomando el punto de colocación y proyectándolo sobre el plano del panel [25]. Esto permite formar tres triángulos ( $\Delta_{O12}$ ,  $\Delta_{O23}$ ,  $\Delta_{O13}$ ). Al trabajar una función que se quiera integrar sobre este grupo de triángulos en coordenadas polares, se puede integrar analíticamente en la variable  $r$ , de tal forma que solo es necesario variar el ángulo de integración para recorrer cada uno de los tres lados de cada triángulo del conjunto, como se muestra a continuación en la figura 3.4.



**Figura 3.4:** Aplicación del método semianalítico sobre un panel triangular.

La integral de la función ( $f(r, \theta)$  en la ecuación (3.3.15)) sobre el panel es entonces la suma de las integrales por separado sobre cada triángulo, de tal forma que para el caso ejemplo de la figura 3.4 y siendo consistente

con el sentido en que se recorren los triángulos, se tiene

$$\begin{aligned} \iint_{\Delta_{123}} f(r, \theta) dA &= \int_{\Delta_{123}} f(\theta) d\theta \\ &= \int_{\Delta_{O12}} f(\theta) d\theta + \int_{\Delta_{O23}} f(\theta) d\theta - \int_{\Delta_{O13}} f(\theta) d\theta \end{aligned} \quad (3.3.15)$$

La magnitud del radio es entonces la suma  $H + |\mathbf{r}_\theta|$ , donde  $H$  es la distancia normal entre el punto de colocación y el plano de proyección, y  $\mathbf{r}_\theta$  es la distancia entre dicha proyección y el lado del triángulo que se está calculando. Las integrales de línea que quedan son discretizada con cuadratura de Gauss. Los operadores Single y Double Layer para el panel singular son, para el caso de Laplace, los siguientes ([16])

$$V_L = \sum_{\text{lados}} \sum_{k=1}^{K_G} \omega_k \cdot (|\mathbf{r}(\theta_k)| - |H|) \cdot \frac{\theta}{2} \quad (3.3.16)$$

$$K_L = \sum_{\text{lados}} \sum_{k=1}^{K_G} \omega_k \cdot \left( \frac{H}{|\mathbf{r}(\theta_k)|} \right) - \text{sign}(H) \cdot \frac{\theta}{2} \quad (3.3.17)$$

Se tienen así los valores, calculados numéricamente, de los Single y Double Layer Potentials en todos los paneles para una (macro)molécula embebida en una membrana con solvente fuera. El siguiente paso es proceder a formar la matriz del sistema de ecuaciones y proseguir con la secuencia descrita en la sección 3.2 para PyGBE.

La inclusión del método de imágenes para cálculos con membrana en PyGBE significó la modificación de una serie de archivos para permitir al programa leer las propiedades de la bicapa (espesor y permitividad), seleccionar el tipo de cálculo que se requiere (PB o método de imágenes), y el número de imágenes en el método, entre otros. Se

procede por tanto a presentar los resultados del método y lo relativo a la labor de cómputo realizada, pero previo a esto se introduce una herramienta que facilita de gran forma el trabajo de entendimiento y comparación de los programas.

### 3.3.3 El ciclo termodinámico en solvatación

En el cálculo de energías de solvatación es muy útil analizar los estados en que se encuentra el sistema en cuestión. Para esto se realiza un ciclo termodinámico, que facilita el entendimiento de lo que se está calculando. En este trabajo se utiliza principalmente para entender justamente qué son los resultados que se están obteniendo, y compararlos con el programa APBSmem.

Para el caso trabajado, el ciclo termodinámico común para APBSmem y PyGBe es el que se muestra en la figura 3.5 y que consiste de cuatro estados. El estado ① es la membrana y el conjunto de cargas que componen la molécula, por separado, en el vacío<sup>4</sup>; como la membrana es un dieléctrico que no posee cargas explícitas, no hay una energía electrostática asociada, y para el caso del conjunto de cargas, la energía es simplemente la de Coulomb en el medio (ecuación (2.1.5)). El estado ② corresponde a la membrana y el conjunto de cargas, nuevamente por separado, pero esta vez en agua en vez de en el vacío. La proteína y la membrana mantienen sus permitividades, por lo que en este caso hay una condición de salto en el borde de la proteína (ecuación (2.4.1)). El estado ③ es un sistema en que la molécula está embebida en la membrana, y este conjunto está a su vez en el vacío; las permitividades

---

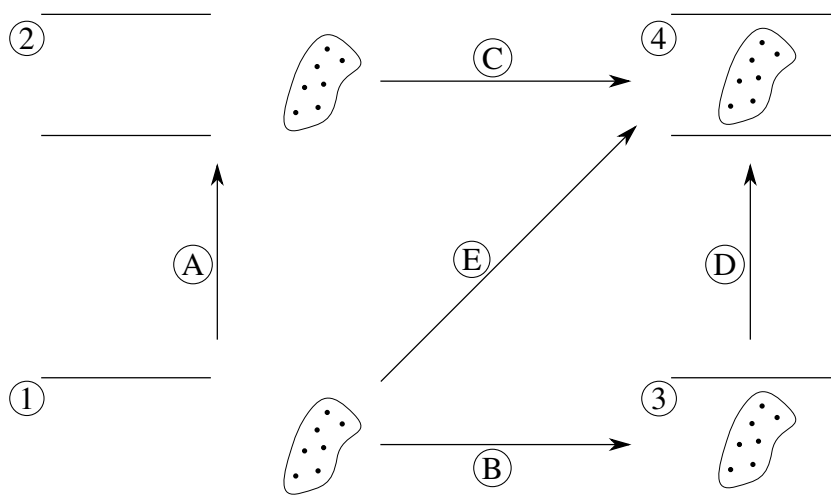
<sup>4</sup>Para efectos de ambos programas, el “vacío” es en realidad un medio dieléctrico polarizable de permitividad igual a la de la molécula.

de este último y la de la molécula siguen siendo iguales entre sí. Finalmente, el estado ④ es semejante al estado ③, pero por fuera del conjunto membrana-proteína hay agua con su respectiva permitividad. Las diferencias energéticas de la ① a la ⑤ se detallan a continuación.

