



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
POSGRADO EN CIENCIA E INGENIERÍA DE LA COMPUTACIÓN

Modelo de Interacción Háptica para Aplicaciones en
Simulación Virtual de Procedimientos de Punción en
Medicina

T E S I S

QUE PARA OPTAR POR EL GRADO DE:
**MAESTRO EN CIENCIA E INGENIERÍA DE LA
COMPUTACIÓN**

PRESENTA:

HECTOR RICARDO RIVAS GUTIERREZ

Director de Tesis:

Dr. Miguel Ángel Padilla Castañeda
Instituto de Ciencias Aplicadas y Tecnologías

Ciudad Universitaria, Cd. Mx., enero de 2020



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

INSTITUTO DE INVESTIGACIONES EN MATEMÁTICAS APLICADAS Y SISTEMAS,
UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

Tesis para obtener el grado de Maestro en Ciencia e Ingeniería de la Computación

Todos los derechos reservados. No se permite la reproducción total o parcial de esta obra, ni su incorporación a un sistema informático, ni su transmisión en cualquier forma o por cualquier medio (electrónico, mecánico, fotocopia, grabación u otros) sin autorización previa y por escrito de los titulares de los derechos de autor. La infracción de dichos derechos puede constituir un delito contra la propiedad intelectual.

Primera Edición, 13 de enero de 2020

Agradecimientos

A mis padres, porque siempre me han apoyado en todas las decisiones que he tomado y me han motivado para seguir adelante.

Agradezco a mi familia por todo el apoyo que me ha brindado, que han sido y serán siempre una gran inspiración y motivación en mi vida.

Agradezco a mis compañeros y amigos con quienes compartí grandes momentos.

Agradezco a mi director de tesis y amigo, doctor Miguel Padilla, por ser un guía para el desarrollo de este trabajo.

Agradezco a la Secretaría de Educación, Ciencia, Tecnología e Innovación (SECTEI) por el apoyo recibido con folio 965c19.

Por último, agradezco al Consejo Nacional de Ciencia y Tecnología (CONACYT) por el apoyo económico para desarrollar mis estudios de maestría.

Resumen

El trabajo que se desarrolla en esta tesis consiste en un modelo de punción háptica que incorpora la suma de diferentes fuerzas para simular el procedimiento de punción. El modelo se puede adecuar a las propiedades mecánicas de los tejidos simulados a partir del ajuste de diferentes parámetros propuestos.

Para la implementación de este modelo se utilizó el framework de simulación SOFA. En este framework se incorporó el modelo propuesto a través del desarrollo de un plugin. Este plugin se puede incorporar a diferentes escenas para el desarrollo de simuladores en el práctica médica.

La aplicación del modelo propuesto se muestra a través del desarrollo de dos casos de estudio: un simulador virtual para el entrenamiento de la etapa de venopunción en el procedimiento de cateterización vascular periférica; y un simulador del procedimiento neuroquirúrgico de ventriculostomía. El sistema propuesto consiste en una simulación de un cuerpo deformable que proporciona retroalimentación de fuerzas durante el proceso de punción a través del dispositivo háptico Phantom Touch.

A través del dispositivo háptico se generan fuerzas de reacción que permiten al usuario experimentar la sensación háptica del procedimiento realizado a partir del modelo propuesto en este trabajo. La simulación de deformaciones de los tejidos se realizó utilizando el método de elemento finito. Además, el simulador propuesto proporciona al usuario información sobre el ángulo de inserción de la aguja, ya que éste es un valor de referencia importante en la práctica clínica.

El software desarrollado permite exportar la información sobre posición, trayectoria y fuerzas resultantes de la simulación para un posterior aná-

lisis, así como para la definición de métricas para la evaluación de practicantes.

Además, se realizó una evaluación del simulador de venopunción a través de experimentos realizados por estudiantes de medicina de la Universidad Nacional Autónoma de México.

Índice general

1. Introducción	1
1.1. Antecedentes	3
1.2. Objetivo	5
1.2.1. Objetivo General	5
1.2.2. Objetivos Particulares	5
1.3. Justificación	6
1.4. Estructura de la Tesis	9
2. Modelo Biomecánico	11
2.1. Esfuerzos y Deformaciones	11
2.1.1. Modelo de Esfuerzos	11
2.1.2. Modelo de Deformaciones	14
2.2. Modelo Elástico Lineal	16
2.3. Ecuaciones que Rigen el Modelo Elástico Lineal	17
2.3.1. Modelo Dinámico de Estructuras Elásticas	19
3. Interacción Háptica	21
3.1. Renderizado Háptico	21
3.2. Simulación de Punción Háptica	25
3.2.1. Modelo de Punción Háptica	25
3.2.2. Modelo de Fricción	28
3.3. Dispositivo Háptico Phantom Touch	30
4. Framework de Simulación Multimodelo SOFA	33
4.1. Representación multimodelo en SOFA	34
4.1.1. Modelo Mecánico	35
4.1.2. Modelo de Colisiones	35

4.1.3.	Modelo Visual	35
4.1.4.	Mapeo entre modelos	36
4.2.	Estructura de Datos	37
4.2.1.	Gráfica de la Escena	37
4.2.2.	Contenedores de Datos y Motores	38
4.2.3.	Geometría y Topología	38
4.3.	Algoritmos de Simulación	39
4.3.1.	Solucionadores de Ecuaciones Diferenciales	39
4.3.2.	Solucionadores de Ecuaciones Lineales	39
4.3.3.	Solucionadores de Restricciones	40
4.3.4.	Detección y Respuesta a Colisiones	41
4.4.	Interfaz Háptica Humano Computadora	42
5.	Desarrollo del Modelo de Punción Háptica	45
5.1.	Modelo de Punción Háptica	45
6.	Casos de Estudio	51
6.1.	Simulador Háptico-Virtual de Venopunción	51
6.1.1.	Práctica de Acceso Venoso Periférico	51
6.1.2.	Modelos Anatómicos e Instrumentos Virtuales	52
6.1.3.	Modelo de Simulación	53
6.1.4.	Resultados	56
6.2.	Simulador Háptico-Virtual de Ventriculostomía	64
6.2.1.	Inserción de Catéter Ventricular	64
6.2.2.	Modelos Anatómicos e Instrumentos Virtuales	65
6.2.3.	Modelo de Simulación	65
6.2.4.	Resultados	69
7.	Conclusiones y Trabajo a Futuro	79
7.1.	Conclusiones	79
7.2.	Trabajo a Futuro	80
A.	Solución Numérica del Modelo Biomecánico	81
A.1.	Método de Elemento Finito	81
A.1.1.	Formulación Débil de las Ecuaciones de Equilibrio	81
A.1.2.	Formulación Elástica por el Método de Elemento Finito	83
A.1.3.	Discretización del Modelo Dinámico	86
A.2.	Solución del Modelo Dinámico	88
A.2.1.	Método de Euler Implícito	88

A.3. Algoritmo de Gradiente Conjugado	90
B. Desarrollo de un Plugin en SOFA	93
B.1. Ejemplo de desarrollo de un plugin en SOFA	93
B.1.1. Archivo initMyPlugin.h	94
B.1.2. Archivo initMyPlugin.cpp	95
B.1.3. Archivo CMakeLists.txt	96
B.1.4. Instalación en CMake	97
Bibliografía	97

Índice de figuras

2.1. Fuerza $d\mathbf{P}$ sobre el área dA con vector normal unitario \mathbf{n}	12
2.2. Ilustración de los componentes del tensor de esfuerzos.	12
3.1. Diagrama esquemático de renderizado háptico	23
3.2. Ejemplo del modelo de proxy virtual.	24
3.3. Fuerzas de acción durante la inserción de una aguja	26
3.4. Parámetros de fricción del modelo modificado de Karnopp	29
5.1. Máquina de estados del modelo de punción háptica	46
5.2. Diagrama del modelo de inserción de la aguja.	48
5.3. Gráfica del modelo de máxima profundidad de penetración	49
6.1. Modelos anatómicos de la extremidad superior derecha.	53
6.2. Modelos vasculares.	54
6.3. Modelo visual y de colisiones del catéter.	55
6.4. Resultado del proceso de decimar la malla triangular.	55
6.5. Generación de la malla tridimensional en Gsmh.	56
6.6. Escena de simulación de la práctica de venopunción.	57
6.7. Modelos anatómicos y de simulación de la extremidad superior derecha.	58
6.8. Representación de la simulación de punción vascular.	59
6.9. Fuerza generada por el dispositivo háptico durante la punción en la simulación.	59
6.10. Posición registrada por el dispositivo háptico durante la simulación.	60
6.11. Gráfica de la trayectoria registrada por el dispositivo háptico durante la simulación	60
6.12. Fuerza generada por el dispositivo háptico en la dirección de punción en la fase de inserción.	61

6.13. Software de segmentación de estructuras anatómicas a partir de imágenes de tomografía tomadas de un caso clínico.	66
6.14. Resultado de la segmentación y preprocesamiento de la estructuras anatómicas obtenidas a partir de imágenes de tomografía.	67
6.15. Catéter ventricular	68
6.16. Diseño de acople de catéter ventricular para el dispositivo háptico Phantom Touch.	68
6.17. Visualización de la malla de hexaedros en el framework SOFA. (a) Visualización en el plano transversal. (b) Detalle de la malla de hexaedros.	69
6.18. Visualización de la malla de colisiones utilizadas.	70
6.19. Escena de simulación para la práctica de ventriculostomía. (a), (b) Inserción del catéter ventricular. (c), (d) Visualización en transparencia del tejido óseo.	71
6.20. Líneas de referencia incorporadas al ambiente de simulación del procedimiento de ventriculostomía.	72
6.21. (a) Detalle de la pieza manufacturada para el acoplamiento del catéter ventricular. (b) Montaje en el dispositivo háptico Phantom Touch y se muestra el modelo en impresión 3D de un paciente real obtenido de la segmentación de tomografías.	74
6.22. Gráfica de fuerza generada por el dispositivo háptico durante la simulación de punción ventricular.	75
6.23. Gráfica de fuerza generada por el dispositivo háptico durante la simulación de punción ventricular en la dirección de punción.	75
6.24. Gráfica de la posición registrada por el dispositivo háptico durante la simulación de punción ventricular.	76
6.25. Gráfica de la trayectoria registrada por el dispositivo háptico durante la simulación de punción ventricular.	77
6.26. Gráfica del error de la posición del dispositivo háptico con respecto a la recta de referencia ideal.	78
6.27. Gráfica del error de la posición con respecto a la recta de referencia ideal en el intervalo de punción.	78

Índice de tablas

2.1. Resumen de las ecuaciones correspondientes al modelo elástico lineal en tres dimensiones.	18
2.2. Resumen de variables primarias correspondientes al modelo elástico lineal en tres dimensiones.	18
3.1. Especificaciones Técnicas de Dispositivo Phantom Touch	31
6.1. Parámetros utilizados en el modelo de punción propuesto.	61
6.2. Parámetros utilizados en la escena de simulación.	62
6.3. Resultados del cuestionario de evaluación.	63
6.4. Parámetros utilizados en el modelo de punción en el simulador de ventriculostomía.	73
6.5. Parámetros utilizados en la escena de simulación de la práctica de ventriculostomía.	73

Introducción

En los últimos años, se ha visto cada vez mayor interés por parte de los centros de formación en introducir sistemas que permitan el desarrollo de las habilidades y destreza en procedimientos clínicos con mayor rapidez [1, 2, 3]. Una propuesta que ha ido en aumento es la integración de sistemas hápticos en el desarrollo de simuladores de entrenamiento en el área de la salud [1]. Además, los sistemas hápticos en el entrenamiento han permitido desarrollar una experiencia más inmersiva que facilita la transición hacia la atención de un paciente real [2].

El término háptica se refiere a las tecnologías que tienen que ver con tacto, lo cual incluye la retroalimentación cutánea y kinestésica [4]. De acuerdo con los resultados obtenidos por Feygin et al. [5], el uso de sistemas hápticos se puede utilizar para el entrenamiento de habilidades neuromotrices a lo que denominaron asistencia háptica. Entre sus descubrimientos está que la retroalimentación visual sirve mejor para enseñar la forma de la trayectoria, mientras que aspectos temporales de la tarea eran más efectivos a través de la asistencia háptica [5].

En este trabajo se propone el desarrollo de un modelo de punción háptica y su implementación en un ambiente de simulación virtual de procedimientos percutáneos y de cateterismo. El objetivo a largo plazo es utilizar el entrenamiento háptico para desarrollar las habilidades de practicantes del área de la salud, así como su evaluación durante el proceso de aprendizaje.

Se obtuvieron modelos anatómicos de estructuras anatómicas a partir de modelos idealizados o modelados, y a través de la segmentación tridimensional de imágenes

1. INTRODUCCIÓN

de tomografía computarizada (CT) de acuerdo a la metodología desarrollada en [6]. Estos modelos se utilizaron para crear el entorno háptico y visual de la simulación.

En este trabajo se presentan dos casos de estudio, la simulación del procedimiento de punción venosa para acceso periférico, que consiste en insertar una aguja conocida como punzocat en una vena de una extremidad, que generalmente ocurre en la fosa antecubital o la región dorsal metacarpiana [7].

El otro caso de estudio consiste en la simulación del procedimiento de ventriculostomía realizado en la práctica de neurocirugía. Este procedimiento consiste en la inserción de un catéter, denominado estilete, a través de una perforación realizada en el cráneo, hasta alcanzar el foramen de Monro que conecta los ventrículos laterales con el tercer ventrículo.

1.1 Antecedentes

El número de sistemas de entrenamiento para diversos procedimientos médicos basados en retroalimentación háptica ha ido cada vez en aumento [3]. Coles et al. [8] los clasifica en seis categorías principales: (1) palpación, (2) laparoscopia, (3) procedimientos endovasculares, (4) endoscopia, (5) artroscopia e (6) inserción de aguja. El grupo de inserción, que es el que nos interesa en este trabajo, engloba diversos procedimientos médicos; sin embargo, muy pocos trabajos se enfocan en la intervención intravenosa o punción ventricular.

Entre los trabajos que integran sistemas hápticos y la inserción de aguja en vasos sanguíneos está el de Shin et al. [9] quienes desarrollaron un dispositivo de punción que provee retroalimentación de tres capas: elasticidad de la piel, fricción de la capa muscular, y elasticidad de los vasos sanguíneos.

En el ámbito veterinario también ha habido interés en incorporar este tipo de tecnología en el entorno educativo. Un ejemplo es el que presentan Lee et al. [2] quienes incorporaron realidad aumentada en un simulador de entrenamiento de venopunción canina. El sistema consiste en un modelo de una pata canina obtenido a través de la segmentación de imágenes de TC y su fabricación en silicón de un modelo 3D, una jeringa con un arreglo de vibromotores y un sistema de simulación de realidad aumentada que se sobrepone al modelo 3D y muestra la vascularidad objetivo.

Otro trabajo con una orientación similar es el de Xia y Sourin [10] quienes desarrollaron un sistema de venopunción con imágenes de pacientes en 2D, donde se sobrepone un modelo 3D de la vena y además, proporciona retroalimentación háptica a través de un dispositivo Phantom Omni (ahora Touch).

A pesar de ser un procedimiento diferente, la inserción de agujas en acupuntura es una tarea muy parecida que también requiere de gran precisión por parte de quien lo realiza. Heng et al. [11] desarrollaron un ambiente virtual de entrenamiento en acupuntura utilizando realidad virtual y retroalimentación de fuerza a través de un dispositivo Phantom Desktop. El sistema provee entre otras cosas, información sobre la velocidad y el ángulo de inserción de la aguja para ayudar a mejorar las habilidades del practicante.

En el ámbito de entrenamiento en neurocirugía existen diversos trabajos relacionados [12, 13, 14, 15]; sin embargo, actualmente pocos simuladores ofrecen simulación

1. INTRODUCCIÓN

para la práctica del procedimiento de ventriculostomía [16] y un número aún más reducido de simuladores háptico-virtuales.

En cuanto a simuladores virtuales de ventriculostomía podemos encontrar el desarrollado por Lemole et al. [17] quienes utilizan la plataforma de realidad virtual de ImmersiveTouch para simular la tarea de ventriculostomía, ellos utilizaron imágenes de tomografía para reconstruir modelos tridimensionales. Además, los investigadores utilizaron unos lentes espectroscópicos con un rastreador electromagnético para observar las imágenes de realidad virtual reflejadas en una superficie de espejo. El operador utiliza un dispositivo háptico Phantom (Sensable Technologies, Woburn, MA) para manipular el modelo virtual y percibir diferentes tejidos durante el procedimiento.

Por otra parte, Brown et al. [18] desarrollaron un sistema de entrenamiento virtual para el procedimiento de endoscopia ventricular. A partir de imágenes de resonancia magnética se reconstruyeron los modelos virtuales. El sistema desarrollado permite representar modelos heterogéneos utilizando diferentes modelos virtuales. Se utilizó un dispositivo para adaptar un endoscopio a un dispositivo háptico Phantom Omni. Este sistema de acoplamiento se basa en el diseño presentado por Tendrick et al. [19].

Otro prototipo desarrollado es el de Panchaphongsaphak et al. [18], quienes proponen un nuevo dispositivo háptico de un grado de libertad para simular la punción ventricular. Las fuerzas de reacción se transmiten al usuario a través de un actuador lineal mediante un mecanismo deslizante y un cable. Este sistema se plantea incorporar con el sistema de realidad virtual BrainTrain [20] presentado previamente.

El sistema desarrollado por Çakmak et al. [21] permite realizar entrenamiento háptico colaborativo y supervisado de forma remota del procedimiento de endoscopia ventricular. Este sistema permite la interacción háptica con modelos deformables obtenidos a partir de imágenes de resonancia magnética de pacientes con hidrocefalia. También se incorporó el efecto visual de sangrado en el líquido cefalorraquídeo cuando el procedimiento no se realiza de forma adecuada.

1.2 Objetivo

1.2.1 Objetivo General

- Desarrollar e implementar un modelo de punción háptica para simuladores virtuales de entrenamiento médico en procedimientos percutáneos y de cateterismo.

1.2.2 Objetivos Particulares

- Investigar e implementar un modelo de punción háptica en el framework de trabajo.
- Obtener los modelos anatómicos a partir de imágenes de tomografía o modelados por software.
- Modelar las herramientas virtuales.
- Segmentar y definir las regiones de simulación.
- Exportar los datos de interacción del usuario para su posterior análisis.