**Tabla 3.1:** Expresión de las diferencias energéticas, en función de los estados 1 a 4.  $\Delta G$  es la diferencia en la energía de Gibbs entre los distintos estados del ciclo 3.5.

①	$\Delta G_{②-①}$
②	$\Delta G_{③-①}$
③	$\Delta G_{④-②}$
④	$\Delta G_{④-③}$
⑤	$\Delta G_{④-①}$

Se debe resaltar que, como en cualquier ciclo termodinámico, un camino cerrado suma cero si se es consistente. Esta propiedad es sumamente importante, pues PyGBe solo puede calcular directamente diferencias de energías que involucren el estado ① en la figura 3.5; gracias a que un camino cerrado da cero, se pueden calcular las diferencias de energía faltantes - en este caso ③ y ④ - a partir del resto del ciclo.



**Figura 3.5:** Ciclo termodinámico estudiado según lo presentado en los párrafos anteriores. Lo que se calcula con PyGBe y APBS es la diferencia  $1 \rightarrow 4$ .

---

## Capítulo 4

### Resultados

En primer lugar se recuerda que la meta de todo este trabajo es simular energías de solvatación de proteínas embebidas en membranas, analizando tres casos simples y contrastando resultados con un programa ampliamente aceptado y usado (APBS y APBSmem), tanto en términos de precisión del cálculo como en tiempo de cómputo requerido. Los tres casos simples son un ion de carga unitaria, un arreglo de tres iones, también individualmente de carga unitaria, uno de ellos de radio tal que encierra a los otros, y el tercer caso es una molécula de bromoetano (Apéndice A).

#### 4.1 Cálculo numérico de las energías

Las pruebas se realizaron con una membrana de espesor  $a = 20 \text{ \AA}$ , con una de las dos interfaces ubicada en el plano  $z = -10 \text{ \AA}$  respecto del origen  $(0, 0, 0)$  en coordenadas cartesianas, y la otra en  $z = 10 \text{ \AA}$ .

Las permitividades están adimensionalizadas, su valor siendo el descrito en el apartado 2.2.2. Los subíndices son, respectivamente de izquierda a derecha, indicadores de *membrane* (*membrana*), *protein* (*proteína*) y *water* (*agua*).

Los cálculos se realizan para tres densidades de malla, para los

tres casos de prueba, y para los dos programas - PyGBe y APBSmem. Los cálculos realizados corresponden a la diferencia ④ - ① en el ciclo termodinámico 3.5, con las siguientes propiedades

**Tabla 4.1:** Permitividades en los estados de interés.

Estado	$\varepsilon_m$	$\varepsilon_p$	$\varepsilon_w$
①	-	4.0	4.0
④	2.0	4.0	80.0

#### 4.1.1 Resultados mediante PyGBe

Para calcular las energías de solvatación con PyGBe, se utiliza MSMS para generar las mallas en archivos de formato `.vert` y `.face`, como se explica en la sección 3.2. Se crean tres mallas por caso de estudio; la densidad de elementos se aumenta a una tasa constante para poder realizar un análisis mediante extrapolación de Richardson, como se presenta más adelante. Los casos estudiados son `centered_ion`, `3q` y `bromoethane`. El primer caso es un único ion en (0,0,0) de radio 4 Å, el segundo son 3 iones, uno de radio 4 Å en (0,0,0) y los otros de radio 1 Å ubicados completamente dentro del volumen del primer ion, y el tercer caso es una molécula de 8 átomos con diversos radios y cargas. Las densidades de elementos de borde utilizadas, en  $\frac{vert}{\text{\AA}^2}$ , son 2, 4 y 8.

Para calcular los siguientes resultados se utilizaron 3 puntos de cuadratura para elementos lejanos, 37 en los paneles cercanos, y 9 puntos para el caso semianalítico. Además, el número de imágenes se mantuvo constante en  $N = 41$  para estas pruebas; más adelante se presenta un análisis en función de variar el parámetro  $N$  para ver cómo es la convergencia. Para el caso del bromoetano, como la molécula tiene una superficie menor (y por ende tiene menos paneles), se decide

aumentar el número de imágenes a  $N = 61$ . Esto para mejorar la precisión y porque el consumo de memoria lo permite. Sobre el primer caso, `centered_ion`, se presentan a continuación las energías de solvatación ( $E_{\text{solv}}$ ) obtenidas para las 3 mallas, junto con los tiempos de cálculo totales y la memoria utilizada. Se puede ver que la convergencia es proporcional a la de PyGBe sin membrana.

**Tabla 4.2:** Energías de solvatación calculadas, tiempos de cálculo y memoria RAM utilizada para los cálculos según la densidad de malla probada, para el archivo `centered_ion`.

Densidad	Número de elementos	$E_{\text{solv}}$ (PyGBe)	$E_{\text{solv}}$ (PyGBe + Membrana)	Tiempo de cálculo	RAM utilizada
vert./ $\text{\AA}^2$	[-]	kcal/mol	kcal/mol	[s]	[GB]
d02	736	-9.919124	5.258441	45.146	3.6
d04	1572	-9.886872	5.243907	176.890	15.9
d08	3124	-9.872508	5.236550	696.899	63.9

Para el segundo caso, `3q`, las energías son mayores, por lo que como se ve más adelante, el error relativo es bastante menor que en el caso anterior (`centered_ion`) y el que se ve después (`bromoethane`). A continuación se presenta una tabla equivalente a la anterior, pero en este caso solo se presentan las energías, pues los tiempos y la memoria consumidos varían muy poco ya que la discretización de la superficie es la misma que en el caso anterior.

Finalmente, para la molécula de bromoetano se presenta nuevamente una tabla como la 4.2, pues la geometría en este caso es distinta a una esfera y se debió realizar un mapeo diferente. La molécula tiene un área menor, lo que se traduce en un tiempo de cálculo y consumo de memoria menores para la misma densidad de elementos. Para este caso se omite la extrapolación de Richardson debido a que los resultados de los cálculos no están en sucesión creciente o decreciente, por lo que dicho método de extrap-

**Tabla 4.3:** Energías de solvatación calculadas según la densidad de malla para el archivo 3q.

Densidad	Número de elementos	E_solv (PyGBe)	E_solv (PyGBe + Membrana)
vert./Å <sup>2</sup>	[-]	kcal/mol	kcal/mol
d02	736	-90.703972	47.973011
d04	1572	-90.391445	47.841024
d08	3124	-90.252742	47.774469

olación no es realmente aplicable. La tabla de resultados en cuestión es la siguiente:

**Tabla 4.4:** Energías de solvatación calculadas, tiempos de cálculo y memoria RAM para cada una de las tres mallas de la molécula de bromoethane.

Densidad	Número de elementos	E_solv (PyGBe)	E_solv (PyGBe + Membrana)	Tiempo de cálculo	RAM utilizada
vert./Å <sup>2</sup>	[-]	kcal/mol	kcal/mol	[s]	[GB]
d02	322	-0.348698	0.218434	17.852	1.0
d04	694	-0.339504	0.221408	60.248	4.1
d08	1448	-0.339504	0.220199	224.459	19.2

#### 4.1.2 Resultados mediante APBS - APBSmem

Para poder validar los resultados obtenidos con PyGBe se requiere una referencia contra la cual comparar. El software APBSmem se utiliza para este propósito, en vista de su renombre. Ahora bien, el programa lo que hace es parcelar una “caja” (el dominio de integración) en nodos en los que calcula, dependiendo de la ubicación y condiciones de dicho nodo, la ecuación gobernante correspondiente. Por eso es que se requiere setear condiciones en los bordes de la mencionada caja. APBSmem, al igual que APBS, permite establecer dichas condiciones. Para el caso trabajado existen tres, que son explicados en el apartado 3.2.2. Se



presentan por ende tres tablas para el caso del archivo `centered_ion`, dos para el caso `3q` (se explica el por qué al presentar los resultados), y el valor de evaluación más fino para la molécula de `bromoethane`, en vista que se omite la extrapolación de Richardson para esta molécula.

Como en `APBSmem` hay que definir un dominio de integración rectangular, se opta por uno cúbico de  $128 \text{ \AA}$  de lado, con tres cantidades de elementos de modo que cada uno de estos sea de  $0.25 \text{ \AA}$ ,  $0.5 \text{ \AA}$  y  $1 \text{ \AA}$ ; de esta forma se mantiene una tasa de refinación de malla constante. Se utiliza `APBS` con mapas dieléctricos modificados para incluir la membrana, y el programa entrega la misma diferencia que `PyGBe`, pero en unidades de  $\text{kJ/mol}$ . Esto no requiere más que una división por  $4.184 \text{ kJ/kcal}$  para hacer los resultados directamente comparables. Siguiendo el mismo orden que para `PyGBe`, los resultados para el primer caso de estudio son los siguientes

**Tabla 4.5:** Resultados, tiempos de cálculo y RAM utilizada para el cálculo sobre el archivo `centered_ion`, mediante `APBS`.

Tipo de condición	$\Delta x$	Número de elementos	$E_{\text{solv}}$	$E_{\text{solv}}$	Tiempo de cálculo	RAM utilizada
	$\text{\AA}$	[-]	$\text{kJ/mol}$	$\text{kcal/mol}$	[s]	[GB]
zero	1.00	128	22.65401	5.41420	16.548	0.5
	0.50	256	22.46140	5.36840	131.126	3.8
	0.25	512	22.30540	5.33112	1208.873	29.7
sdh	1.00	128	20.39885	4.87544	16.872	0.5
	0.50	256	20.20733	4.82967	132.265	3.8
	0.25	512	20.05135	4.79239	1150.529	29.7
mdh	1.00	128	20.39885	4.87544	16.361	0.5
	0.50	256	20.20733	4.82967	134.597	3.8
	0.25	512	20.05135	4.79239	1226.290	30.8

A continuación se muestran los resultados para `3q`, y la razón de solo

presentar dos sets de resultados para esta molécula es que se repite el comportamiento apreciado en la tabla anterior, 4.5; las energías de solvatación para las condiciones *sdh* y *mdh* no varían, probablemente porque los casos de estudio son muy simples. Por esto es que únicamente se presentan los datos correspondientes a condiciones de contorno *zero* y *sdh*.

**Tabla 4.6:** Resultados, tiempos de cálculo y RAM utilizada para las tras mallas en el caso de estudio 3q, mediante APBS.

Tipo de condición	$\Delta x$	Número de elementos	E_solv	E_solv	Tiempo de cálculo	RAM utilizada
	Å	[-]	kJ/mol	kcal/mol	[s]	[GB]
zero	1.00	128	207.06438	49.48957	0.6	23.602
	0.50	256	204.90048	48.97239	4.0	137.004
	0.25	512	203.40408	48.61474	29.9	1174.478
sdh	1.00	128	186.77688	44.64074	0.6	23.958
	0.50	256	184.61382	44.12376	3.9	127.519
	0.25	512	183.11772	43.76619	29.9	1093.347

Para el caso de la molécula de bromoetano se presenta simplemente el valor más fino calculado en APBSmem; esto para condiciones *zero* y *sdh*.

**Tabla 4.7:** Resultado, tiempo de cálculo y RAM utilizada para la malla más fina en el caso de estudio de bromoethane, mediante APBS.