1. INTRODUCCIÓN

1.3 Justificación

En la práctica clínica, existen diversos procedimientos que se relacionan con la inserción de una aguja en los tejidos. Por ello, es importante desarrollar un modelo que permita simular el procedimiento de punción para ser integrado en simuladores de entrenamiento. En este trabajo se propone implementar el modelo de punción háptica en dos casos de estudio que simulan dos procedimientos de gran importancia clínica y que se llevan a cabo de manera muy frecuente. Estos casos de estudio consisten en la práctica de punción vascular venosa periférica y el procedimiento de ventriculostomía cerebral.

Actualmente, los simuladores comerciales para la práctica de cateterismo vascular periférico se basan en maniqués de silicón que, aunque proveen cierto grado de realismo, tienden a tener un desgaste muy rápido [22] y, sin la asistencia de un experto, no se tiene ninguna guía acerca de la técnica adecuada que se debe realizar. Es por ello que se propone un modelo de simulación a través del cual permita al practicante recibir retroalimentación tanto visual como táctil para experimentar y como una herramienta para su entrenamiento.

Por otro lado, en cuanto a la práctica de ventriculostomía, a pasar de ser uno de los procedimientos realizados con mayor frecuencia en la práctica de neurocirugía, el número de simuladores que existen actualmente para esta práctica es muy limitado [16]. Por tal motivo, se propone integrar el modelo háptico desarrollado en un simulador virtual para la práctica de ventriculostomía. Además, estudios sugieren que el uso de simuladores virtuales para la práctica de ventriculostomía, es un método válido para potenciar las habilidades de los residentes de neurocirugía [23, 24, 25].

La principal aportación de este trabajo se centra en el objetivo de desarrollar un modelo háptico de punción flexible y que pueda ser integrado en ambientes de entrenamiento virtual. Con ello, se plantea a largo plazo crear esquemas que permitan desarrollar las habilidades psicomotrices de los practicantes en el área de la salud. Asimismo, el entrenamiento previo a la atención directa con pacientes es de gran relevancia para disminuir el índice de falla por ejecución incorrecta de procedimientos técnicos y de esta forma disminuir el riesgo para el paciente durante una intervención. Por otra parte, incluir la sensación táctil en la simulación permite incrementar el grado de realismo de un procedimiento, lo cual genera un mayor grado de confianza para el practicante a la hora de atender a pacientes en un entorno real [3].

En conclusión, la transmisión de habilidades de procedimientos técnicos ha sido una tarea de gran interés en los centros formativos en diversas especialidades médicas debido a que la destreza se obtiene únicamente con la práctica clínica, por lo que el desarrollo de simuladores virtuales con modelos de interacción háptica que se aproximen a la experiencia real, resulta de gran relevancia para los practicantes en el área médica.

1. INTRODUCCIÓN

1.4 Estructura de la Tesis

La tesis que se presenta en este trabajo está conformada por los capítulos descritos brevemente a continuación:

Capítulo 1 **Introducción**

En este capítulo se incluye la introducción del trabajo, en la que se describe de manera general el trabajo presentado, los antecedentes sobre simuladores encontrados en la bibliografía en cuanto a simulación de procedimientos de punción, los objetivos y la justificación del trabajo realizado.

Capítulo 2 **Modelo Biomecánico**

Aquí se expone el modelo biomecánico que describe el comportamiento de los tejidos simulados. Éste consiste en el modelo matemático basado en la teoría del medio continuo que describe la mecánica de sólidos deformables y en particular el modelo de elasticidad lineal.

Capítulo 3 **Interacción Háptica**

En este capítulo se describen los modelos en los que se fundamenta la interacción háptica en la simulación virtual y que permiten la interacción entre el humano y el ambiente virtual a través de un dispositivo háptico.

Capítulo 4 **Framework de Simulación Multimodelo SOFA**

El framework en el que se desarrolla el simulador presentado en este trabajo se describe en este capítulo. Se exponen los principales componentes que constituyen el framework multimodelo SOFA.

Capítulo 5 **Desarrollo del Modelo de Punción Háptica**

En este capítulo se expone la principal aportación de este trabajo, ya que se describe el modelo de interacción háptica propuesto para la simulación virtual. En este modelo se incluyen fuerzas que simulan la reacción, ruptura y fricción de los tejidos, así como la trayectoria y profundidad de penetración del instrumento.

1. INTRODUCCIÓN

Capítulo 6 **Casos de Estudio**

En este capítulo se exponen dos casos de estudio que consisten en dos escenas de simulación desarrolladas en el framework SOFA: la simulación de la práctica de venopunción y la simulación del procedimiento de ventriculostomía. En estos simuladores se incluye el modelo de punción propuesto, con diferentes parámetros de acuerdo a los tejidos y al procedimiento que se está simulando.

Capítulo 7 **Conclusiones y Trabajo a Futuro**

Por último en el capítulo 7 se presentan las conclusiones de este trabajo a partir de los resultados obtenidos, así como algunas propuestas que permitirán dar continuidad a este trabajo.

Modelo Biomecánico

El modelo biomecánico que se considera para este estudio es el modelo de elasticidad lineal. En este capítulo se hace un resumen del modelo de un sólido continuo en estado estacionario y su extensión para obtener el modelo dinámico, para mayores referencias se puede consultar [26, 27].

2.1 Esfuerzos y Deformaciones

2.1.1 Modelo de Esfuerzos

Se asume que el cuerpo es continuo y se consideran dos tipos de fuerzas: de cuerpo y superficiales.

Considerando la superficie del cuerpo mostrado en la figura 2.1. Esta superficie puede ser externa o interna obtenida de una sección del cuerpo. El vector \mathbf{n} es un vector unitario normal a la superficie y que apunta hacia afuera del cuerpo. El vector incremental de fuerza $d\mathbf{P}$ actúa sobre la superficie diferencial dA . Cuando dA se aproxima a cero, se asume que la razón $d\mathbf{P}/dA$ se aproxima al valor dado por

$$\mathbf{t} = \frac{d\mathbf{P}}{dA}; \quad dA \rightarrow 0; \quad \mathbf{t} = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \quad (2.1)$$

El vector \mathbf{t} se denomina vector de tracción y tiene dimensiones $[N/m^2]$. Para cada sección, en general, este vector será diferente cuando se tome otra sección que pase por el mismo punto, por lo que, se busca obtener el tensor de esfuerzo que contiene

2. MODELO BIOMECÁNICO

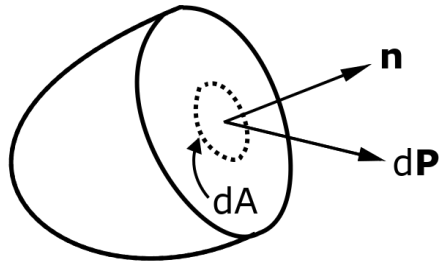


Figura 2.1: Fuerza $d\mathbf{P}$ sobre el área dA con vector normal unitario \mathbf{n} .

toda la información necesaria para determinar el vector de tracción para una sección arbitraria que pase a través de un punto en el cuerpo.

Si se considera un vector de tracción tal que se obtiene cuando se toma una sección perpendicular a los ejes coordenados, se obtienen los vectores \mathbf{S}_x cuando el vector normal \mathbf{n} (figura 2.1) apunta en la dirección x , \mathbf{S}_y cuando el vector normal \mathbf{n} apunta en la dirección y y \mathbf{S}_z cuando el vector normal \mathbf{n} apunta en la dirección z . Considerando los componentes, que se ilustran en la figura 2.2, en cada eje se obtiene el tensor

$$\mathbf{S} = \begin{bmatrix} \mathbf{s}_x^T \\ \mathbf{s}_y^T \\ \mathbf{s}_z^T \end{bmatrix} = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{yx} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{zx} & \sigma_{zy} & \sigma_{zz} \end{bmatrix} \quad (2.2)$$

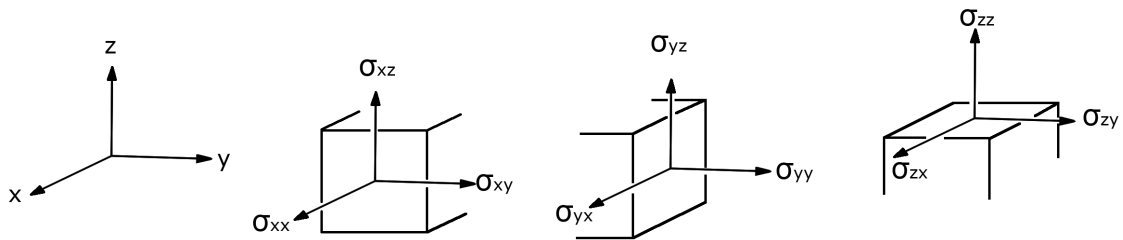


Figura 2.2: Ilustración de los componentes del tensor de esfuerzos.

Esta matriz se denomina tensor de esfuerzos y es fácil probar que la matriz es simétrica, ya que por la condición de equilibrio de momento

$$\sigma_{xy} = \sigma_{yx}; \quad \sigma_{xz} = \sigma_{zx}; \quad \sigma_{yz} = \sigma_{zy} \quad (2.3)$$

También es posible mostrar que cualquier vector de tracción se puede escribir como

$$\mathbf{t} = \mathbf{S}\mathbf{n} \quad (2.4)$$

donde

$$\mathbf{n} = \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix} \quad (2.5)$$

Además, se puede observar que en la superficie del cuerpo la ecuación 2.4 representa una condición de frontera que expresa la relación entre las fuerzas en la superficie externa y el tensor de esfuerzos. La ecuación 2.4 se puede escribir por sus componentes como

$$t_x = \mathbf{s}_x^T \mathbf{n}; \quad t_y = \mathbf{s}_y^T \mathbf{n}; \quad t_z = \mathbf{s}_z^T \mathbf{n} \quad (2.6)$$

Por otra parte, considerando equilibrio en una parte arbitraria del cuerpo. Las fuerzas que actúan en esta parte arbitraria del cuerpo están dadas por el vector \mathbf{t} sobre la superficie del cuerpo S y las fuerzas de cuerpo \mathbf{b} en la región V . El equilibrio requiere que

$$\int_S \mathbf{t} dS + \int_V \mathbf{b} dV = \mathbf{0} \quad (2.7)$$

Esta ecuación está compuesta por tres ecuaciones. Para la componente x

$$\int_S t_x dS + \int_V b_x dV = 0 \quad (2.8)$$

que con la ecuación 2.6 se puede reescribir como

$$\int_S \mathbf{s}_x \mathbf{n} dS + \int_V b_x dV = 0 \quad (2.9)$$

El primer término se puede reformular utilizando el teorema de la divergencia para obtener

$$\int_V (\text{div } \mathbf{s}_x + b_x) dV = 0 \quad (2.10)$$

2. MODELO BIOMECÁNICO

Como se cumple para cualquier volumen arbitrario V , se concluye que

$$\text{div } \mathbf{s}_x + b_x = 0 \quad (2.11)$$

Sustituyendo \mathbf{s}_x y utilizando la definición de divergencia, se tiene que

$$\frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{xy}}{\partial y} + \frac{\partial \sigma_{xz}}{\partial z} + b_x = 0 \quad (2.12)$$

Haciendo un análisis similar con las componentes y y z se obtienen las siguientes ecuaciones

$$\begin{aligned} \frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{xy}}{\partial y} + \frac{\partial \sigma_{xz}}{\partial z} + b_x &= 0 \\ \frac{\partial \sigma_{yx}}{\partial x} + \frac{\partial \sigma_{yy}}{\partial y} + \frac{\partial \sigma_{yz}}{\partial z} + b_y &= 0 \\ \frac{\partial \sigma_{zx}}{\partial x} + \frac{\partial \sigma_{zy}}{\partial y} + \frac{\partial \sigma_{zz}}{\partial z} + b_z &= 0 \end{aligned} \quad (2.13)$$

Estas ecuaciones expresan la condición de equilibrio para el cuerpo.

Si definimos

$$\tilde{\nabla}^T = \begin{bmatrix} \frac{\partial}{\partial x} & 0 & 0 & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} & 0 \\ 0 & \frac{\partial}{\partial y} & 0 & \frac{\partial}{\partial x} & 0 & \frac{\partial}{\partial z} \\ 0 & 0 & \frac{\partial}{\partial x} & 0 & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \end{bmatrix}; \quad \boldsymbol{\sigma} = \begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{zz} \\ \sigma_{xy} \\ \sigma_{xz} \\ \sigma_{yz} \end{bmatrix} \quad (2.14)$$

donde $\boldsymbol{\sigma}$ contiene todas las componentes del tensor de esfuerzos y $\tilde{\nabla}^T$ es un operador diferencial matricial. Con estas definiciones el sistema de ecuaciones 2.13 se puede reescribir como

$$\tilde{\nabla}^T \boldsymbol{\sigma} + \mathbf{b} = \mathbf{0} \quad (2.15)$$

2.1.2 Modelo de Deformaciones

Una deformación se define como el cambio de distancia entre dos puntos materiales vecinos y como el cambio de ángulo entre dos líneas que se intersectan.

2.1 Esfuerzos y Deformaciones

Un punto en el cuerpo se describe por las coordenadas (x, y, z) . Después de la deformación, el punto se ha movido, lo cual se representa con las coordenadas $(x + u_x, y + u_y, z + u_z)$. Los cambios u_x, u_y, u_z se denominan componentes de desplazamiento y se representan como

$$\mathbf{u} = \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix} \quad (2.16)$$

Además, los desplazamientos de un punto (x, y, z) están dados por $\mathbf{u} = \mathbf{u}(x, y, z)$. Para un punto vecino $(x + dx, y + dy, z + dz)$, los desplazamientos serán $\mathbf{u} + d\mathbf{u}$. Utilizando la regla de la cadena, $d\mathbf{u}$ está dado por

$$\begin{aligned} du_x &= \frac{\partial u_x}{\partial x} dx + \frac{\partial u_x}{\partial y} dy + \frac{\partial u_x}{\partial z} dz \\ du_y &= \frac{\partial u_y}{\partial x} dx + \frac{\partial u_y}{\partial y} dy + \frac{\partial u_y}{\partial z} dz \\ du_z &= \frac{\partial u_z}{\partial x} dx + \frac{\partial u_z}{\partial y} dy + \frac{\partial u_z}{\partial z} dz \end{aligned} \quad (2.17)$$

Las derivadas de la forma $\partial u_x / \partial x$ se denominan gradientes de desplazamiento.

Las elongaciones infinitesimales de líneas paralelas a los ejes coordenados se denotan como deformaciones normales, dadas por

$$\epsilon_{xx} = \frac{\partial u_x}{\partial x}; \quad \epsilon_{yy} = \frac{\partial u_y}{\partial y}; \quad \epsilon_{zz} = \frac{\partial u_z}{\partial z} \quad (2.18)$$

Por otra parte, el cambio angular resultante de la deformación de dos líneas AB y AC ortogonales entre sí y paralelas a los ejes coordenados, está dado por

$$\gamma_{xy} = \frac{\partial u_x}{\partial y} + \frac{\partial u_y}{\partial x}; \quad \gamma_{xz} = \frac{\partial u_x}{\partial z} + \frac{\partial u_z}{\partial x}; \quad \gamma_{yz} = \frac{\partial u_y}{\partial z} + \frac{\partial u_z}{\partial y} \quad (2.19)$$

Este cambio se denomina deformación cortante, donde los subíndices indican a qué ejes son perpendiculares las líneas AB y AC. Estas deformaciones son simétricas $\gamma_{ij} = \gamma_{ji}$.

Con las deformaciones normales (2.18) y las deformaciones cortantes (2.19), se obtiene una descripción completa de la deformación del cuerpo. Si definimos las

2. MODELO BIOMECÁNICO

matrices

$$\boldsymbol{\epsilon} = \begin{bmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ \epsilon_{zz} \\ \gamma_{xy} \\ \gamma_{xz} \\ \gamma_{yz} \end{bmatrix}; \quad \tilde{\nabla} = \begin{bmatrix} \frac{\partial}{\partial x} & 0 & 0 \\ 0 & \frac{\partial}{\partial y} & 0 \\ 0 & 0 & \frac{\partial}{\partial z} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} & 0 \\ \frac{\partial}{\partial z} & 0 & \frac{\partial}{\partial x} \\ 0 & \frac{\partial}{\partial z} & \frac{\partial}{\partial y} \end{bmatrix} \quad (2.20)$$

donde $\boldsymbol{\epsilon}$ contiene todas las deformaciones y $\tilde{\nabla}$ es un operador diferencial matricial. Con estas definiciones podemos escribir

$$\boldsymbol{\epsilon} = \tilde{\nabla} \mathbf{u} \quad (2.21)$$

Cabe destacar que el operador $\tilde{\nabla}$ cumple con la condición de equilibrio en ambas ecuaciones 2.15 y 2.21. La ecuación 2.21 se conoce como relación cinemática.

2.2 Modelo Elástico Lineal

La ecuación constitutiva que rige el modelo de elasticidad lineal es la ley de Hooke expresada como

$$\boldsymbol{\sigma} = \mathbf{D} \boldsymbol{\epsilon} \quad (2.22)$$

donde

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{zz} \\ \sigma_{xy} \\ \sigma_{xz} \\ \sigma_{yz} \end{bmatrix}; \quad \mathbf{D} = \begin{bmatrix} D_{11} & D_{12} & \cdots & D_{16} \\ D_{21} & D_{22} & \cdots & D_{26} \\ \vdots & \vdots & \ddots & \vdots \\ D_{61} & D_{62} & \cdots & D_{66} \end{bmatrix}; \quad \boldsymbol{\epsilon} = \begin{bmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ \epsilon_{zz} \\ \gamma_{xy} \\ \gamma_{xz} \\ \gamma_{yz} \end{bmatrix} \quad (2.23)$$

\mathbf{D} es la matriz constitutiva. Al ser un modelo lineal \mathbf{D} es constante para una posición dada. También se le conoce como elasticidad de Cauchy, ya que posee propiedades lineales y una relación uno a uno entre esfuerzos y deformaciones. Además, si se requiere que la energía de deformación para un estado de deformación dado solo dependa del estado de la deformación y no de la forma en que se obtuvo esa deformación, se obtiene lo que se denomina como hiperelasticidad o elasticidad de Green. En este caso la matriz constitutiva \mathbf{D} es simétrica

$$\mathbf{D} = \mathbf{D}^T \quad (2.24)$$

2.3 Ecuaciones que Rigen el Modelo Elástico Lineal

Además, es fácil verificar que para todo $\epsilon \neq \mathbf{0}$ la matriz \mathbf{D} es definida positiva [26], entonces

$$\epsilon^T \mathbf{D} \epsilon > 0 \quad (2.25)$$

Como \mathbf{D} es definida positiva, se concluye que \mathbf{D} es no singular, esto es que

$$\det \mathbf{D} \neq 0 \quad (2.26)$$

Lo cual significa que \mathbf{D} es invertible para obtener

$$\epsilon = \mathbf{C} \sigma; \quad \mathbf{C} = \mathbf{D}^{-1} \quad (2.27)$$

donde \mathbf{C} se denomina matriz de compliancia o de flexibilidad [26].