Tipo de condición	$\Delta x$	Número de elementos	E_solv	E_solv	Tiempo de cálculo	RAM utilizada
	Å	[-]	kJ/mol	kcal/mol	[s]	[GB]
zero	0.25	512	0.90307	0.21584	1194.226	30.8
sdh	0.25	512	0.90299	0.21582	1206.741	29.7

## 4.2 Extrapolación de Richardson y convergencia

Para poder comprobar de manera fehaciente que utilizar el método de imágenes acoplado a PyGBe es una forma válida de simular una proteína embebida en una membrana, se utiliza una extrapolación de Richardson para comparar los cálculos obtenidos con los dos programas. La extrapolación de Richardson es una herramienta que permite, como indica el nombre, realizar una extrapolación mediante tres puntos para obtener el valor teórico de convergencia si la malla se afinara consecutivamente a una tasa constante. Es por esta razón que todos los cálculos realizados previamente presentan un refinamiento de malla constante. La fórmula para obtener el resultado teórico extrapolado es

$$G_t = \frac{G_1 \cdot G_3 - (G_2)^2}{G_1 - 2G_2 + G_3} \quad (4.2.1)$$

Para esto se debe cumplir que  $d = \Delta x_3 : \Delta x_2 = \Delta x_2 : \Delta x_1$ , donde  $G_i = G(\Delta x_i)$ . Además de calcular el valor teórico extrapolado, se calcula la tasa de convergencia, que nos indica qué tan rápido converge el programa a dicha solución; mientras mayor el valor, mejor es la tasa de convergencia. La fórmula para esto es

$$P = \frac{\ln \left( \frac{G_3 - G_2}{G_2 - G_1} \right)}{\ln(d)} \quad (4.2.2)$$

Con este método aclarado es que se presentan los resultados extrapolados para los casos de estudio (`centered_ion` y `3q`), así como también una comparación porcentual entre los resultados extrapolados obtenidos mediante PyGBe y APBS – APBSmem. Para el caso de `bromoethane` se calcula una diferencia porcentual entre los resultados para las mal-

las más finas solamente. Como APBSmem tiene dos resultados distintos (tres en realidad, pero se utilizan solo `zero` y `sdh` por los motivos explicados previamente) dependiendo de las condiciones de contorno utilizadas, se presentan dos errores porcentuales; sin embargo, se adelanta que el menor error corresponde a la condición de borde `zero` en todos los casos estudiados. La razón principal que provoca esto, por lo que se infiere, se desarrolla en la sección de análisis de los resultados obtenidos. Los valores extrapolados calculados con la fórmula 4.2.1 - para el brometano el valor de la malla más fina - se presentan en la tabla 4.8

**Tabla 4.8:** Valores teóricos extrapolados mediante extrapolación de Richardson de los distintos casos de estudio.

	centered_ion	3q	bromoethane
PyGBe	5.22901	47.70677	0.22020
APBSmem zero	5.16772	47.81296	0.21584
APBSmem sdh	4.62880	42.96407	0.21582

También se muestran a continuación las tasas de convergencia según la fórmula 4.2.2; con esta información podemos ver la rapidez con la que los cálculos hechos convergen al resultado extrapolado, y sirve como herramienta también para discernir qué tan bueno es el modelo utilizado; mientras mayor el valor, más rápida es la convergencia.

**Tabla 4.9:** Convergencia al valor teórico extrapolado de los resultados, para PyGBe y las dos condiciones importantes de APBS.

	centered_ion	3q	bromoethane
PyGBe	0.98224	0.98778	-
APBSmem zero	0.29655	0.53213	-
APBSmem sdh	0.29617	0.53186	-

### 4.3 Análisis de Resultados

Teniendo los valores de la tabla 4.8, se pueden calcular las diferencias porcentuales entre los resultados para cada código. Como la idea es utilizar APBSmem para validar los valores obtenidos con PyGBe, se presentan las diferencias entre estos y los de las dos condiciones de contorno para APBS.

**Tabla 4.10:** Diferencia de PyGBe con inclusión de membrana, con las dos condiciones de contorno de APBS.

	centered_ion	3q	bromoethane
APBSmem zero	1.18597	0.22209	2.01983
APBSmem sdh	12.96685	11.03877	2.02894

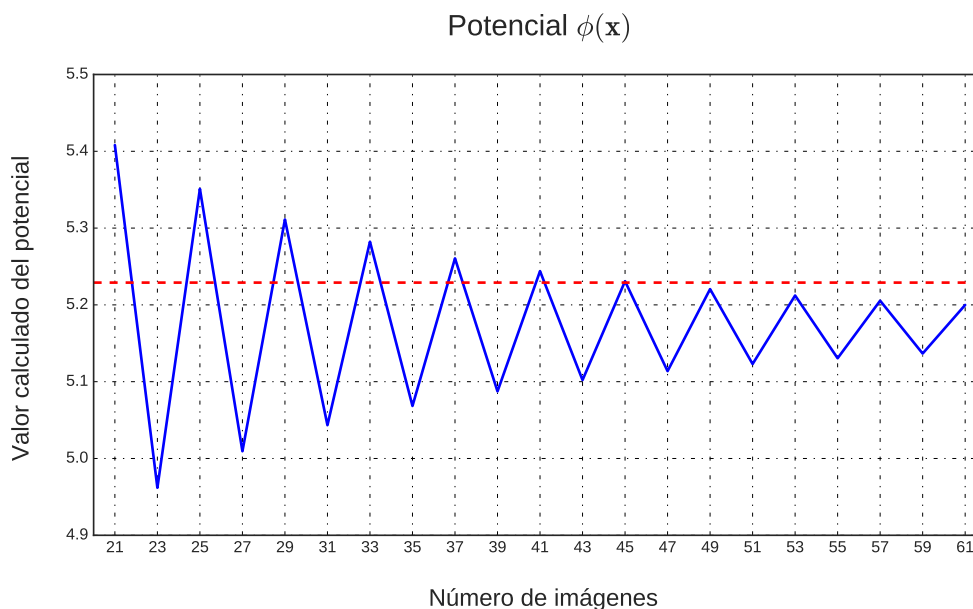
Como se adelantó en la sección anterior, las menores diferencias se obtienen cuando se fija una condición de borde tipo `zero` en APBSmem. Físicamente, la condición de contorno `sdh` debiese considerar el efecto del campo de reacción del medio más allá de las interfaces para ser válido, pues existe un salto causado por la membrana que tanto esta condición como `mdh` no toman en cuenta. Distinto es el caso de una condición `zero`, que solo requiere un dominio de integración grande, ya que independiente de la reacción del medio, el potencial en cualquier dirección tiende a 0 a medida que aumenta la distancia a las cargas que lo generan.

La idea es que el error respecto de APBSmem sea lo menor posible, por supuesto; de preferencia se apuntaba a un error menor a 2%, y se comprueba en la tabla 4.10 que esto se cumple en los dos primeros casos de estudio. Es difícil lograr un error menor en el caso del bromoetano, pues las energías de solvatación son muy pequeñas

comparadas con `centered_ion` y `3q`.

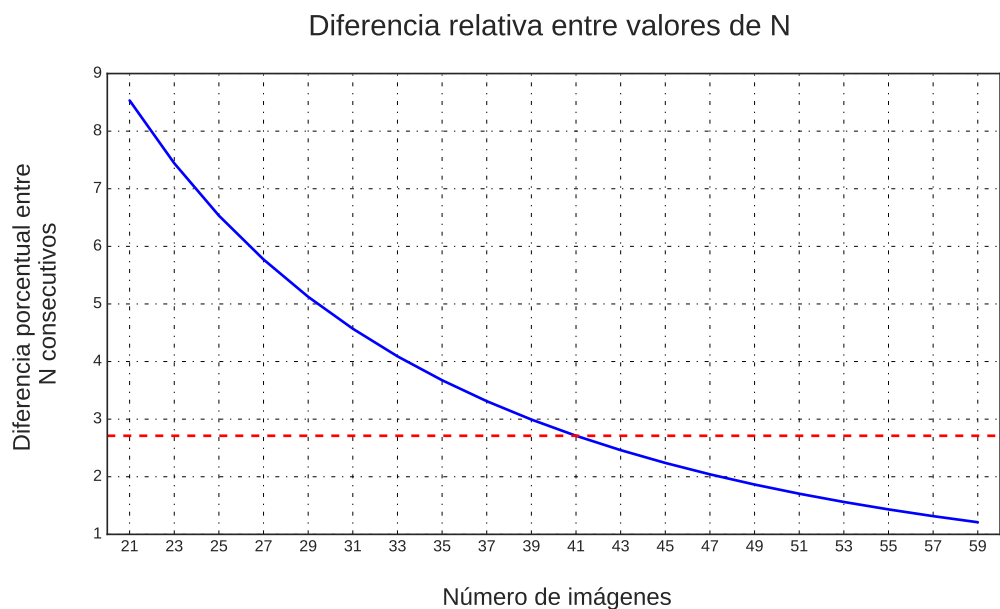
Con los resultados para  $N = 41$  cargas imagen ya tabulados, corresponde realizar un análisis de sensibilidad para ver qué tanto influye el número de imágenes y determinar un valor óptimo. También se muestra a partir del estudio de esta variable que  $N = 41$  es suficiente para el espectro que cubre este trabajo. Para esto se utiliza el caso del ion centrado con 1572 elementos de borde (d04), por razones de costo computacional principalmente. Se estudia el rango de la variable entre  $[21, 61]$ , yendo en aumentos de 2 (21, 23, 25...) y se presentan tanto las energías para el caso del ion centrado como las diferencias porcentuales entre distintos  $N$ . Se recuerda que, como ejemplo,  $N = 21$  significa que hay 10 imágenes por arriba ( $+z$ ), 10 por abajo ( $-z$ ), y el punto de cuadratura mismo (término  $i = 0$  en la sumatoria 3.3.1).

Viendo el gráfico 4.1, resulta evidente la naturaleza oscilatoria de la convergencia, lo que no es anormal en modelos de este tipo; el método de imágenes es, a fin de cuentas, una expansión cuyo signo va variando término a término. La línea horizontal roja indica el valor extrapolado mediante la fórmula de Richardson, y la idea es mostrar que este no difiere significativamente del valor al que tiende a converger el potencial. A pesar de lo bueno de este hecho, también se puede apreciar que el valor extrapolado fue trabajado con  $N = 41$  y es clara su cercanía justamente al potencial con ese número de cargas imagen. Por otro lado, se ve que dentro de todo la convergencia no es rápida, y para asegurar un resultado con un buen nivel de precisión se requieren muchos términos en la sumatoria. Del gráfico 4.2 se puede rescatar que, como se espera que suceda en una expansión, a medida que se agregan

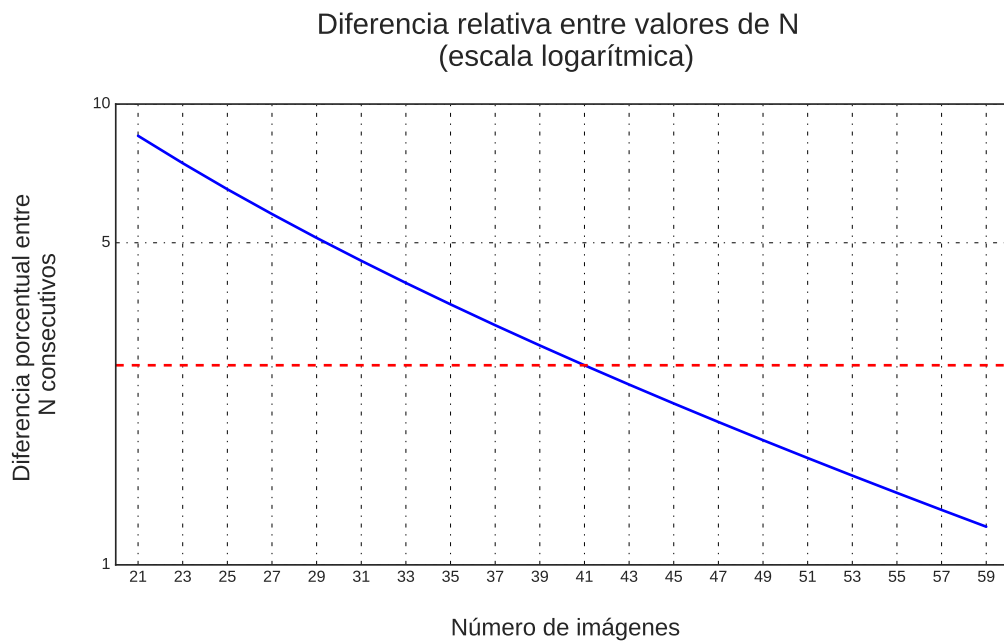


**Figura 4.1:** Resultados de PyGBE con membrana para el caso `centered_ion`, según se varía el número de cargas imagen del método. La línea roja es el resultado extrapolado para 41 cargas imagen.

términos, la variación entre los valores va disminuyendo; esto es claro a partir del gráfico de Potencial  $v/s$   $N$ , pero se hace más evidente en este caso. Para ver qué tipo de decaimiento tiene el error, se grafica dicho eje (el de las ordenadas) en escala logarítmica. Como se ve en el gráfico 4.3, la relación se torna lineal, por lo que el error disminuye con el logaritmo de  $N$ . Con esto queda definido que el método de imágenes aplicado a través de BEM sirve para modelar el potencial de una molécula, macromolécula o similar, embebida en una membrana con otro medio dieléctrico por fuera de las interfaces de esta última. El siguiente capítulo cierra las ideas desarrolladas en este escrito, y presenta además puntos que se considera importantes trabajar si es que se desea mejorar o continuar el hilo desarrollado en el presente texto.