Por otra parte, si se asume que el material es isotrópico, la matriz de \mathbf{D} adopta la forma

$$\mathbf{D} = \frac{E}{(1 + \nu)(1 - 2\nu)} \begin{bmatrix} 1 - \nu & \nu & \nu & 0 & 0 & 0 \\ \nu & 1 - \nu & \nu & 0 & 0 & 0 \\ \nu & \nu & 1 - \nu & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{2}(1 - \nu) & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2}(1 - \nu) & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{2}(1 - \nu) \end{bmatrix} \quad (2.28)$$

donde \mathbf{D} solo depende de dos coeficientes, el módulo de Young E y el coeficiente o relación de Poisson ν . Estos coeficientes se obtienen a partir de datos experimentales.

2.3 Ecuaciones que Rigen el Modelo Elástico Lineal

En resumen, las ecuaciones que rigen el modelo en estado estacionario son las siguientes.

Los esfuerzos σ deben cumplir con las ecuaciones diferenciales de equilibrio

$$\tilde{\nabla}^T \sigma + \mathbf{b} = \mathbf{0} \quad (2.29)$$

donde \mathbf{b} es el vector de fuerza de cuerpo. Las deformaciones ϵ deben cumplir con la relación cinemática

$$\epsilon = \tilde{\nabla} \mathbf{u} \quad (2.30)$$

2. MODELO BIOMECÁNICO

donde \mathbf{u} es el vector de desplazamiento. Si se asumen pequeñas deformaciones, estas ecuaciones se cumplen para cualquier material sólido que se puede considerar como un continuo. Si se asume que la relación constitutiva tiene un comportamiento elástico lineal, se cumple la ley de Hooke dada por

$$\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\epsilon} \quad (2.31)$$

Estas ecuaciones se tienen que resolver para ciertas condiciones de frontera que se expresan como vectores de tracción \mathbf{t} o desplazamientos impuestos \mathbf{u}

$$\begin{aligned} \mathbf{t} &= \mathbf{S}\mathbf{n} = \mathbf{h} & \text{en } S_h \\ \mathbf{u} &= \mathbf{g} & \text{en } S_g \end{aligned} \quad (2.32)$$

donde \mathbf{h} y \mathbf{g} son vectores conocidos. S_h es una parte de la frontera S donde tiene efecto el vector de tracción \mathbf{t} y S_g la parte de la frontera S en donde está dado el desplazamiento \mathbf{u} . La suma de las fronteras S_h y S_g constituyen la frontera completa.

En las tablas 2.1 y 2.2 se muestra el resumen de las ecuaciones y de las variables primarias, respectivamente, del modelo elástico lineal en tres dimensiones.

Tipo de ecuación		Número de ecuaciones
Equilibrio	$\tilde{\nabla}^T \boldsymbol{\sigma} + \mathbf{b} = \mathbf{0}$	3
Relaciones Constitutivas	$\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\epsilon}$	6
Cinemáticas	$\boldsymbol{\epsilon} = \tilde{\nabla}\mathbf{u}$	6
Total		15

Tabla 2.1: Resumen de las ecuaciones correspondientes al modelo elástico lineal en tres dimensiones [26].

Tipo de incógnita		Número de incógnitas
Esfuerzos	$\boldsymbol{\sigma}$	6
Deformaciones	$\boldsymbol{\epsilon}$	6
Desplazamientos	\mathbf{u}	3
Total		15

Tabla 2.2: Resumen de variables primarias correspondientes al modelo elástico lineal en tres dimensiones [26].

2.3.1 Modelo Dinámico de Estructuras Elásticas

Cuando los desplazamientos de un cuerpo elástico varían con el tiempo, se consideran dos conjuntos de fuerzas en el modelo. La primera es la fuerza de inercia, que por una aceleración caracterizada por $\ddot{\mathbf{u}}$ se puede remplazar por su equivalente estático $-\rho\ddot{\mathbf{u}}$ utilizando el principio de d'Alembert. Esta fuerza tiene componentes en la misma dirección que los del vector de desplazamiento \mathbf{u} dado por unidad de volumen. En este contexto ρ es la función de densidad (masa por unidad de volumen) [27].

La segunda fuerza surge debido a las fuerzas que se oponen al movimiento o fuerzas de fricción. Estas fuerzas se originan debido a movimientos microestructurales, resistencia del aire, etc., y generalmente están relacionadas de forma no lineal con la velocidad $\dot{\mathbf{u}}$. Por simplicidad, solo se considera la resistencia viscosa lineal que resultan en fuerzas volumétricas unitarias en un problema de equivalente estático de magnitud $-\mu\dot{\mathbf{u}}$. Donde μ es un conjunto de parámetros de viscosidad [27]. En la práctica, tanto la función de densidad ρ como los parámetros de viscosidad μ , se obtienen de manera experimental a partir de la caracterización del cuerpo que se desea modelar.

Con lo anterior, las fuerzas de cuerpo \mathbf{b} de la ecuación 2.29 se pueden reemplazar por

$$\mathbf{b} = \bar{\mathbf{b}} - \rho\ddot{\mathbf{u}} - \mu\dot{\mathbf{u}} \quad (2.33)$$

donde $\bar{\mathbf{b}}$ es la fuerza de cuerpo por unidad de volumen. Finalmente, las ecuaciones que rigen el modelo elástico lineal dinámico se pueden expresar como

$$\tilde{\nabla}^T \boldsymbol{\sigma} + \bar{\mathbf{b}} = \rho\ddot{\mathbf{u}} + \mu\dot{\mathbf{u}} \quad (2.34)$$

con condiciones de frontera

$$\begin{aligned} \mathbf{t} = \mathbf{S}\mathbf{n} = \mathbf{h} & \quad \text{en } S_h \\ \mathbf{u} = \mathbf{g} & \quad \text{en } S_g \end{aligned} \quad (2.35)$$

2. MODELO BIOMECÁNICO

Interacción Háptica

El término *háptica* se cree que se deriva del término griego *haptesthai*, que significa *relacionado con el sentido del tacto*. En la literatura de neurociencias, háptica se refiere al estudio del sentido del tacto, específicamente a través de la *cinestesia*, que mediante fuerzas se relaciona con la percepción del equilibrio, de la posición de las partes de cuerpo y a través de los receptores cutáneos, asociados con la percepción y manipulación de objetos. En el ámbito de la robótica y realidad virtual, háptica se refiere de manera amplia a las interacciones entre robots, humanos y ambientes reales, remotos o simulados en diferentes combinaciones. [28].

La fuerza de retroalimentación que se experimenta en un simulador virtual es una aproximación de la sensación que se experimenta en un procedimiento real. El incluir la sensación háptica en los sistemas de entrenamiento permite añadir el estímulo táctil a la simulación, esto a su vez, permite al usuario tener una experiencia cada vez más cercana a lo que experimentaría al realizar un procedimiento en el mundo real.

En este capítulo se describe el modelo de renderizado de fuerzas en un ambiente virtual. Además, se presenta el modelo de punición háptica propuesto por Vidal [29] y un modelo de fricción utilizando el modelo propuesto por Richard et al [30] que es una modificación del modelo propuesto por Karnopp [31].

3.1 Renderizado Háptico

Se le denomina renderizado háptico al proceso de calcular y generar las fuerzas que resultan de los contactos con objetos virtuales a partir de la interacción del usua-

3. INTERACCIÓN HÁPTICA

rio [28].

Un factor importante a considerar en el proceso de renderizado háptico es la frecuencia a la que se generan las fuerzas, ya que se sabe que los mecanorreceptores en la piel humana responden a frecuencias de hasta 10 kHz [32], por lo que de acuerdo al teorema de Nyquist-Shannon se requeriría que el dispositivo generara fuerzas a una frecuencia de al menos 20 kHz. Sin embargo, los dispositivos actuales no cubren ese ancho de banda y en la práctica la mayoría de los sistemas hápticos se implementan con al menos 1 kHz de muestreo que es adecuado para ambientes virtuales que involucran materiales suaves [28].

El proceso computacional de renderizado háptico se puede formular en siete pasos secuenciales para cada ciclo que se debe de completar, típicamente, en menos de 1 ms para mantener estabilidad y realismo:

1. sensado;
2. cinemática;
3. detección de colisiones;
4. determinar el punto de superficie;
5. cálculo de fuerza;
6. cinemática;
7. acción.

Como se muestra en la figura 3.1 los desplazamientos registrados en las articulaciones del dispositivo (Θ) (1) son procesados a través de la cinemática (2), en el ambiente virtual ocurre la detección de colisiones (3), determinación de punto de superficie (4), cálculo de fuerzas (5), a partir de estos resultados, se obtiene la cinemática en el espacio de articulaciones (6), y se amplifican las señales para llevar a cabo la generación de fuerzas en el dispositivo (7).

La posición y velocidad se obtienen de los sensores, generalmente, en el espacio articular. Éstos se deben de convertir a través del modelo de cinemática directa y la matriz Jacobiana del mecanismo del dispositivo, a la posición y velocidad cartesiana de la interfaz con el operador. Se refiere como punto de interacción háptica al punto final del objeto virtual que manipula el operador.

El algoritmo de detección de colisiones determina si la posición del punto de interacción háptica al momento actual presenta una colisión con un objeto virtual,

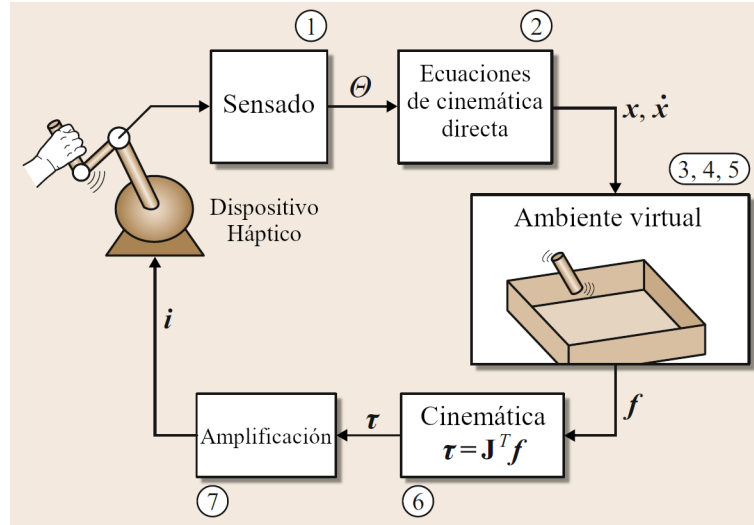


Figura 3.1: Diagrama esquemático de renderizado háptico. Los objetos en el ambiente virtual se mueven de acuerdo a los desplazamientos del dispositivo háptico [28].

generalmente, esto significa determinar si el punto de interacción se encuentra penetrando o sobre la superficie del objeto, representado por polígonos o splines.

Una vez que se determina que el punto de interacción se encuentra dentro del objeto virtual, se debe calcular la fuerza a renderizar. Para determinar la fuerza, comúnmente se utiliza un modelo de resorte o ley de Hooke

$$\mathbf{f} = k\mathbf{x} \quad (3.1)$$

donde \mathbf{x} es un vector que va del punto final de interacción háptica al punto mas cercano de la superficie del objeto. Cuando el valor de k es suficientemente grande, la superficie del objeto se sentirá como una pared perpendicular a \mathbf{x} . En ambientes virtuales con geometrías complejas formadas por mallas de polígonos donde cada polígono es en esencia una pared virtual. En este tipo de modelos al punto de interacción háptica se le denomina proxy [33]. En la figura 3.2 se muestra un ejemplo del modelo de proxy.

El modelo basado sólo en rigidez puede ser modificado para proveer de otros efectos, mediante el uso del modelo de acoplamiento virtual. Es posible agregar un modelo de amortiguamiento paralelo o perpendicular a la superficie. Además, se puede incorporar un modelo de fricción para proveer una sensación más realista de la

3. INTERACCIÓN HÁPTICA

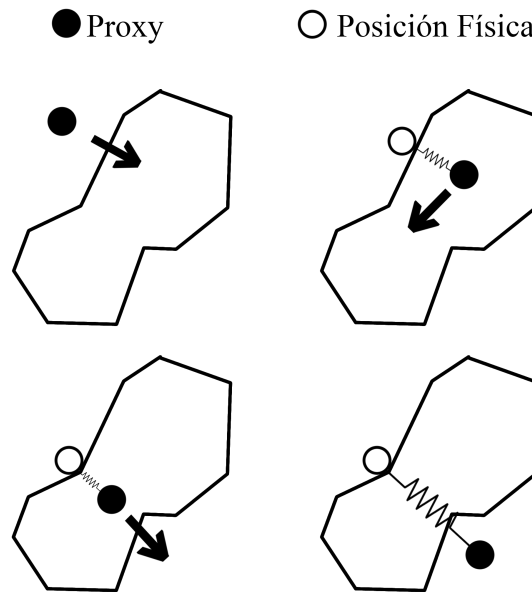


Figura 3.2: Ejemplo del modelo de proxy virtual. Basado en el diagrama de Ruspini, et al. [33].

superficie, entre otros efectos.

El modelo de resorte es un caso especial del acoplamiento virtual. El acoplamiento virtual es una abstracción del modelo de interpenetración de renderizado de fuerzas. Se asume que los objetos son rígidos pero se conectan con el operador a través de un resorte virtual, lo cual impone una rigidez máxima efectiva.

Para obtener mayor estabilidad en los contactos muchas veces es conveniente utilizar un modelo más sofisticado, por ejemplo, agregando un término de amortiguamiento generalizando el modelo de renderizado de fuerza de la ecuación 3.1 como

$$\mathbf{f} = k\mathbf{x} + b\dot{\mathbf{x}} \quad (3.2)$$

donde los parámetros k y b pueden ser ajustados empíricamente para operar de manera más estable y con mejor desempeño.

La fuerza calculada en el espacio cartesiano debe de ser transformada a pares en

el espacio articular. Generalmente el calculo se realiza mediante

$$\boldsymbol{\tau} = \mathbf{J}^T \mathbf{f} \quad (3.3)$$

donde $\boldsymbol{\tau}$ es el par enviado a los actuadores, \mathbf{f} es el vector de fuerza deseado, y \mathbf{J}^T es la transpuesta de la matriz Jacobiana del dispositivo háptico.

3.2 Simulación de Punción Háptica

3.2.1 Modelo de Punción Háptica

En esta sección se presenta un modelo de renderizado háptico para la simulación de punción. El modelo que se presenta está basado en la simulación de inserción de una aguja propuesto por Vidal [29]. El modelo de Vidal se basa en que la fuerza de reacción de los tejidos se mapea a partir de tablas precalculadas de acuerdo a la intensidad en escala de grises de los píxeles de imágenes médicas. Además, se modela la restricción del movimiento de la aguja a la trayectoria de inserción.

Este modelo considera dos modelos hápticos que simulan la penetración. El primero consiste en que cuando la fuerza aplicada sobre el tejido alcanza cierto umbral, el modelo de renderizado cambia al siguiente modo que permite la penetración del tejido. Este modelo se representa como

$$\mathbf{f}(t) = \mathbf{f}_1(t) + \mathbf{f}_2(t) + \mathbf{g} \quad (3.4)$$

donde $\mathbf{f}(t)$ es la fuerza de reacción que se aplica al dispositivo háptico al tiempo t . \mathbf{g} es una constante de fuerza que corrige el efecto debido a la gravedad.

1. El vector $\mathbf{f}_1(t)$ representa la fuerza de reacción a la resistencia del tejido durante la penetración de la aguja. Se utilizan tres submodelos para calcular $\mathbf{f}_1(t)$ dependiendo si: (a) la aguja está penetrando los tejidos, (b) la aguja está en contacto con el hueso o (c) la aguja se extrae del tejido.
 - a) Sea $\mathbf{PosAguja}(t)$ el vector dado por la posición al tiempo t . Cuando la aguja se presiona sobre el tejido, la resistencia de este aumenta, esto es, $|\mathbf{PosAguja}(t-1) - \mathbf{PosAguja}(t)|$ será mayor, de forma que el practicante percibirá mayor resistencia.

3. INTERACCIÓN HÁPTICA

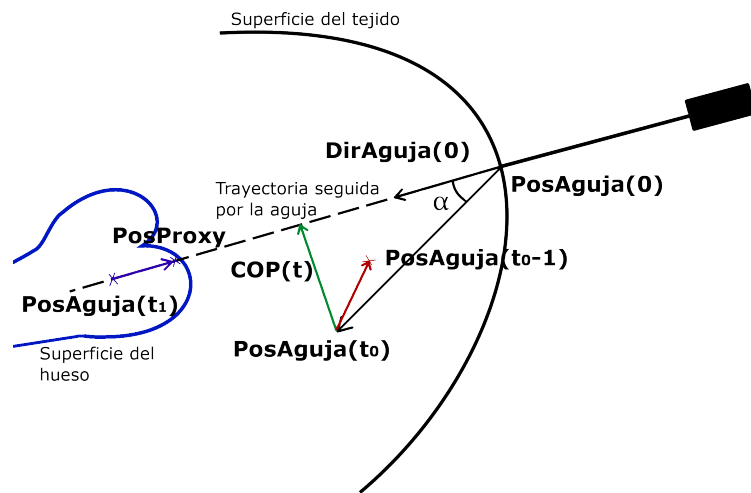


Figura 3.3: Fuerzas en acción durante la inserción de una aguja (con $\text{PosAguja}(t)$ y $\text{DirAguja}(t)$ como la posición y la dirección de la aguja en el tiempo t , respectivamente, t es igual a 0 al comienzo de la inserción). En rojo se muestra la fuerza de reacción de la resistencia de los tejidos suaves al tiempo t_0 . En azul se muestra la fuerza que restringe la aguja para permanecer fuera del hueso al tiempo t_1 (PosProxy es la última posición de la punta de la aguja antes del contacto con el hueso). En verde se muestra la fuerza que restringe la aguja para seguir una línea recta. Esta figura se basa en la descripción del artículo de Vidal [29]

3.2 Simulación de Punción Háptica

Sea $\mathbf{f}_{1a}(t)$ la fuerza de reacción del modelo basado en la ley de Hooke donde el efecto de reacción es proporcional a la profundidad de penetración de la aguja entre el tiempo $t-1$ y el tiempo t . En la figura 3.3 se muestra con flechas rojas la fuerza de reacción del tejido

$$\mathbf{f}_{1a} = K_1(\mathbf{PosAguja}(t-1) - \mathbf{PosAguja}(t)) \quad (3.5)$$

donde K_1 es una constante que se relaciona con las propiedades del tejido.