**Figura 4.2:** Variación porcentual en el resultado, en función del número de imágenes empleado. La línea roja es el resultado extrapolado para 41 cargas imagen.



**Figura 4.3:** Variación porcentual, mostrada en escala logarítmica, de los resultados en función del número de cargas imagen.



---

## Capítulo 5

### Conclusiones

Para poder llevar a cabo este trabajo, se dedicó mucho tiempo al aprendizaje de la teoría y a la búsqueda de una forma de utilizar el método de imágenes para representar interfaces. Esto es de esperarse, pues lo que más esfuerzo toma aprender no es cómo utilizar una herramienta - en este caso `Python` y sus bibliotecas - sino qué es lo que se debe hacer con ella y, en caso de que falle, por qué hay problemas. Con esto mencionado, corresponde regresar a los objetivos planteados en la introducción de este escrito y preguntarse si efectivamente se lograron. En primer lugar, se logra averiguar sobre la forma de modelar interfaces paralelas mediante el método de imágenes, por lo que este punto se considera un éxito. En segundo lugar, dicho método es acoplado a `PyGBe` efectivamente, y cumple con errores porcentuales bajos, lo que lleva a tachar estos objetivos de la lista, pues se cumplen. En cuanto a tiempo de cálculo, la implementación hecha es bastante competitiva, como se ya se mencionó en la sección 4.3. Con todas estas metas cumplidas, la modificación hecha a `PyGBe` para incluir una forma básica de una “membrana” se puede considerar exitosa, pero tiene una falencia clave que consiste en el alto consumo de memoria que va ligado a  $N_i$ , el número de imágenes **por** elemento de borde. Sin embargo, la cercanía de los resultados a aquellos obtenidos con `APBSmem` da cuenta de la eficacia del método estudiado en este trabajo. Por lo tanto, y como conclusión, se puede afirmar que la prueba de concepto cumple con lo que se esperaba y de la

forma que se esperaba; los tiempos de cálculo para PyGBe mejorarán automáticamente si se logra reducir el número de imágenes necesarias para alcanzar resultados con buena precisión. También se rescata de este trabajo que, para casos en que se requieran dominios de integración muy grandes (esto en APBS - APBSmem), PyGBe es mucho más eficiente en cuanto a velocidad de cálculo, debido a que el programa de referencia debe generar los mapas pertinentes, luego modificarlos para incluir la membrana, y finalmente, después de todo esto, realizar los cálculos.

## 5.1 Continuación del trabajo

Se deben realizar una serie de mejoras y cambios a PyGBe en su estado actual para realmente hacerlo un competidor fuerte en el cálculo de energías de sistemas molécula - membrana. Estos cambios quedan como trabajos futuros, pero al menos en cierta forma parten de lo que se plasma en este informe. Primero esta lo más importante, que realmente lo haría funcionalmente interesante, y que corresponde a la capacidad de calcular energías para moléculas que tienen elementos dentro **y fuera** de la membrana: las proteínas transmembrana. Es decir, moléculas o macromoléculas que están parcialmente embebidas en la membrana, pero que tienen residuos que son hidrofílicos y por ende también hacen que esté en contacto con el solvente de forma directa (figura 2.4 por ejemplo).

Otra modificación que se vuelve atractiva es la inclusión del efecto de iones salinos a través de la ecuación de Poisson - Boltzmann por fuera de la membrana. Ya existen trabajos al respecto, como aquel por Lin, Xu, Tang y Cai [19], que formulan el método de elementos de borde

para una distribución de iones dada por la ecuación PB; la idea ahora es acoplar dicha formulación para incluirla en PyGBe, aprovechando las otras ventajas de este software.

Por último, una mejora evidente que se requiere es reducir la cantidad de memoria que se debe dedicar para estos cálculos mediante el método de imágenes, pues este resulta el principal obstáculo en este momento.

Otras formas de mejorar la precisión del modelo pueden venirse a la mente, pues aún queda mucho trabajo por delante, pero remitiéndose al marco de esta investigación, se logran los objetivos principales, se prueba de forma categórica que el método de imágenes sirve para cálculos de moléculas **en** membranas, y además se crea una pequeña base teórica para hacer más fluida la continuación de esta línea de investigación en el futuro, dado el caso.

---

## Apéndice A

### Archivo Bromoethane.pqr

ATOM	1	C1	TMP	1	0.013000	-0.725000	0.821000	-0.094000	1.908000
ATOM	2	C2	TMP	1	1.140000	-0.443000	1.786000	-0.005700	1.908000
ATOM	3	Br1	TMP	1	2.846000	-0.545000	0.859000	-0.181800	2.020000
ATOM	4	H1	TMP	1	-0.951000	-0.668000	1.337000	0.047100	1.487000
ATOM	5	H2	TMP	1	0.000000	-0.000000	0.000000	0.047100	1.487000
ATOM	6	H3	TMP	1	0.103000	-1.725000	0.384000	0.047100	1.487000
ATOM	7	H4	TMP	1	1.172000	-1.178000	2.594000	0.070000	1.387000
ATOM	8	H5	TMP	1	1.067000	0.563000	2.208000	0.070000	1.387000

Para una descripción de lo que significa cada columna, se puede revisar la referencia [3].

---

## Apéndice B

### Energía de solvatación

La energía de solvatación es la energía de un sistema solvente - soluto; se presenta una definición extraída de la referencia [21].