Sin embargo, para evitar la sensación de vibración debido al ruido de la señal, se utiliza un filtro digital sobre las últimas m iteraciones

$$\mathbf{f}_1(t) = \frac{1}{m} \sum_{i=t-m}^t \mathbf{SubFuerza}(i) \quad (3.6)$$

donde

$$\mathbf{SubFuerza}(i) = K_1(\mathbf{PosAguja}(i-1) - \mathbf{PosAguja}(i)) \quad (3.7)$$

En este modelo, el valor de K_1 se obtiene de tablas de precalculadas de acuerdo a la intensidad del vóxel de imágenes de tomografía. Por lo que, de acuerdo a una intensidad de vóxel dada en unidades de Hounsfield, se mapea un valor de coeficiente K_1 . Con esta función, mayor resistencia es dada a estructuras más duras, como el hueso, y menor resistencia a tejidos suaves. Los parámetros que mapean esta función se ajustan de manera experimental.

Cuando la aguja traspasa de un tejido a otro, se percibe una fuerza mayor. En este modelo, cuando la punta de la aguja se coloca en una interfaz entre dos tejidos, la fuerza $\mathbf{SubFuerza}(i)$ se multiplica por otra constante $K_{interface}$.

- b) Por otro lado, para estructuras más rígidas, como el hueso, este modelo no es aceptable, ya que la fuerza sería mucho mayor y el modelo se vuelve inestable. Para evitar la penetración de tejidos como el hueso, la fuerza \mathbf{f}_1 se cambia por un modelo similar al de proxy [33]. En este modelo el proxy virtual representa la posición del dispositivo cuando entra al hueso mientras que el dispositivo penetra ligeramente al hueso. El dispositivo

3. INTERACCIÓN HÁPTICA

háptico se restringe a alcanzar el proxy que se encuentra sobre la superficie del hueso

$$\mathbf{SubFuerza}(i) = K_{hueso}(\mathbf{PosProxy} - \mathbf{PosAguja}(i)) \quad (3.8)$$

donde $\mathbf{PosProxy}$ es la posición de la aguja durante el contacto con el hueso.

- c) Cuando la aguja es extraída, la fuerza aplicada es relativamente constante.
2. Durante un procedimiento real, es difícil desplazar la aguja fuera de su trayectoria inicial. En la simulación, una vez que la aguja está en el cuerpo, ésta es restringida a seguir una trayectoria definida por su orientación al momento de la inserción como se muestra en color verde la fuerza normal $\mathbf{COP}(t)$ a la trayectoria $\mathbf{DirAguja}(0)$ en la figura 3.3.

La fuerza $\mathbf{f}_2(t)$ corresponde a la fuerza que se requiere para restringir el movimiento perpendicular a la trayectoria de inserción de la aguja

$$\mathbf{f}_2(t) = K_2(|\mathbf{f}_c(t)|)\mathbf{f}_c(t) \quad (3.9)$$

donde

$$\mathbf{f}_c(t) = \frac{K_3}{n} \sum_{j=t-n}^t \mathbf{COP}(j) \quad (3.10)$$

$$K_2(x) = 1 - \mathit{gauss}(x) \quad \forall x, K_2(x) \in [0, 1]$$

K_3 es una constante que se obtiene experimentalmente de acuerdo al efecto deseado y $\mathit{gauss}(x)$ es una función Gaussiana. $\mathbf{COP}(t)$ es un vector que se puede calcular de forma geométrica a partir de $\mathbf{DirAguja}(0)$, $\mathbf{PosAguja}(0)$ y $\mathbf{DirAguja}(t)$, que son vectores conocidos. De igual manera, para mejorar la estabilidad, la fuerza se promedia a partir de las últimas n iteraciones. Para evitar las vibraciones de la aguja cuando se tienen pequeñas perturbaciones en la trayectoria, esto es, cuando $|\mathbf{f}_c(t)|$ es despreciable, $K_2(x)$ se utiliza como una función suave que mantiene el valor en cero.

3.2.2 Modelo de Fricción

En esta sección se presenta un modelo de fricción propuesto por Richard et al. [30] que es una versión modificada del modelo propuesto por Karnopp [31]. Este

3.2 Simulación de Punción Háptica

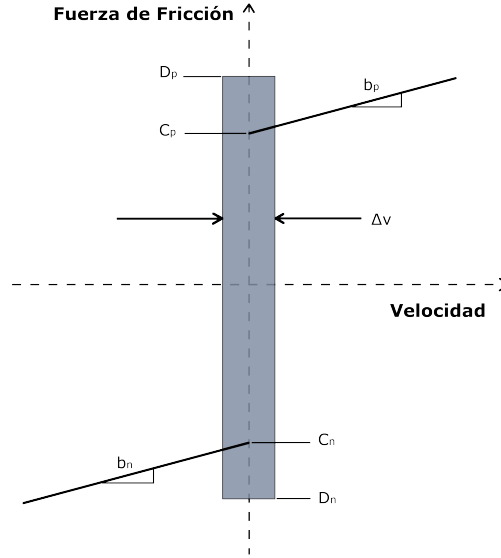


Figura 3.4: Parámetros de fricción del modelo modificado de Karnopp [30]

modelo incluye el comportamiento del modelo de fricción de Coulomb y el efecto de viscosidad; además, permite simular fricción asimétrica para valores positivos y negativos de velocidad. El modelo en una dimensión se expresa como

$$f_{friccin}(\dot{x}, f_a) = \begin{cases} C_n \operatorname{sgn}(\dot{x}) + b_n \dot{x} & \text{para } \dot{x} < -\Delta v/2 \\ \max(D_n, f_a) & \text{para } -\Delta v/2 \leq \dot{x} < 0 \\ \min(D_p, f_a) & \text{para } 0 \leq \dot{x} < \Delta v/2 \\ C_p \operatorname{sgn}(\dot{x}) + b_p \dot{x} & \text{para } \dot{x} \geq \Delta v/2 \end{cases} \quad (3.11)$$

donde

- C_p y C_n son los valores positivo y negativo de la fricción dinámica;
- b_p y b_n son los valores positivo y negativo de fricción viscosa;
- \dot{x} es la velocidad relativa entre las superficies de contacto;
- D_p y D_n son los valores positivo y negativo de la fricción estática;
- Δv es el valor por debajo del cual la velocidad se consierda cero, y;
- f_a es la suma de fuerzas aplicadas al sistema que no están relacionados con la fricción

Este modelo simula el comportamiento básico de deslizamiento-adhesión y al permitir la asimetría se obtiene un modelo que ofrece mayor libertad de ajuste a los datos que se obtienen de manera experimental en diferentes pruebas de materiales. En la figura 3.4 se muestran los parámetros que caracterizan a este modelo.

3.3 Dispositivo Háptico Phantom Touch

Un dispositivo háptico de escritorio es un brazo robótico con una pluma o pieza que le permite al usuario manipular el dispositivo, este dispositivo es capaz de registrar, seguir el movimiento del lápiz, y proveer retroalimentación de fuerza [34]. El dispositivo Phantom Touch de la empresa 3D Systems¹ es un dispositivo háptico de escritorio que proporciona fuerzas de retroalimentación a la mano del usuario, esto permite percibir objetos virtuales y evocar sensaciones táctiles acorde con la interacción de los objetos en un ambiente virtual.

Este dispositivo cuenta con seis grados de libertad de movimiento y tres grados de libertad de retroalimentación de fuerza. Las especificaciones técnicas del dispositivo se describen en la tabla 3.1.

El dispositivo Touch puede ser utilizado en aplicaciones de simulación médica, aprendizaje, evaluación, control robótico, diseño, rehabilitación, entre otras.

Con la finalidad de implementar las funcionalidades deseadas del dispositivo se utiliza el conjunto de herramientas de desarrollo OpenHaptics, proporcionado por la misma empresa.

¹<https://es.3dsystems.com/haptics-devices/touch>

3.3 Dispositivo Háptico Phantom Touch

Especificaciones	TOUCH
Espacio de Trabajo	160 x 120 x 70 mm
Peso	~1.42 kg
Rango de Movimiento	Movimiento de mano con pivote en la muñeca
Resolución nominal	~0.055 mm
Fuerza de Fricción	<0.26 N
Fuerza máxima	3.3 N
Fuerza Continua Máxima (24 hrs)	>0.88 N
Rigidez	eje-x >1.26 N/mm eje-y >2.31 N/mm eje-z >1.02 N/mm
Inercia (masa aparente)	~45 g
Retroalimentación de Fuerza (3 grados de libertad)	x, y, z
Sensores de posición/Entrada (6 grados de libertad), cardán	x, y, z (codificadores digitales) Pitch, Roll, Yaw ($\pm 5\%$ potenciómetros lineales)
Interface	USB 2.0

Tabla 3.1: Especificaciones Técnicas de Dispositivo Phantom Touch

3. INTERACCIÓN HÁPTICA

Framework de Simulación Multimodelo SOFA

En este capítulo, basado principalmente en el artículo de Faure et al [35], se describe de manera general la biblioteca SOFA (Simulation Open Framework Architecture¹) que es una biblioteca de código abierto escrita principalmente en el lenguaje C++ que se enfoca en simulación aplicada a la medicina. Esta biblioteca está orientada principalmente a la simulación de sólidos deformables, sin embargo también se extiende a otros tipos de simulación física, como de fluidos, fenómenos eléctricos, termodinámicos, etc. Además es una biblioteca que integra de manera modular y organizada con una estructura de datos tipo gráfica de escenas.

Cada componente de la simulación como los grados de libertad (GDL), fuerzas, restricciones, modelos de ecuaciones diferenciales, algoritmos de solución de sistemas de ecuaciones lineales y de integración, algoritmos de detección de colisiones o interfaz con dispositivos de interacción se encuentra encapsulado y permiten la comunicación entre sí. Los objetos que constituyen la simulación pueden representarse utilizando diferentes modelos, donde cada uno está optimizado para utilizarse para diferentes tareas, como por ejemplo, el cálculo de fuerzas internas, la detección de colisiones o la visualización. Para la sincronización de los modelos se utilizan mecanismos de mapeo que permiten la interacción entre ellos. La librería permite combinar de manera transparente las CPU y GPU para aprovechar el poder computacional de las arquitecturas.

¹<https://www.sofa-framework.org>

4. FRAMEWORK DE SIMULACIÓN MULTIMODELO SOFA

Una gran ventaja de la implementación de SOFA es su estructura modular ya que permite a los investigadores y desarrollares enfocarse en el módulo de interés y a la vez reutilizar módulos desarrollados por otros expertos. Esta librería permite desarrollar simulación con múltiples objetos en lo que se denomina escena. La escena sigue una estructura jerárquica donde se describen los objetos y algoritmos utilizados a través de un archivo en formato XML (eXtensible Markup Language), lo cual facilita evaluar de forma muy dinámica los diferentes componentes con un mismo propósito y sin alterar los demás.

Desde su lanzamiento en 2007 SOFA ha ido evolucionando y se encuentra continuamente actualizando de modo que se ha convertido en una biblioteca de alto desempeño utilizado por un gran número de investigadores, universidades y empresas [35].

4.1 Representación multimodelo en SOFA

En cuanto a la representación de los modelos físicos, SOFA utiliza una representación distinta y optimizada dependiendo del tipo de cálculo que debe de realizar. De manera general un objeto físico en SOFA se representa a través de tres modelos: un modelo interno con grados de libertad independiente, la masa y las leyes constitutivas, un modelo de colisiones con geometría de contactos, y un modelo visual con una geometría más fina y diferentes parámetros de renderizado. La ventaja de esta estructura es que cada modelo se puede diseñar de manera independiente de los demás y se pueden crear combinaciones más complejas. Además, durante la simulación los modelos se acoplan entre ellos a través de un mecanismo de mapeo donde se propagan las fuerzas y los desplazamientos a través de cada modelo [35].

Un modelo deformable en SOFA se puede representar a través de los tres modelos convencionales utilizados en esta biblioteca. El primero consiste en un modelo de deformaciones que representa el comportamiento mecánico del objeto, la geometría de este modelo está optimizada para el cálculo de las fuerzas internas y resolver la dinámica del objeto, por ejemplo utilizando el Método de Elemento Finito. El segundo modelo es el modelo de colisiones que utiliza una geometría menos fina para tener un buen balance entre precisión y rapidez durante la simulación. Debido a que el modelo de colisiones utiliza una geometría menos fina, es necesario utilizar el tercer modelo que es el modelo visual. Este modelo permite tener una representación gráfica más suave y con mucho más detalle sin comprometer el comportamiento general de la simulación.

4.1.1 Modelo Mecánico

Este modelo, también denominado modelo de deformaciones, representa las partículas, denominadas nodos de simulación, que conforman el modelo del sólido deformable como discretización del continuo. Las coordenadas de los nodos son los grados de libertad del objeto y generalmente, están gobernados por ecuaciones del tipo:

$$\mathbf{a} = \mathbf{P}\mathbf{M}^{-1} \sum_i \mathbf{f}_i(\mathbf{x}, \mathbf{v}) \quad (4.1)$$

donde, \mathbf{x} , \mathbf{v} y \mathbf{a} son los vectores de posición, velocidad y aceleración, \mathbf{f}_i son las diferentes funciones de fuerza, \mathbf{M} es la matriz de masa y \mathbf{P} es la matriz de proyección para restringir los desplazamientos mediante las condiciones de frontera.

El modelo correspondiente en SOFA es un conjunto de componentes conectados a uno nodo de escena en común donde cada componente se encarga de una tarea en específico implementadas utilizando funciones virtuales con un paradigma orientado a objetos. Cada operador de la ecuación 4.1 corresponde a un componente del nodo de la escena en cuestión.

4.1.2 Modelo de Colisiones

La biblioteca cuenta con diferentes algoritmos de detección de colisiones como el algoritmo de distancia entre pares de primitivas (triángulos y esferas), puntos en campos de distancia, distancia entre mallas en colisión utilizando rastreo de rayos, entre otros.

Se utiliza un modelo separado para vincular los modelos a la estructura de datos de los diferentes algoritmos de detección de colisiones. Además, este modelo permite tener una representación geométrica menos detallada del objeto con el objetivo de reducir el tiempo de procesamiento del algoritmo de detección de colisiones o bien, si se desea tener mayor precisión en la detección y respuesta de las colisiones, se puede incrementar la resolución de la geometría de esta superficie.

4.1.3 Modelo Visual

Uno de las ventajas más importantes de SOFA es que las mallas que se usan para la visualización pueden ser diferentes a aquellas que se utilizan para el modelo de deformaciones. Es por ello que se pueden utilizar modelos con mucho mayor detalle que los modelos mecánico o de colisiones. Además, aunque se pueden utilizar otras

4. FRAMEWORK DE SIMULACIÓN MULTIMODELO SOFA

librerías para este fin, SOFA tiene implementado una biblioteca de renderizado basada en OpenGL.

4.1.4 Mapeo entre modelos

Como se mencionó anteriormente, SOFA depende de diferentes modelos que componen la simulación, por lo que debe de existir una comunicación entre éstos. Uno de los modelos, generalmente el mecánico, actúa como maestro imponiendo desplazamientos a los esclavos, generalmente el visual y el de colisiones, utilizando lo que se denomina como mapeos. Los modelos mapeados pueden crear jerarquías de tal forma que estos a su vez sean maestros de otros con grados de libertad independientes en el nodo raíz. Los grados de libertad que son independentistas son los maestros de los modelos de colisiones basados en triángulos. Cuando los modelos de colisiones chocan, se crean pares de puntos de contacto, cada punto es un esclavo de un modelo de contacto[35].

Sea f la función que mapea las posiciones \mathbf{x}_m de un modelo maestro a las posiciones \mathbf{x}_s de un esclavo:

$$\mathbf{x}_s = f(\mathbf{x}_m) \quad (4.2)$$

De forma similar se mapean las velocidades:

$$\mathbf{V}_s = \mathbf{J}\mathbf{V}_m \quad (4.3)$$

La matriz Jacobiana $\mathbf{J} = \frac{\partial \mathbf{x}_s}{\partial \mathbf{x}_m}$ codifica la relación lineal entre las velocidades del maestro y las del esclavo. Las aceleraciones se pueden mapear como:

$$\mathbf{a}_s = \mathbf{J}\mathbf{a}_m + \frac{\partial \mathbf{J}}{\partial \mathbf{X}_m} \mathbf{V}_m \quad (4.4)$$

En mapeos lineales, los operadores f y \mathbf{J} son los mismos, ya que, de otra forma, f sería no lineal con respecto a x_m y no se podría expresar como una matriz. Para superficies embebidas en celdas deformables, la matriz \mathbf{J} contiene las coordenadas del baricentro. Para superficies añadidas a cuerpos rígidos, cada fila de la matriz codifica la relación $v = \dot{o} + \boldsymbol{\omega} \times (x - o)$ para cada vértice, donde \dot{o} es la velocidad dada por el desplazamiento del centro de masa del elemento y $\boldsymbol{\omega} \times (x - o)$ su velocidad dada su rotación.

Las velocidades y las posiciones se transmiten de arriba hacia abajo en la jerarquía, mientras que las fuerzas se propagan de abajo hacia arriba hasta llegar a los grados de libertad independientes donde se aplica la ley de Newton $\mathbf{F} = \mathbf{M}\mathbf{a}$. Dadas las fuerzas \mathbf{f}_s aplicadas al modelo esclavo, el mapeo calcula y acumula las fuerzas equivalentes \mathbf{f}_m aplicadas a su maestro. Debido a que fuerzas equivalentes deben tener la misma potencia, se mantiene la relación:

$$\mathbf{v}_m^T \mathbf{f}_m = \mathbf{v}_s^T \mathbf{f}_s \quad (4.5)$$

La relación cinemática $\mathbf{v}_s = \mathbf{J}\mathbf{v}_m$ nos permite reescribir la ecuación anterior como

$$\mathbf{v}_m^T \mathbf{f}_m = \mathbf{v}_m^T \mathbf{J}^T \mathbf{f}_s \quad (4.6)$$

Como la relación se mantiene para cualquier velocidad posible de \mathbf{v}_m , el principio de trabajo virtual nos permite simplificar la ecuación anterior para obtener:

$$\mathbf{f}_m = \mathbf{J}^T \mathbf{f}_s \quad (4.7)$$

Cuando existe más de un esclavo, cada uno de ellos contribuye con las fuerzas que se transmiten al maestro utilizando su mapeo. El arreglo jerárquico permite calcular los desplazamientos y aplicar fuerzas en todos los niveles de la estructura. En SOFA existen diferentes mapeos dependiendo de la geometría o propósito de la aplicación.