*Es el cambio en la energía de Gibbs cuando un ion o molécula se transfiere desde el vacío a un solvente. Las contribuciones principales a la energía de solvatación provienen de:*

- *La energía por la formación de la cavidad en la que se ubican las especies disueltas de soluto.*
- *La energía de orientación de los dipolos.*
- *La energía de interacción isotrópica, de origen electrostático y de dispersión.*
- *La energía anisotrópica por interacciones específicas, por ejemplo enlaces de hidrógeno.*

Para efectos de este trabajo, en vista de que se trata exclusivamente el aporte electrostático en sistemas, la energía de solvatación corresponde únicamente al reordenamiento dipolar y, por ende, al campo de reacción del solvente por la inserción de la molécula.

---

## Apéndice C

### Funcionamiento de APBSmem

APBSmem funciona por defecto a través de una interfaz gráfica, pero esto puede resultar limitante, pues el resultado entregado es una energía de solvatación entre estados fijos, y hasta donde se sabe, no hay forma de modificar esto por GUI. Afortunadamente se puede trabajar con APBSmem mediante terminal, lo que permite un mayor grado de flexibilidad a la hora de realizar cálculos.

Se presenta a continuación una estructura de trabajo que se puede aplicar para obtener los resultados de APBSmem utilizando solo una de sus rutinas acoplada al programa APBS base; esto está basado en un post en un foro oficial de sourceforge por parte de uno de los desarrolladores de APBSmem. También se explica paso a paso las acciones que realiza cada parte de este instructivo, al menos según el entendimiento del autor.

1. En primer lugar hay que correr APBSmem con cualquier configuración de parámetros (la que viene por defecto, por ejemplo). Para esto, lo único que se requiere es importar un archivo `.pqr` mediante la opción *Browse* en la parte superior de la interfaz gráfica, y luego correr la simulación (*Run*). Se pide especificar un directorio en el que escribir los archivos de salida<sup>1</sup>; esto crea una carpeta con varios documentos, de los cuales interesan los siguientes:

---

<sup>1</sup>Se debe tener cuidado, pues para simulaciones con una malla fina (y por ende densa), cada archivo sobrepasa fácilmente un GB de espacio

- `archivo.dummy.in`: Este archivo tiene un solo bloque, y se encarga de escribir mapas de permitividad ( $\varepsilon$ ), carga ( $q$ ), e inverso de la longitud de Debye ( $\kappa$ ) en formato `.dx`.
- `diel1.dx`: Es un set de ficheros que contienen la permitividad eléctrica en todos los puntos del dominio, **sin consideración de la membrana**. Se generan tres archivos (`dielx`, `diely`, `dielz`), pero hasta donde se sabe son los tres iguales, pues se considera que el tensor  $\varepsilon_{i,j}$  es diagonal e isotrópico.<sup>2</sup>
- `charge1.dx`: Es un archivo con la densidad de carga en todos los puntos del dominio.
- `kappa1.dx`: Es un archivo con los mapas de accesibilidad iónica en todos los puntos del dominio.
- `archivo.solv.in`: Este archivo tiene dos bloques que corresponden a los dos estados cuyas energías se desean calcular para obtener así la energía de solvatación a partir de la diferencia de las primeras. Este archivo está preparado para que APBS importe los tres sets de mapas mencionados **con** consideración de la membrana.

2. Con los archivos anteriores listos, se puede prescindir de la GUI de APBSmem. Lo que sigue es modificar el archivo `.dummy.in` para dejarlo con las propiedades físicas de interés. Como este archivo entrega un mapa sin membrana, lo que se ingresa son las propiedades del solvente y de la proteína sabiendo que después una parte del solvente será remplazada para acoplar la membrana. Es

---

<sup>2</sup>No se han hecho pruebas para comprobar otros casos, por lo que el hecho de que los tres archivos sean **siempre** iguales no es definitivo.

importante que el número de nodos en el dominio sea consistente entre este archivo y el `.solv.in`; de lo contrario, existe un conflicto al correr APBS por razones evidentes. Correr APBS sobre `.dummy.in` no realiza cálculos, solo sirve para escribir los mapas que se utilizan posteriormente. Para lograr esto, simplemente se debe correr por terminal el *ejecutable* (binario) `apbs` con archivo de entrada `archivo.dummy.in`, y los mapas se escribirán en el directorio desde el que se está ejecutando `apbs`<sup>3</sup>. Alternativamente, se puede definir el directorio de salida para el set de archivos como segundo argumento de `apbs`.

```
/apbs_bin_directory/apbs archivo.dummy.in
```

3. Una vez impresos los mapas, se debe llamar a la subrutina de APBSmem que genera un nuevo set de mapas con inclusión de la membrana. Para esto, se debe tener descomprimido APBSmem, además de tener JRE (Java Runtime Environment) en el equipo. Para el trabajo presente, este programa solo se utilizó para modificar la permitividad (la accesibilidad ya era 0 en todo el dominio) de los puntos correspondientes al interior de la membrana, pero hay una serie de argumentos válidos, como por ejemplo espesor de las cabezas hidrofílicas y su permitividad, entre otros. Para incluir la membrana, se debe estar operando en el directorio donde están mapas dieléctricos; el comando es el siguiente (no se deben considerar los saltos de línea, son por tema de espacio solamente. Debe ir un espacio en vez de un salto de línea cuando se ejecuta el programa):

---

<sup>3</sup>En el directorio desde el que está actuando la terminal, **NO** el directorio donde esta el binario de `apbs`



```
java -cp /apbsmem_directory/apbsmem.jar
apbsmem.DrawMembrane out-directory
z_m0 l_m pdie sdie mdie idie V
I R_m1 R_m0 a0 flood focus
```

$z\_m0$  corresponde a la coordenada sobre el eje  $Z$  en la cual se coloca la membrana. Hay que recordar que la membrana está conformada por el espacio entre dos interfaces planas infinitas, con vectores normales  $(0, 0, \pm 1)$ .

$l\_m$  es el espesor de dicha membrana, por lo que la interfaz superior se encuentra en  $z = z\_m0 + l\_m$ .

$(p, s, m, i)_{die}$  son la permitividad de la proteína, solvente, membrana, y cabezas fosfolipídicas, respectivamente. El programa busca todos los puntos que corresponden a solvente y los reemplaza por la permitividad de la membrana y  $\kappa = 0$  si es que están en el dominio designado para la membrana. Se desconoce cómo se incluye el efecto de las cabezas, pues siempre se las trabajó con espesor 0.  $V$  es el potencial de membrana **reducido**. La fórmula de potencial reducido está en la publicación correspondiente a APBSmem [9], pero para una temperatura de  $298.15\ K$ , se puede dividir el potencial en  $[mV]$  directamente por 25.69 para obtener el potencial reducido.

$I$  corresponde a la concentración de iones. Se considera una concentración simétrica de iones (misma de *ions* que de *counterions*).

$R\_m(1, 0)$  son los radios de exclusión superior e inferiores. No se utilizaron estas opciones durante este trabajo, por lo que se desconoce su implicancia en la escritura de los mapas modificados. Sin embargo, existe información sobre su aplicación en el artículo

asociado a APBSmem [9], en donde se indica que se utiliza para evitar que, en casos con solvente encerrado por la proteína de alguna forma, el programa sobrescriba esos puntos como membrana. `a0` especifica el espesor de las cabezas hidrofílicas de la membrana, en Å.

`flood` es un booleano, que define si se emplea el método flooding. Este método permite al programa identificar el solvente a remplazar como todo lo que este alrededor y por fuera de la proteína, evitando así sobrescribir cavidades y canales de paso como membrana.

`focus` corresponde al nivel de focusing que se utiliza para la generación del set de mapas. Hay más información al respecto en [2] sobre el método de focusing.

4. Finalmente, se llama nuevamente a APBS con los mapas creados en el ítem anterior, que incluyen la membrana, y se calcula la diferencia entre los estados de interés; por defecto estos son la proteína en la membrana y la proteína en el solvente.

---

## Bibliografía

- [1] <https://apbsmem.sourceforge.io>.
- [2] <https://apbs-pdb2pqr.readthedocs.io/en/latest/index.html>,.
- [3] <https://apbs-pdb2pqr.readthedocs.io/en/latest/formats/pqr.html>,.
- [4] D. Andelman. Chapter 12 - Electrostatic properties of membranes: The Poisson-Boltzmann Theory. In *Structure and Dynamics of Membranes*, volume 1 of *Handbook of Biological Physics*, pages 603–642. North-Holland, 1995.
- [5] N. A. Baker, D. Sept, M. J. Simpson, J. Holst, and J. A. McCammon. Electrostatics of nanosystems: Application to microtubules and the ribosome. *Proceedings of the National Academy of Sciences*, 98(18):10037–10041, 2001.
- [6] C. A. Brebbia and J. Dominguez. *Boundary Elements: An introductory course*. WIT Press, second edition, 1992.
- [7] W. Cai. *Computational methods for electromagnetic phenomena: Electrostatics in solvation, scattering, and electron transport*. Cambridge University Press, first edition, 2013.
- [8] W. Cai, S. Deng, and D. Jacobs. Extending the Fast Multipole Method to charges inside or outside a dielectric sphere. *Journal of Computational Physics*, 223(2):846–864, 2007.

- [9] K. M. Callenberg, O. P. Choudhary, G. L. de Forest, N. A. Gohara, D. W. and Baker, and M. Grabe. APBSmem: A graphical interface for electrostatic calculations at the membrane. *PLOS ONE*, 5(9): 1–12, 2010.
- [10] N. Campbell and J. Reece. *Biología*. Editorial Médica Panamericana, seventh edition, 2007.
- [11] C. D. Cooper. Biomolecular electrostatics with continuum models: A Boundary Integral implementation and applications to biosensors, 2015.
- [12] E. Durand. *Electrostatique, Vol. 3*. Elsevier Masson, 1966.
- [13] Alberts B. et al. *Molecular Biology of the Cell*. WW Norton and Co., sixth edition, 2015.
- [14] Z. Gan and Z. Xu. Multiple-image treatment of induced charges in Monte Carlo simulations of electrolytes near a spherical dielectric interface. *Physical Review. E, Statistical, Nonlinear, and Soft Matter Physics*, 84:016705, 2011.
- [15] D. J. Griffiths. *Introduction to Electrodynamics*. Prentice Hall, third edition, 1999.
- [16] J. L. Hess and A.M.O. Smith. Calculation of potential flow about arbitrary bodies. *Progress in Aerospace Sciences*, 8:1–138, 1967. ISSN 0376-0421.
- [17] J. D. Jackson. *Classical Electrodynamics*. Wiley, third edition, 1998.

- [18] L. Kantorovich. *Quantum Theory of the Solid State: An introduction*. Springer, first edition, 2004.
- [19] H. Lin, Z. Xu, H. Tang, and W. Cai. Image approximations to electrostatic potentials in layered electrolytes/dielectrics and an ion-channel model. *Journal of Scientific Computing*, 53(2):249–267, 2012.
- [20] J. Lombard. Once upon a time the cell membranes: 175 years of cell boundary research. *Biology Direct*, 9:32, 2014.
- [21] V. Minkin. Glossary of terms used in Theoretical Organic Chemistry. *Pure and Applied Chemistry*, 71(10):1919–1981, 2009.
- [22] Benoît Roux. Influence of the membrane potential on the free energy of an intrinsic protein. *Biophysical Journal*, 73:2980–2989, 1997.
- [23] P. Yang, S. Liaw, and C. Lim. Representing an infinite solvent system with a rectangular finite system using image charges. *The Journal of Physical Chemistry B*, 106(11):2973–2982, 2002.
- [24] B. J. Yoon and A. M. Lenhoff. A Boundary Element Method for molecular electrostatics with electrolyte effects. *Journal of Computational Chemistry*, 11(9):1080–1086, 1990.
- [25] Z. Zhu, J. Huang, B. Song, and J. White. Improving the robustness of a Surface Integral formulation for wideband impedance extraction of 3D structures. In *IEEE/ACM International Conference on Computer Aided Design. ICCAD 2001. IEEE/ACM Digest of Technical Papers (Cat. No.01CH37281)*, pages 592–597. IEEE, 2001.