4.2 Estructura de Datos

La estructura de una escena esta organizada de manera jerárquica en cuanto al conjunto de objetos y sus modelos. Además, se puede crear una red de dependencias entre atributos de componentes.

4.2.1 Gráfica de la Escena

La estructura principal de la escena está compuesta por nodos, donde cada nodo tiene un propósito específico y son utilizados para juntar componentes con los mismos grados de libertad o la misma topología. Los grados de libertad que se conectan a través de mapeos en una estructura cinemática deben ubicarse en nodos diferentes, y el maestro debe ubicarse como padre o más arriba que el esclavo en la jerarquía

4. FRAMEWORK DE SIMULACIÓN MULTIMODELO SOFA

para asegurar que se visite primero en el recorrido de arriba hacia abajo. También es posible utilizar nodos para organizar componentes. Para aplicar un algoritmo de simulación a un conjunto de objetos, se necesita organizar los objetos en un grupo y añadir el componente a la raíz del grupo o más arriba en la jerarquía. El que haya subgrupos anidados no afecta el comportamiento de los algoritmos siempre y cuando el maestro se encuentre más arriba que los esclavos en la jerarquía. Por otra parte, la conexión entre componentes que no sean hermanos se tiene que hacer de manera explícita.

La estructura de datos se procesa utilizando lo que se denomina visitantes, quienes se encargan de utilizar funciones virtuales en cada nodo que recorren y a su vez a los componentes que contiene el nodo.

4.2.2 Contenedores de Datos y Motores

Los parámetros de cada componente se almacenan en objetos denominados contenedores de datos. Cuando se necesita calcular un parámetro a partir de otros se utilizan los motores. Los motores contienen datos de entrada y salida, en el método de actualización de estos componentes se calcula el valor de la salida a partir de los datos de entrada.

4.2.3 Geometría y Topología

La geometría de un modelo describe la ubicación espacial de los vértices de una malla, mientras que la topología de la malla indica cómo están conectados los vértices a través de aristas, triángulos o cualquier tipo de elemento de malla. Esta información es necesaria para la visualización de la malla, el modelo mecánico, la detección de colisiones, el renderizado háptico o bien para describir campos escalares o vectoriales.

Los objetos topológicos se componen de cuatro miembros: Contenedores, Modificadores, Geometría, y Algoritmos. Estos miembros se utilizan para crear, modificar los arreglos topológicos o para acceder a información geométrica o de adyacencia.

Otra ventaja de SOFA es la capacidad para hacer mapeos topológicos. Estos mapeos permiten definir una topología de malla a partir de otra utilizando los mismos grados de libertad. Esto permite crear mallas triangulares a partir de una de tetraedros, por ejemplo.

4.3 Algoritmos de Simulación

SOFA provee de implementaciones de algoritmos de solución de ecuaciones diferenciales, de sistemas lineales, de restricciones, algoritmos de detección de colisiones, e incluso componentes implementados en la unidad de procesamiento gráfico.

4.3.1 Solucionadores de Ecuaciones Diferenciales

Se requiere de estos algoritmos de solución de ecuaciones diferenciales para generar la simulación en cada paso de tiempo, ya que permiten integrar de forma numérica en el tiempo y calcular posiciones y velocidades en el siguiente paso de tiempo. SOFA posee al menos ocho solucionadores, incluyendo Euler semiimplícito, Runge-Kutta4 explícito, Euler implícito, entre otros.

Como se menciona en el apéndice A los solucionadores implícitos se tiene la ventaja de que son más estables para fuerzas de gran rigidez o grandes pasos de tiempo. Estos algoritmos requieren solucionar sistemas de ecuaciones del tipo:

$$\underbrace{(\alpha\mathbf{M} + \beta\mathbf{B} + \gamma\mathbf{K})}_A \delta\mathbf{v} = \mathbf{b} \quad (4.8)$$

donde \mathbf{M} es la matriz de masa, mientras que $\mathbf{K} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}$ y $\mathbf{B} = \frac{\partial \mathbf{f}}{\partial \mathbf{v}}$ son las matrices de rigidez y amortiguamiento respectivamente. La solución de estos sistemas de ecuaciones requiere de un solucionador de ecuaciones lineales que se describe a continuación.

4.3.2 Solucionadores de Ecuaciones Lineales

Uno de los solucionadores implementados en SOFA es el de Gradiente Conjugado que, como se describe en el capítulo A, es un algoritmo iterativo para solucionar sistemas de ecuaciones lineales como el que resulta de algoritmos implícitos de solución de ecuaciones lineales.

En SOFA también están disponibles solucionadores directos que pueden servir como preconditionadores del algoritmo de gradiente conjugado o bien para obtener la solución directamente. Por lo menos hay diez algoritmos de solución de ecuaciones lineales.

4.3.3 Solucionadores de Restricciones

SOFA también posee algoritmos basados en multiplicadores de Lagrange para manejar restricciones complejas como articulaciones y contactos entre objetos móviles. Estos algoritmos se pueden combinar con los métodos de integración explícitos o implícitos. Cada restricción depende de la posición relativa de los objetos que están interactuando así como de otros parámetros:

$$\begin{aligned}\Phi(\mathbf{X}_1, \mathbf{X}_2, \dots) &= 0 \\ \Psi(\mathbf{X}_1, \mathbf{X}_2, \dots) &\geq 0\end{aligned}\tag{4.9}$$

donde Φ representa las leyes de interacción bilateral (sujeciones, articulaciones deslizantes, etc.) y Ψ representa las leyes de interacción unilateral (contactos, fricción, etc.). Las restricciones pueden ser no lineales. Por simplicidad, a continuación se presentan las ecuaciones para la interacción entre dos objetos 1 y 2, sin embargo, la solución es válida para un número arbitrario de objetos.

Los multiplicadores de Lagrange se calculan en cada paso de tiempo y se suman como términos de fuerza a la ecuación 4.8:

$$\begin{aligned}\mathbf{A}_1 \delta \mathbf{V}_1 &= \mathbf{b}_1 + \mathbf{H}_1^T \lambda \\ \mathbf{A}_2 \delta \mathbf{V}_2 &= \mathbf{b}_2 + \mathbf{H}_2^T \lambda\end{aligned}\tag{4.10}$$

donde

$$\mathbf{H}_1 = \left[\frac{\partial \Phi}{\partial X_1}; \frac{\partial \Psi}{\partial X_1} \right] \quad \mathbf{H}_2 = \left[\frac{\partial \Phi}{\partial X_2}; \frac{\partial \Psi}{\partial X_2} \right]\tag{4.11}$$

Cuando las restricciones se aplican a un modelo mapeado, las restricciones se mapean hacia arriba en la jerarquía como fuerzas que se aplican a los grados de libertad independientes. La solución de estas restricciones se lleva a cabo a través de los siguientes pasos [35]:

1. Movimiento Libre: Los objetos que están interactuando entre sí se solucionan de manera independiente haciendo $\lambda = 0$ en la ecuación 4.10. Obteniendo así lo que se denomina movimiento libre δV_1^f y δV_2^f para cada objeto. Después de integrar se obtiene X_1^f y X_2^f .
2. Solución de la restricción: Las ecuaciones con restricciones se pueden linealizar y vincular con la dinámica del modelo.

$$\begin{bmatrix} \Phi(\mathbf{X}_1, \mathbf{X}_2) \\ \Psi(\mathbf{X}_1, \mathbf{X}_2) \end{bmatrix} = \begin{bmatrix} \Phi(\mathbf{X}_1^f, \mathbf{X}_2^f) \\ \Psi(\mathbf{X}_1^f, \mathbf{X}_2^f) \end{bmatrix} + \underbrace{h\mathbf{H}_1 \delta \mathbf{V}_1^c + h\mathbf{H}_2 \delta \mathbf{V}_2^c}_{h[\mathbf{H}_1 \mathbf{A}_1^{-1} \mathbf{H}_1^T + \mathbf{H}_2 \mathbf{A}_2^{-1} \mathbf{H}_2^T] \lambda}\tag{4.12}$$

Con $\delta\mathbf{V}^c = \delta\mathbf{V} - \delta\mathbf{V}^f$. Junto con las ecuaciones 4.9, forman un problema de complementariedad mixta que se puede solucionar con diferentes algoritmos, se calculan los valores de λ utilizando el algoritmo iterativo Gauss-Seidel que verifica continuamente que se satisfagan las restricciones.

3. Movimiento Correctivo: Cuando se tiene el valor de λ , el movimiento correctivo se calcula como:

$$\begin{aligned} \mathbf{x}_1^{t+h} &= \mathbf{x}_1^f + h\delta\mathbf{V}_1^c & \text{con } \delta\mathbf{V}_1^c &= \mathbf{A}_1^{-1}\mathbf{H}_1^T\lambda \\ \mathbf{x}_2^{t+h} &= \mathbf{x}_2^f + h\delta\mathbf{V}_2^c & \text{con } \delta\mathbf{V}_2^c &= \mathbf{A}_2^{-1}\mathbf{H}_2^T\lambda \end{aligned} \quad (4.13)$$

Un componente denominado ciclo de animación, generalmente agregado hasta arriba en la gráfica de SOFA, tiene la tarea de imponer este nuevo esquema al resto de la gráfica.

Las ecuaciones 4.12 y 4.13 requieren obtener la inversa de la matriz \mathbf{A} , denominada matriz de compliancia, y que cambia en cada paso de tiempo en un modelo no lineal. En general, el cálculo de esta inversa puede tomar mucho tiempo para una simulación en tiempo real, por lo que se utilizan diferentes estrategias como tomar solamente la diagonal de la matriz, precalcular la matriz inversa, o factorización asíncrona en la unidad de procesamiento gráfico. Estas estrategias se implementan en una categoría de componentes llamada corrección de restricciones.

4.3.4 Detección y Respuesta a Colisiones

La detección de colisiones se divide en varias fases, cada fase está implementada en un componente diferente, y está estructurada por un componente denominado CollisionPipeline. Cada objeto que puede colisionar posee una geometría asociada o mapeada de grados de libertad independientes. La fase gruesa devuelve pares de volúmenes delimitadores que pueden colisionar. Por otra parte, el componente de la fase fina devuelve pares de primitivas geométricas junto con los puntos de contacto asociados. A continuación el administrador de contactos que genera interacciones de varios tipos basados en reglas personalizables. Una de estas reglas es la repulsión que está implementada basada en penalización o con restricciones utilizando multiplicadores de Lagrange que son procesadas por los solucionadores junto con las otras fuerzas y restricciones. Para evitar colisiones entre objetos se pueden crear grupos de forma que se manejan por un solucionador común.

4. FRAMEWORK DE SIMULACIÓN MULTIMODELO SOFA

Esta biblioteca posee diversas implementaciones para la detección de colisiones. Sin embargo, una limitación es que los contactos se pueden procesar mecánicamente solamente hasta que todos los contactos hayan sido procesados por el ciclo de colisiones. Esto impide que se pueda obtener una respuesta mecánica inmediatamente después de que se detecta la colisión y así prevenir la colisión entre primitivas del mismo objeto.

4.4 Interfaz Háptica Humano Computadora

La interacción entre el usuario y la simulación interactiva es de gran interés para proporcionar mayor inmersión en el entorno de simulación. Una de las formas de interacción con la simulación se lleva a cabo a través de dispositivos hápticos. De esta forma la retroalimentación visual junto con la retroalimentación háptica contribuyen con una experiencia cada vez más apegada a un procedimiento real.

Una de las dificultades para bibliotecas como SOFA es que las fuerzas que proporcionan la retroalimentación háptica se tienen que calcular idealmente a una frecuencia de 1 kHz mientras que para la retroalimentación visual tiene que ser de al menos 30 Hz para tener la sensación de tiempo real. Otro problema es que la retroalimentación háptica puede agregar artificialmente energía en la simulación lo cual genera inestabilidades si no se tiene un control pasivo.

SOFA tiene implementación de dos técnicas de renderizado háptico. Una es el acoplamiento háptico y la otra permite el renderizado de las restricciones descritas en la sección 4.3.3.

El acoplamiento virtual es una forma de renderizado háptico en donde el usuario controla el movimiento o fuerzas del dispositivo y el éste aplica, en respuesta, fuerzas o movimientos al usuario. La mayoría de los dispositivos basados en esta técnica proponen un acoplamiento por impedancia donde la posición del dispositivo es proporcionada por la biblioteca y ésta consulta fuerzas de respuesta que proporciona la aplicación. En este acoplamiento directo la simulación se comporta como un controlador de lazo abierto.

La técnica anterior no es adecuada cuando se desea tener retroalimentación estable y robusta en un entorno virtual. Para mejorar el desempeño de esta técnica se utiliza un acoplamiento mecánico. Este acoplamiento consiste en definir una rigidez amortiguada entre la posición medida en el dispositivo y la posición simulada en el

4.4 Interfaz Háptica Humano Computadora

entorno virtual. Hay un compromiso entre la estabilidad y el desempeño por lo que se requiere de ajustar el valor rigidez del acoplamiento.

La simulación del resorte amortiguado se simula dos veces. Una vez en el ciclo háptico y otra vez en el ciclo de simulación. Si ambos ciclos están sincronizados, el resultado es el mismo. También se puede utilizar en modo asíncrono, actualización rápida del ciclo háptico y de menor velocidad en la simulación. En este caso, la retroalimentación háptica permanece estable, sin embargo el retardo entre los dos ciclos genera un amortiguamiento artificial. Si no hay un contacto en la simulación, es posible cancelar este amortiguamiento virtual. Una de las principales ventajas de esta técnica de acoplamiento virtual es que se puede utilizar en cada simulación de SOFA. Por otro lado, se tiene la desventaja de que la sensación no es la misma que la interacción que se está simulando.

La otra técnica de renderizado háptico basado en restricciones consiste en que las interacciones mecánicas se modelan utilizando mecanismos apropiados que simulan las fuerzas y movimientos deseados. Estos mecanismos se modelan como un problema basado en restricciones que se resuelve en dos hilos de procesamiento que corren a diferente frecuencia.

El primer hilo procesa la simulación general incluyendo las deformaciones de tejido suave, mientras que el segundo solamente se encarga de la interacción háptica. A través de este método se puede describir el comportamiento específico de diferentes dispositivos médicos mientras que se tiene un método unificado para la solución de la interacción mecánica entre los objetos deformables y el renderizado háptico.

4. FRAMEWORK DE SIMULACIÓN MULTIMODELO SOFA

Desarrollo del Modelo de Punción Háptica

En este capítulo se describe el modelo de punción háptica propuesto. El desarrollo de este modelo tiene la finalidad de implementarlo en entornos de simulación como los procedimientos presentados en el capítulo 6 donde se describen dos casos de estudio.

5.1 Modelo de Punción Háptica

El modelo de punción que se describe a continuación, fue desarrollado como un plugin para ser integrado en el framework de simulación SOFA. En el apéndice B se describe el procedimiento para integrar un plugin al entorno de SOFA. Este modelo se diseñó como una máquina de estados. En la figura 5.1 se muestra el esquema de la máquina de estados finitos propuesta.

En el modelo general de la máquina de estados que se describe a continuación se utilizó el algoritmo de detección de colisiones del framework SOFA, el ciclo de ejecución se mantiene en este estado mientras no se detecte una colisión. Cuando se detecta una colisión, se comprueba si la herramienta del ambiente virtual colisiona con el modelo marcado con una etiqueta definida, si se cumple esta condición, pasa al siguiente estado, si no, se vuelve a probar si hay más colisiones. Si la colisión ocurre con un modelo que coincide con la etiqueta, se comprueba si la fuerza ejercida sobre el modelo es mayor que un valor umbral, de ser así, se activan las demás fuerzas de interacción con los tejidos, si la herramienta sale de una región de punción definida

5. DESARROLLO DEL MODELO DE PUNCIÓN HÁPTICA

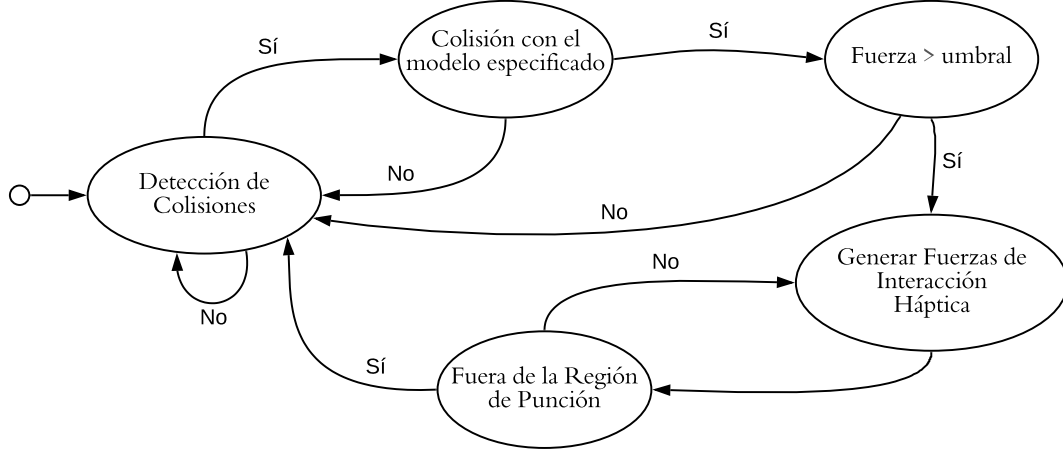


Figura 5.1: Máquina de estados del modelo de punción háptica

se reinicia el ciclo de ejecución.

En el modelo de punción propuesto se considera la suma de cinco fuerzas y la fuerza de compensación inercial del dispositivo. La dirección de punción se obtiene a partir del primer contacto de la aguja con el tejido y las fuerzas se calculan en esta dirección, denominada \mathbf{v}_1 . Estas fuerzas se renderizan en sentido contrario a la dirección del vector de inserción, es decir, hacia el dispositivo háptico, y la frecuencia de renderizado háptico ocurre a ~ 1 kHz. El modelo háptico general se representa como

$$\mathbf{f}_{punción} = \mathbf{f}_{reacción} + \mathbf{f}_{fricción} + \mathbf{f}_{dirección} + \mathbf{f}_{penetración} + \mathbf{f}_{corte} + \mathbf{g} \quad (5.1)$$

donde $\mathbf{f}_{reacción}$ es la fuerza de reacción a la resistencia del tejido durante la penetración de la aguja resultado de la solución por el método de elemento finito del modelo biomecánico. Esta fuerza en la dirección de punción está dada por

$$\mathbf{f}_{reacción} = \begin{cases} 0, & \mathbf{x}_{punta} < \mathbf{x}_1 \\ \mathbf{f}_{fem}(\mathbf{x}), & \mathbf{x}_1 \leq \mathbf{x}_{punta} \leq \mathbf{x}_2 \\ 0, & \mathbf{x}_{punta} > \mathbf{x}_3 \end{cases} \quad (5.2)$$

donde $\mathbf{f}_{fem}(\mathbf{x})$ es la fuerza de reacción de los tejidos que proviene de la respuesta de la solución del modelo biomecánico utilizando el método de elemento finito, \mathbf{x}_{punta} es la posición de la aguja, \mathbf{x}_1 es la posición del tejido sin deformación, \mathbf{x}_2 es la posición del

tejido en su máximo estado de deformación antes de la punción, y \mathbf{x}_3 es la posición del tejido después de la punción.

La fuerza $\mathbf{f}_{fricción}$ representa el modelo de fricción dado por

$$\mathbf{f}_{fricción} = \begin{cases} C_n \text{sgn}(\dot{\mathbf{x}}) + b_n \dot{\mathbf{x}} & \dot{\mathbf{x}} < -\Delta v/2 \\ \max(D_n, f_a), & -\Delta v/2 \leq -\dot{\mathbf{x}} < 0 \\ \min(D_p, f_a), & 0 \leq \dot{\mathbf{x}} < \Delta v/2 \\ C_p \text{sgn}(\dot{\mathbf{x}}) + b_p \dot{\mathbf{x}} & \dot{\mathbf{x}} \geq \Delta v/2 \end{cases} \quad (5.3)$$

donde

C_p y C_n son los valores positivo y negativo de la fricción dinámica;

b_p y b_n son los valores positivo y negativo de fricción viscosa;

$\dot{\mathbf{x}}$ es la velocidad relativa entre las superficies de contacto;

D_p y D_n son los valores positivo y negativo de la fricción estática;

Δv es el valor por debajo del cual la velocidad se consierda cero, y;

f_a es la suma de fuerzas aplicadas al sistema que no están relacionados con la fricción

La función $\text{sgn}(\dot{\mathbf{x}})$ se calcula como el producto punto entre el vector de velocidad y el vector de dirección de punción \mathbf{v}_1 . Esto implica que cuando el vector de velocidad ocurre en el sentido de la penetración, el valor es positivo, y será negativo en caso contrario. Además, para la implementación de la función que modela la fricción y con la finalidad de aumentar la estabilidad, se utilizó un filtro pasa bajas de medias móviles sobre las n últimas iteraciones para el cálculo de la velocidad $\dot{\mathbf{x}}$, ésta se calculó utilizando diferencias finitas de acuerdo con

$$\dot{\mathbf{x}}(t) = \frac{1}{n} \sum_{i=t-n}^t \dot{\mathbf{x}}(i) \quad (5.4)$$

donde $\dot{\mathbf{x}}(i) = \frac{\mathbf{x}(i) - \mathbf{x}(i-1)}{\Delta t}$.

La fuerza $\mathbf{f}_{dirección}$ representa la fuerza normal a la trayectoria de inserción de la aguja, ya que en un procedimiento real es difícil mover la aguja perpendicularmente a la trayectoria inicial. Una vez que se inserta la aguja en el tejido, ésta se mantiene restringida a seguir la trayectoria definida por la orientación que tiene la aguja al momento de la inserción

$$\mathbf{f}_{dirección} = \begin{cases} 0, & t < t_{punción} \\ k\mathbf{f}_n(t), & t \geq t_{punción} \end{cases} \quad (5.5)$$

5. DESARROLLO DEL MODELO DE PUNCIÓN HÁPTICA

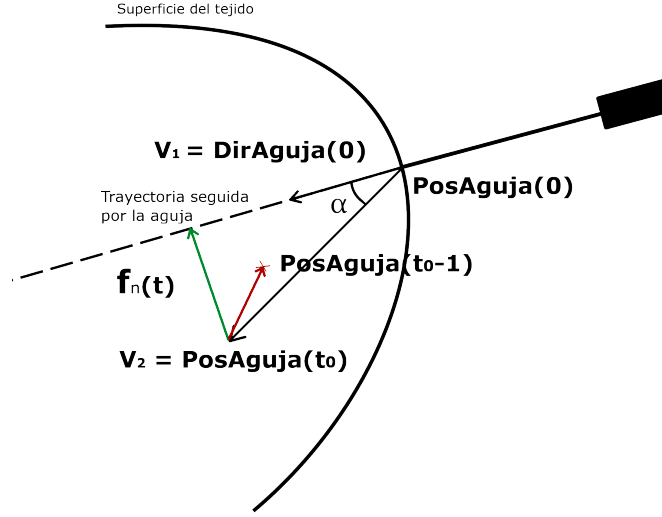


Figura 5.2: Diagrama del modelo de inserción de la aguja. Adaptado de [29]

donde $\mathbf{f}_n(t)$ es el vector perpendicular a la trayectoria de inserción inicial que se obtiene geoméricamente a partir de los vectores \mathbf{v}_1 y \mathbf{v}_2 que se muestran en la figura 5.2. Se utilizó la siguiente ecuación para obtener el vector \mathbf{f}_n en un instante de tiempo

$$\mathbf{f}_n = \mathbf{v}_1 \times \mathbf{v}_2 \times \mathbf{v}_1 \quad (5.6)$$

Con esto, la aguja se mantiene sobre la trayectoria de inserción y al desplazarse fuera de esta trayectoria se percibe una fuerza en sentido opuesto para regresar a la línea de punción. Además, en la ecuación 5.5, $t_{puncion}$ es el momento en el que ocurre la punción en el tejido. El valor k es una constante característica del tejido que modela un tejido lineal y se obtiene de manera experimental. En la figura 5.2 se muestra el diagrama del modelo que mantiene a la aguja en la trayectoria de inserción.

La fuerza $\mathbf{f}_{penetración}$ es una fuerza que limita la penetración de la aguja hasta cierta profundidad definida en la escena de simulación. Esta fuerza está expresada como

$$\mathbf{f}_{penetración} = \begin{cases} 0, & \mathbf{x} < d_1 \\ A(\mathbf{x} - d_1) + f_1, & d_1 \leq \mathbf{x} \leq d_2 \\ f_{max}, & \mathbf{x} > d_2 \end{cases} \quad (5.7)$$

donde A representa la pendiente de la recta que modela la máxima penetración en

el tejido

$$A = \frac{f_{max} - f_1}{d_2 - d_1} \quad (5.8)$$

d_1 es el valor de distancia propuesto que corresponde a la máxima distancia de penetración en el tejido, d_2 es el valor de distancia máximo de la región lineal de la fuerza que modela el tejido, y f_{max} la máxima fuerza ejercida por el dispositivo. En la figura 5.3 se muestra el la gráfica del modelo propuesto de máxima profundidad de penetración.

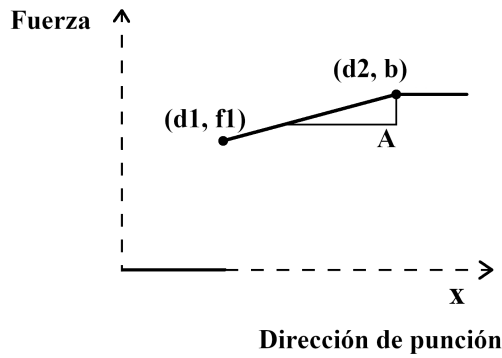


Figura 5.3: Gráfica del modelo de máxima profundidad de penetración

La fuerza \mathbf{f}_{corte} se modeló como un valor de fuerza constante característico del tejido de acuerdo con

$$\mathbf{f}_{corte} = \begin{cases} 0, & t < t_{punción} \\ f_c, & t \geq t_{punción} \end{cases} \quad (5.9)$$

donde f_c es un valor constante que se obtiene experimentalmente.

Por último, el valor de \mathbf{g} es una fuerza en sentido opuesto a la dirección de la aceleración de la gravedad en la escena de simulación y actúa como compensación inercial del dispositivo. Esta fuerza se expresa como

$$\mathbf{g} = -(m_{dispositivo} - m_{herramienta})\mathbf{g}_r \quad (5.10)$$

donde $m_{dispositivo}$ es igual a la masa aparente del dispositivo y, de acuerdo con las especificaciones del fabricante, tiene un valor de 45 gramos como se reporta en la

5. DESARROLLO DEL MODELO DE PUNCIÓN HÁPTICA

tabla 3.1, la masa $m_{herramienta}$ es la masa que se desea percibir de la herramienta virtual y \mathbf{g}_r es el vector de aceleración de la gravedad igual a $(0, -9.81, 0)$.

Finalmente, en el modelo se incluyó una función de saturación que limita las fuerzas a 2 N por seguridad del dispositivo.

Casos de Estudio

En este capítulo se muestran dos casos de estudio en donde se implementó el modelo de punción propuesto. Estos casos corresponden a la simulación de los siguientes procedimientos: a) simulación de la práctica de venopunción y b) del procedimiento de ventriculostomía.

6.1 Simulador Háptico-Virtual de Venopunción

En el caso de estudio que se describe en esta sección, se desarrolló un esquema de simulación computacional del procedimiento de venopunción a partir del modelo elástico de tejido suave, descrito en el capítulo 2, basado en la solución por el método de elemento finito implementado en el framework de simulación SOFA, en el apéndice B se describe la solución numérica del modelo; en este entorno se integró un dispositivo háptico Phantom Touch de la empresa 3D Systems y se implementó el modelo de punción háptica desarrollado.

6.1.1 Práctica de Acceso Venoso Periférico

Las técnicas de venopunción periférica es un procedimiento que se llevan a cabo de manera muy frecuente en la práctica clínica para suministrar fármacos, líquido, sangre, entre otros. Es por ello que el procedimiento de realizar una venopunción adecuada es muy importante en los programas de enfermería, medicina y en particular en especialidades como anestesiología, urgencias y cuidado crítico [36].

6. CASOS DE ESTUDIO

La canulación, o cateterización, venosa periférica que es la más común y probablemente la práctica invasiva más importante que se lleva a cabo en hospitales. En muchas ocasiones al no realizarse correctamente, este procedimiento es fuente de inconformidad para el paciente, morbilidad y mortalidad [37]. Como otros, este es un procedimiento que para garantizar calidad y trato digno para el paciente, se requiere de factores humanos como una formación adecuada, entrenamiento, supervisión y habilidad técnica.

Una de las etapas más importantes en el procedimiento de canulación venosa periférica es la venopunción. Ésta consiste esencialmente en insertar la aguja con el catéter en la vena de acceso ubicada por el practicante de la salud. Aunque el procedimiento de canulación involucra más etapas, el simulador desarrollado en este trabajo se centra en la técnica de venopunción en la zona de la fosa cubital.

6.1.2 Modelos Anatómicos e Instrumentos Virtuales

En la figura 6.1 se muestran los modelos anatómicos idealizados y las texturas de la extremidad superior, éstos fueron obtenidos de Daz3D¹ y modificados en el software de modelado y animación Blender² para adaptarlos a esta aplicación. Parte de esta modificación consiste en realizar una subdivisión del dominio para reducir la complejidad de la simulación a través de la segmentación de la región de interés que consiste en en la región de la zona de la fosa cubital. En la figura 6.1c y 6.1d se muestra la subdivisión de la región de la zona de interés.

Las vascularidades fueron modeladas en el software Blender. Como se muestra en la figura 6.2a y 6.2b, el modelo de las vascularidades se subdividió resultando en dos modelos de simulación independientes adicionales a la simulación de la piel. En la figura 6.2c se muestran los modelos de la extremidad y las vascularidades embebidas.

De igual manera se realizó el modelo del catéter que se utiliza en este procedimiento, el cual se denomina punzocat. En la figura 6.3a se muestra el modelo realizado en el software Blender.

Estos modelos corresponden a los modelos visuales utilizados en la escena de simulación en el framework SOFA.

¹<https://www.daz3d.com/>

²<https://www.blender.org/>

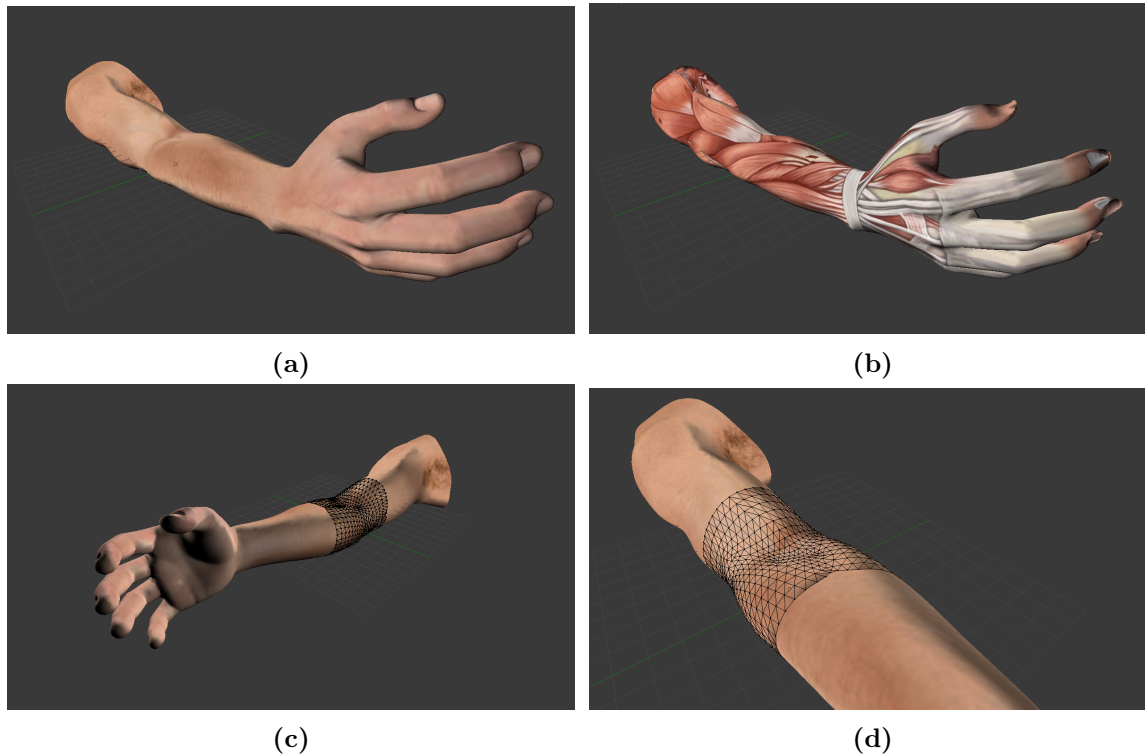


Figura 6.1: Modelos anatómicos de la extremidad superior derecha. (a) Modelo visual de la piel. (b) Modelo visual del tejido muscular. (c), (d) Detalle del segmento simulado correspondiente a la malla de deformaciones.

6.1.3 Modelo de Simulación

Para la simulación, se eligió el modelo biomecánico elástico lineal del segmento de la zona cubital y de las vascularidades a simular, se asume que el modelo es isotrópico, homogéneo y lineal; y que se tiene una masa uniforme. La discretización se realizó utilizando el método de elemento finito y la solución del modelo dinámico se realizó con el método de Euler implícito.

De la malla triangular del modelo, se realizó un proceso que consiste en decimar el número de vértices para obtener un menor número de elementos después de la generación de la malla tridimensional compuesta por tetraedros. En la figura 6.4 se muestra el resultado del proceso de decimar la malla triangular. La generación de la malla tridimensional del segmento de brazo se realizó utilizando el software Gmsh³

³<http://gmsh.info/>

6. CASOS DE ESTUDIO

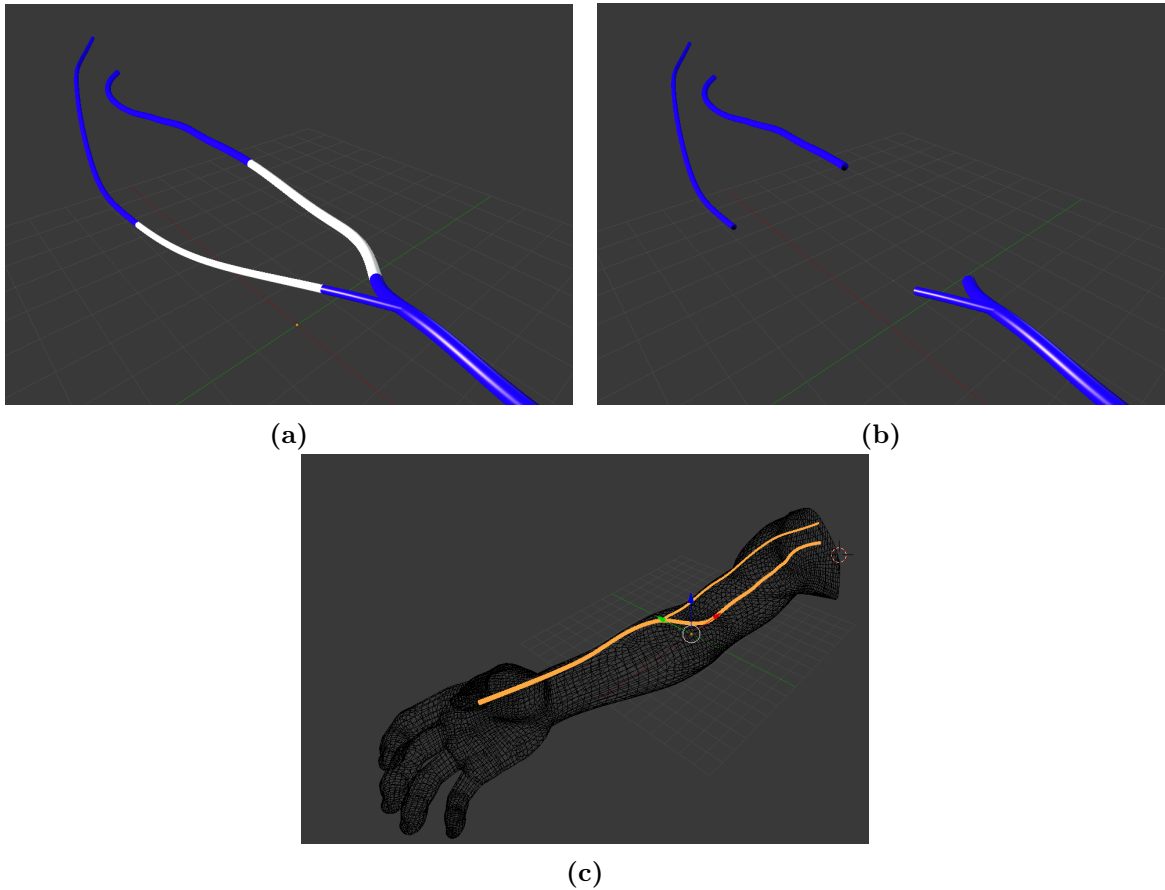


Figura 6.2: Modelos vasculares. (a), (b) Segmentos vasculares de la simulación. (c) Modelo embebido de las vascularidades.

[38]. En la figura 6.4 se muestra el resultado de la malla generada en Gmsh. Es importante considerar que el framework SOFA requiere que el archivo generado esté en el formato de la versión 2 de Gmsh y en formato ASCII.

Por otra parte, las mallas tridimensionales para las vascularidades se generaron mediante el framework SOFA en el que se define el número de nodos a generar. En este caso los elementos utilizados son hexaedros que se aproximan a la geometría del modelo definido.

Para generar la escena se requiere adicionalmente definir qué mallas se utilizarán para integrar la escena de simulación, por lo tanto, las mallas visuales utilizadas

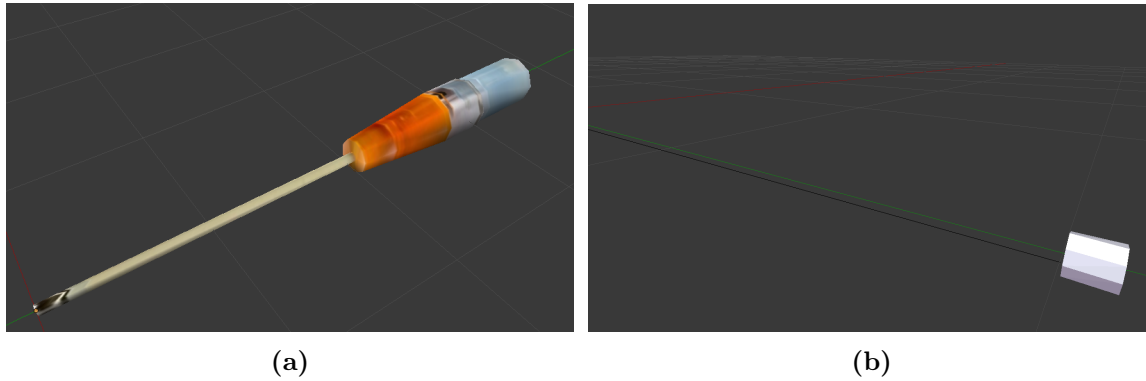


Figura 6.3: Modelo visual y de colisiones del catéter. (a) Modelo virtual del catéter intravenoso. (b) Malla de colisiones asociada.

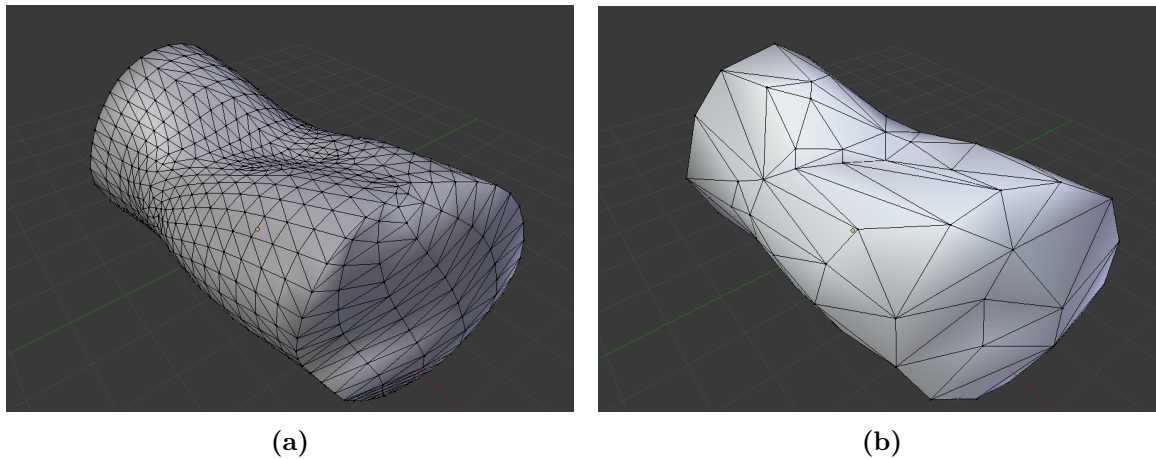


Figura 6.4: Resultado del proceso de decimar la malla triangular. (a) Malla antes de la decimación. (b) Malla después del proceso de decimación.

fueron las que se muestran en las figuras 6.1, 6.2a, 6.3a para la simulación de la extremidad superior, las vascularidades y el instrumento virtual, respectivamente; para la malla de colisiones de la piel, se utilizó la malla mostrada en la figura 6.4a. Finalmente, para la simulación mecánica del brazo se utilizó la malla tridimensional mostrada en la figura 6.5.

Como referencia para el practicante, en la esquina inferior izquierda se incorporó un indicador del ángulo del instrumento obtenido a partir del dispositivo háptico. Además, se incorporó otro estado al modelo de punción háptico de la figura 5.1 para

6. CASOS DE ESTUDIO

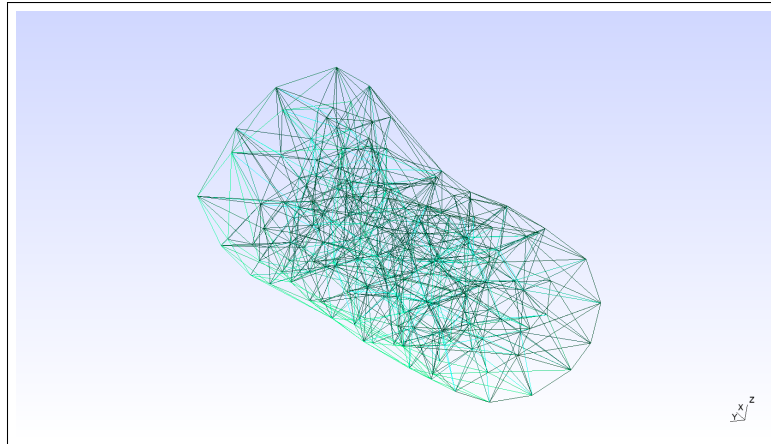


Figura 6.5: Generación de la malla tridimensional en Gsmh.

incorporar la punción vascular. Con esto, se añade el estado de inserción de la aguja a la vascularidad, en donde al insertar la aguja en la vena, se muestra un punto rojo en el catéter virtual simulando la sangre que se deposita en el instrumento durante la práctica real. Otro elemento incorporado para obtener mayor referencia es una línea guía que parte de la herramienta para proporcionar mayor referencia en cuanto a la posición de la aguja con respecto al modelo del brazo.

Por último, se incorporó la funcionalidad de almacenar y exportar datos de la simulación, como son, la posición y fuerzas que ocurren durante la simulación para, posteriormente, hacer un análisis de los datos obtenidos.

6.1.4 Resultados

Una vez integrados los modelos y creada la escena de simulación con los elementos necesarios, se obtuvo el escenario virtual mostrado en la figura 6.6. En esta figura se observan los modelos visuales del brazo, con las texturas de la piel y de estructuras musculares en transparencia, así como la estructura ósea puramente visual. También se observan las estructuras vasculares embebidas en el modelo del brazo. En la esquina inferior izquierda se muestra el ángulo del instrumento con respecto al plano horizontal que corresponde al plano X-Z. Por último, se observa la simulación del catéter con la línea guía de referencia.

Los resultados de la simulación se muestran en las figuras mostradas en esta sección. En la figura 6.7a se muestra la malla de colisiones utilizada en la simulación.

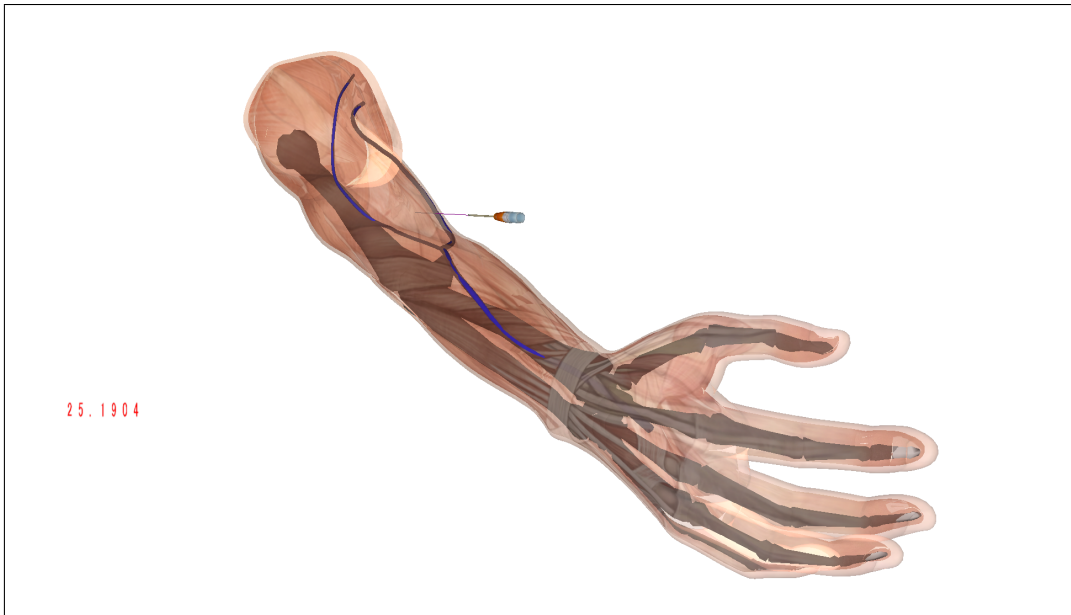


Figura 6.6: Escena de simulación de la práctica de venopunción.

Mientras que en la figura 6.7b se muestra la malla de simulación de deformaciones compuesta por tetraedros, en la figura 6.7c se muestra en detalle la malla de simulación de la sección del brazo. En la figura 6.7d se muestra la malla de simulación que aproxima la estructura de la malla de las vascularidades simuladas.

La interacción con la vena se observa en la figura 6.8, cuando se inserta la aguja en la vascularidad se muestra un punto rojo en el catéter que simula la sangre.

Las gráficas que se muestran en esta sección son resultado de una prueba del simulador de venopunción. En la figura 6.9 se muestran las fuerzas generadas en cada eje coordenado por el dispositivo háptico durante la simulación de punción de una vascularidad. En la figura 6.10 se muestra la posición por cada eje coordenado que registra el dispositivo háptico durante la simulación.

Para obtener una mejor visualización de las gráficas de trayectoria se realizó una rotación de 90 grados alrededor del eje X. En la figura 6.11 se muestra la trayectoria realizada por el usuario durante una prueba del simulador, con rojo se muestra el vector de punción a partir de la primera colisión con del modelo del brazo, y con marcadores azules se graficó la ubicación del modelo virtual de la vascularidad en

6. CASOS DE ESTUDIO

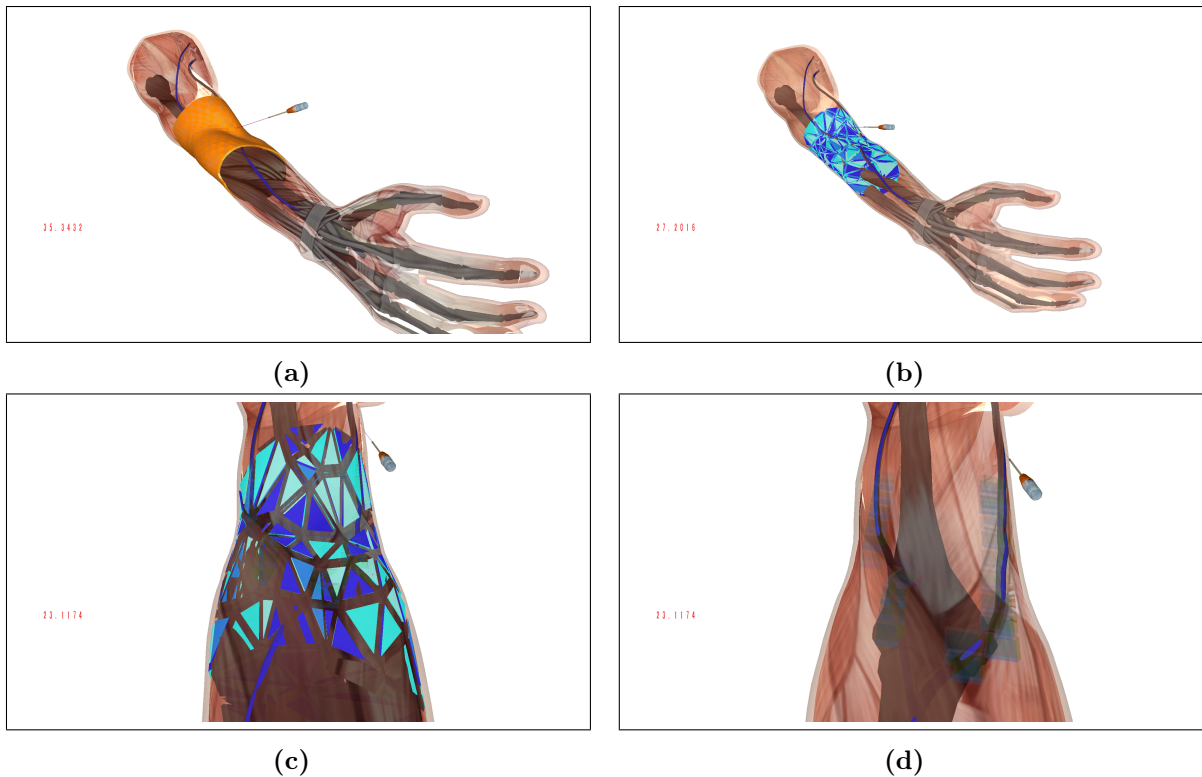


Figura 6.7: Modelos anatómicos y de simulación de la extremidad superior derecha. En la figura (a) se muestra el modelo de colisiones utilizado en la simulación. En las figuras (b) y (c) se muestra el modelo de deformaciones del segmento del brazo. En la figura (d) se muestra el modelo de deformaciones de las vascularidades.

donde se realizó la punción durante un tiempo definido, ya que esta posición cambia de acuerdo a la interacción del usuario con el entorno virtual.

Es posible exportar los datos de fuerza generados por el dispositivo háptico para realizar un análisis posterior. En la figura 6.12 se muestra la fuerza generada por el dispositivo háptico en la trayectoria de inserción de la aguja en el modelo virtual. La fuerza que se grafica corresponde a la que se opone al movimiento que realiza el usuario durante la inserción o extracción de la aguja.

En las tablas 6.1 y 6.2 que se muestran en esta sección se resumen los parámetros utilizados en la simulación. En la tabla 6.1 se muestra el resumen de los parámetros utilizados en el modelo de punción propuesto. Por último, en la tabla 6.2 se mues-

6.1 Simulador Háptico-Virtual de Venopunción

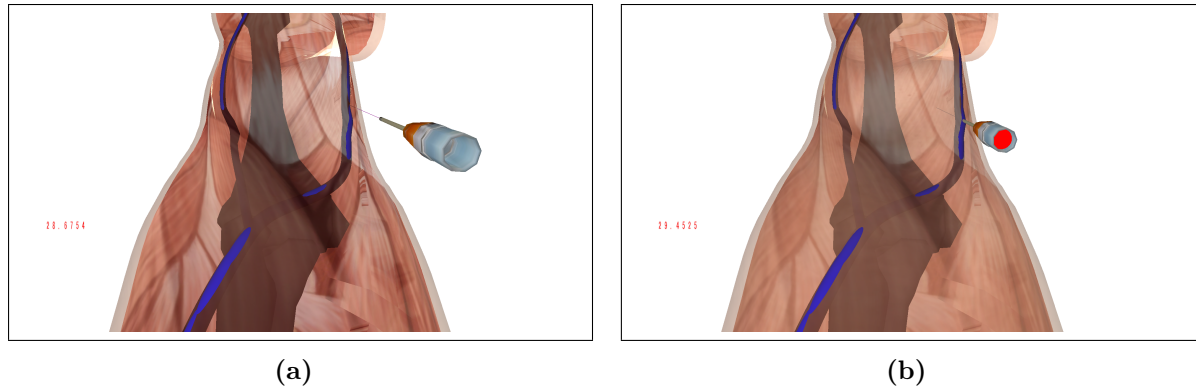


Figura 6.8: Representación de la simulación de punción vascular. En la figura (a) se muestra el catéter vascular antes de la punción. En la figura (b) se muestra con una esfera roja la representación de la sangre en el catéter al ingresar a la vascularidad.

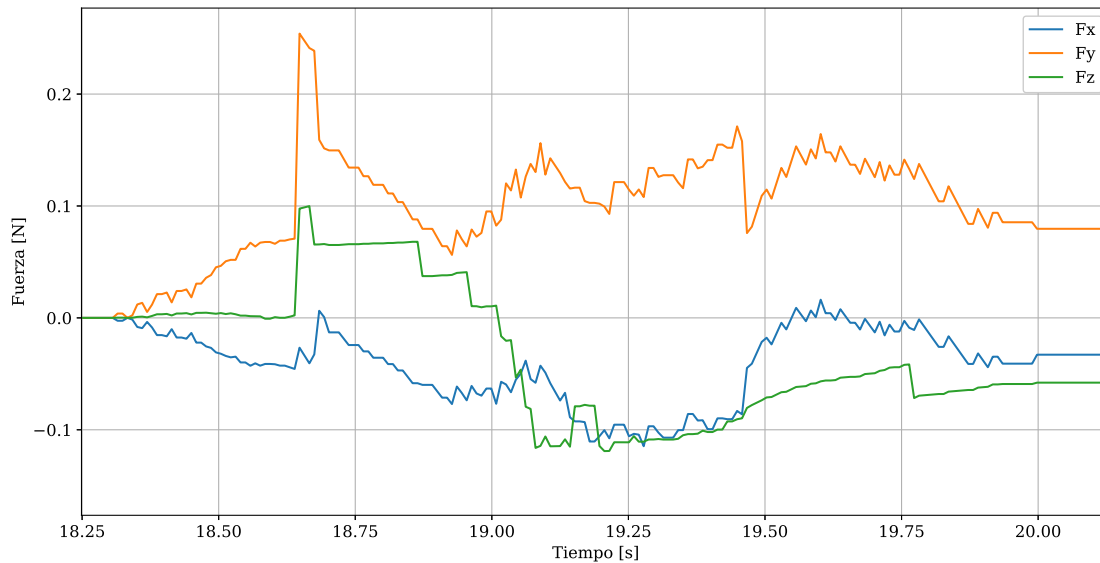


Figura 6.9: Fuerza generada por el dispositivo háptico durante la punción en la simulación.

tran los parámetros utilizados en la simulación, los valores se obtuvieron de manera experimental.

6. CASOS DE ESTUDIO

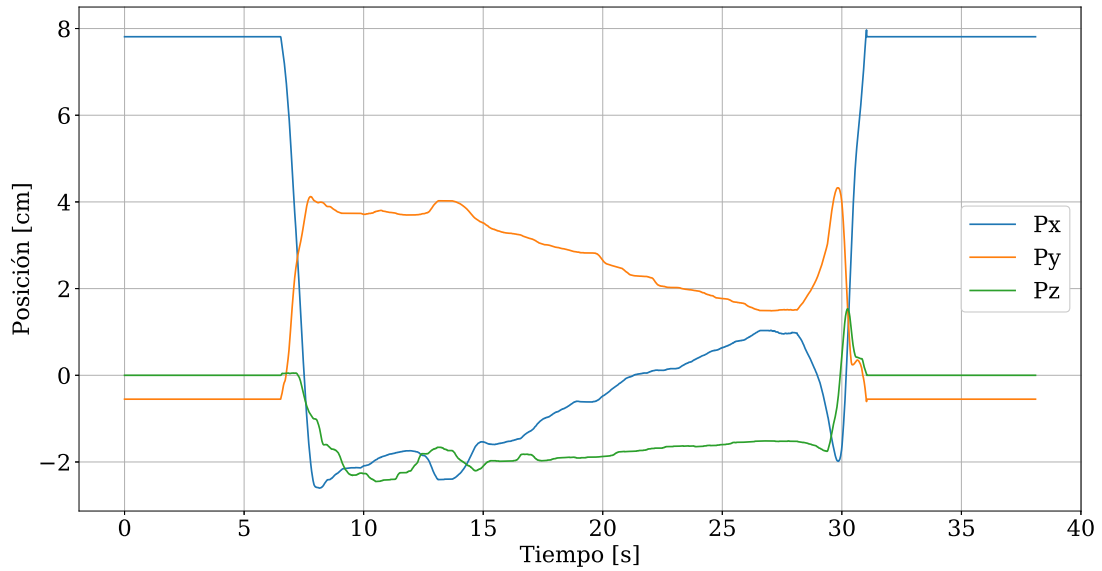


Figura 6.10: Posición registrada por el dispositivo háptico durante la simulación.

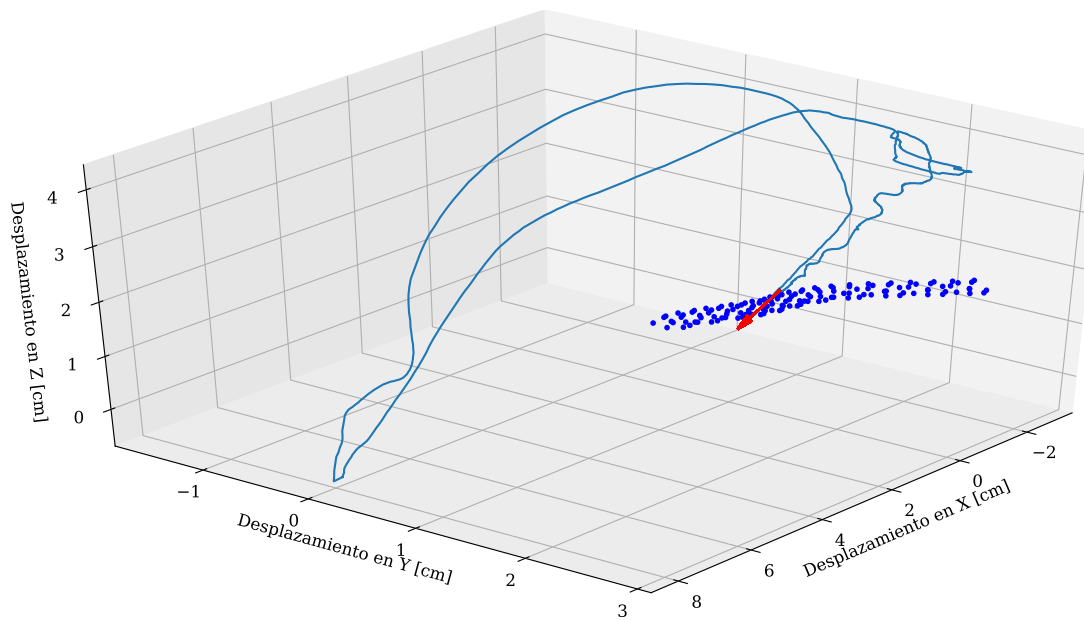


Figura 6.11: Gráfica de la trayectoria registrada por el dispositivo háptico durante la simulación. En rojo se muestra el vector de punción durante la simulación y con marcadores azules se muestra la vascularidad en un instante de tiempo.

6.1 Simulador Háptico-Virtual de Venopunción

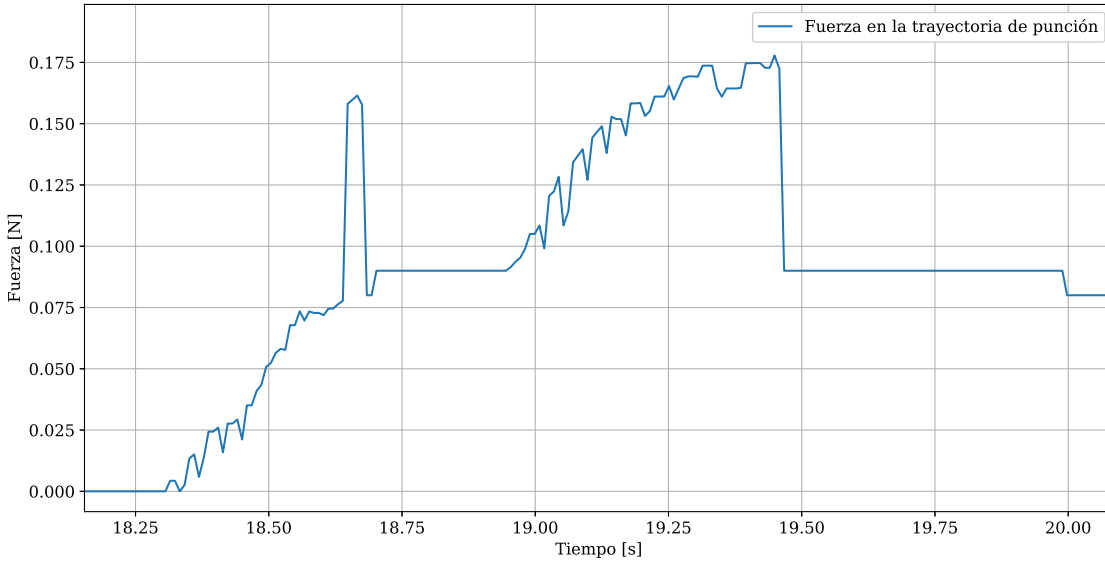


Figura 6.12: Fuerza generada por el dispositivo háptico en la dirección de punción en la fase de inserción.

Parámetro	Valor
Umbral de punción de la piel	0.3 N
Umbral de punción de la vena	0.1 N
Peso virtual del instrumento	1 g
k	5.0
C_p	0.08 N
b_p	9.0 N
D_p	0.01 N
Δv	0.01 m/s
Fuerza de corte (f_c)	0.08 N
Profundidad de penetración (d_1)	2 cm
f_{max}	2 N
f_1	0.2 N
d_2	2.3 cm

Tabla 6.1: Parámetros utilizados en el modelo de punción propuesto.

Evaluación del Sistema de Simulación

Con la finalidad de probar el simulador propuesto en cuanto a su operatividad, realismo y para determinar hasta qué punto el simulador es factible de ser utiliza-

6. CASOS DE ESTUDIO

Parámetro	Valor
Paso de tiempo de simulación	0.02 s
Número de tetraedros del brazo	922
Módulo de Young de la piel	0.05 MPa
Relación de Poisson de la piel	0.45
Módulo de Young de las vascularidades	0.25 MPa
Relación de Poisson de las vascularidades	0.45
Masa de Rayleigh	0.1
Rigidez de Rayleigh	0.1
Masa Uniforme	1 kg

Tabla 6.2: Parámetros utilizados en la escena de simulación.

do como una herramienta de entrenamiento, se realizó una evaluación mediante un cuestionario. La evaluación se realizó a 5 participantes pasantes de la Facultad de Medicina de la Universidad Nacional Autónoma de México.

Esta evaluación se llevó a cabo en tres etapas. La primera etapa consistió en explicar el funcionamiento del sistema y permitir al participante familiarizarse con el ambiente virtual, su visualización y la sensación háptica. En la segunda etapa se le solicitó al participante realizar una serie de experimentos de venopunción, en la cual se registraron sus movimientos. Finalmente, en la tercera etapa se le solicitó llenar un cuestionario en donde se les pidió a los participantes marcar el recuadro correspondiente de acuerdo con su experiencia con el simulador, y también se les solicitó escribir comentarios sobre la misma.

La evaluación de los resultados del cuestionario se realizó utilizando la escala de Likert en donde el participante responde de acuerdo con la siguiente escala: en desacuerdo completamente, 1; en desacuerdo, 2; indeciso(a), 3; de acuerdo, 4; de acuerdo completamente, 5. Los resultados del cuestionario se muestran en la tabla 6.3.

Por otra parte, entre los principales comentarios que se reportaron fueron que el simulador es una muy buena herramienta que sí podría ser utilizada para la práctica de la técnica. Entre las mejoras que se sugieren, se menciona que la posición de la pantalla con relación a la plumilla del dispositivo háptico no permite tomar la posición anatómica de la región, por lo que sería conveniente incorporar un modelo físico y que se incorporaran más capas de tejido. Otro elemento que sería conveniente

6.1 Simulador Háptico-Virtual de Venopunción

Pregunta	Resultado
Operatividad	
1. La inserción del catéter fue difícil de realizar.	3.0 ± 1.581139
2. El movimiento del instrumento es consistente con mis acciones.	4.2 ± 1.30384
3. La orientación del instrumento es intuitivo.	3.2 ± 1.643168
4. Es difícil posicionar el instrumento en el lugar adecuado.	2.8 ± 1.643168
5. La ayuda visual fue útil.	4.2 ± 0.83666
6. Fue fácil colocar el catéter en la ubicación correcta.	4.8 ± 0.447214
7. El tejido suave es fácil de puncionar.	4.4 ± 0.547723
8. La manipulación del instrumento es confusa.	1.6 ± 0.547723
9. El movimiento de la herramienta virtual es preciso.	4.2 ± 0.83666
Realismo	
10. Pude sentir la retroalimentación de fuerzas debido a la colisión entre el instrumento y los tejidos.	4.4 ± 0.894427
11. La imagen mostrada es realista.	4.4 ± 0.894427
12. La retroalimentación de fuerza debido a la interacción con los tejidos es perceptible.	4.2 ± 1.30384
13. No pude percibir la retroalimentación de fuerza.	2.6 ± 1.516575
14. Es útil como una herramienta de entrenamiento.	4.4 ± 0.894427
15. El comportamiento de los tejidos suaves es el esperado durante la interacción.	3.8 ± 1.30384
16. La retroalimentación de fuerza es útil a la hora de tocar los tejidos.	4.4 ± 0.894427
17. La experiencia de la simulación es realista.	4.8 ± 0.447214

En desacuerdo completamente, 1; en desacuerdo, 2; indeciso(a), 3; de acuerdo, 4; de acuerdo completamente, 5.

Tabla 6.3: Resultados del cuestionario de evaluación.

incorporar en la simulación, es la de generar una retroalimentación a través de fluido sanguíneo cuando la aguja traspase la vena por ambos lados, esto ocurre cuando se realiza el procedimiento de manera incorrecta.

Los resultados obtenidos de la evaluación fueron altamente prometedores de acuerdo con lo reportado en los cuestionarios y comentarios realizados por los participantes. Del análisis de operatividad se reporta que los movimientos son intuitivos, consistentes y el sistema es fácil de manipular. Además, el uso del ratón de la computadora resultó conveniente para modificar la visualización del ambiente y así obtener una percepción tridimensional del entorno. En cuanto al realismo, se reportó que la percepción de fuerzas es notable y permite percibir la experiencia de interacción con los tejidos; sin embargo, se mencionó que es conveniente adecuar los parámetros del modelo de punción para aproximar aún más la experiencia a la sensación real. De manera general, se reporta que la experiencia de la simulación es realista y que puede ser útil como una herramienta de entrenamiento. Por otro lado, se puede mencionar que la experiencia se vería mejorada al incorporar un dispositivo de realidad virtual para incrementar la percepción de profundidad mediante la visión estereoscópica.

6.2 Simulador Háptico-Virtual de Ventriculostomía

A continuación se describe el siguiente caso de estudio que consiste en un esquema de simulación de la inserción de un catéter en el procedimiento de drenaje ventricular externo o ventriculostomía. De igual manera, se implementó el modelo de punción desarrollado y se realizó la simulación de los tejidos a partir de la solución del modelo biomecánico elástico lineal por el método de elemento finito en el framework de simulación SOFA. Los modelos anatómicos fueron obtenidos a partir de la segmentación de imágenes de tomografía y se modeló el instrumento de punción.

6.2.1 Inserción de Catéter Ventricular

El drenaje ventricular externo es un procedimiento de emergencia cuya finalidad es reducir la presión intracraneal resultado de la obstrucción del flujo del líquido cefalorraquídeo [39]. La inserción de un drenaje ventricular externo es uno de los procedimientos que se lleva a cabo en la práctica de neurocirugía, y es éste procedimiento uno de los más comunes y que permite salvar más vidas en la unidad de cuidados intensivos de neurología [40].

La inserción de catéter ventricular es un procedimiento que se lleva a cabo como parte del drenaje ventricular externo. La técnica a manos libres utilizando referencias anatómicas es una técnica comúnmente utilizada por neurocirujanos para la colocación del catéter [41]. El sitio de inserción preferido es el hemisferio cerebral frontal derecho, ya que es una región no dominante para la función del lenguaje en más del 90 % de los pacientes [41]. El paciente se coloca en supinación con la cabeza elevada a 45°. El cabello se remueve y se prepara el cráneo de forma estéril. Se hace una perforación en el cráneo (trepano) ubicado en el punto de Kocher para evitar el seno sagital superior y la corteza motora del lóbulo frontal[41]. Este punto se ubica dibujando una línea medial a partir del nasión, (nasión, es el punto que intersecta el hueso frontal y dos huesos nasales del cráneo) 10 cm hacia atrás y de éste punto, 3 cm de forma lateral, a lo largo de la línea ipsolateral mediopupilar [40].

Una vez que se tiene ubicado el lugar del trepano se inyecta anestesia en la región, se hace una incisión lineal hasta alcanzar el hueso y se raspa el periostio. Se utiliza un taladro quirúrgico para penetrar el cráneo en la trayectoria determinada para la canulación ventricular, la piamadre y duramadre se perforan utilizando un escalpelo. El catéter ventricular se inserta no más de 7 cm, dirigido en el plano coronal hacia el canto medial ipsilateral del ojo y en el plano anteroposterior hacia un punto 1.5

cm anterior del tragus ipsilateral, hacia el foramen de Monro ipsilateral [40]. Una vez que se visualiza flujo del líquido cefalorraquídeo al remover el estilete, se puede medir la presión intracraneal. A continuación, el catéter se coloca a través de la piel y sale por una incisión separada, se asegura en su lugar mediante sutura, y después se conecta al sistema de drenaje externo [40].

En este caso de estudio se simula la inserción del catéter ventricular en el foramen de Monro y únicamente se simula el tejido ventricular.

6.2.2 Modelos Anatómicos e Instrumentos Virtuales

Los modelos anatómicos fueron obtenidos a partir de la segmentación tridimensional de imágenes de tomografía tomadas de un caso clínico facilitado por el Instituto Nacional de Neurología y Neurocirugía. La segmentación se realizó en el software 3D Slicer⁴, y se realizó un procesamiento de las mallas obtenidas de acuerdo con la metodología descrita por Teodoro et al. [12]. En la figura 6.13 se muestran las imágenes de tomografía y su visualización tridimensional en el software 3D Slicer.

Una vez realizado el proceso de segmentación y preprocesamiento de la malla, en el software Blender se realizó la perforación de la malla del cráneo como se muestra en la figura 6.14b, la perforación se realizó de acuerdo con las medidas mencionadas en la sección 6.2.1 para la realización del procedimiento de ventriculostomía.

El instrumento virtual utilizado consiste en la simulación de un catéter ventricular. El catéter ventricular mostrado en la figura 6.15 fue diseñado en el software Inventor de la empresa Autodesk⁵. Además, en el mismo software se diseñó una herramienta para acoplar un catéter real en el dispositivo háptico Phantom Touch y se manufacturó con filamento de ácido poliláctico (PLA) mediante el proceso de modelado por deposición fundida, comúnmente conocido como impresión 3D. En la figura 6.16 se muestra el diseño del acople para el catéter ventricular.

6.2.3 Modelo de Simulación

Como se mencionó previamente, un esquema general de simulación en el framework SOFA requiere de tres modelos principales para representar la escena de simulación, estos son: el modelo visual, modelo mecánico y el modelo de colisiones.

⁴<https://www.slicer.org/>

⁵<https://www.autodesk.com/products/inventor/overview>

6. CASOS DE ESTUDIO

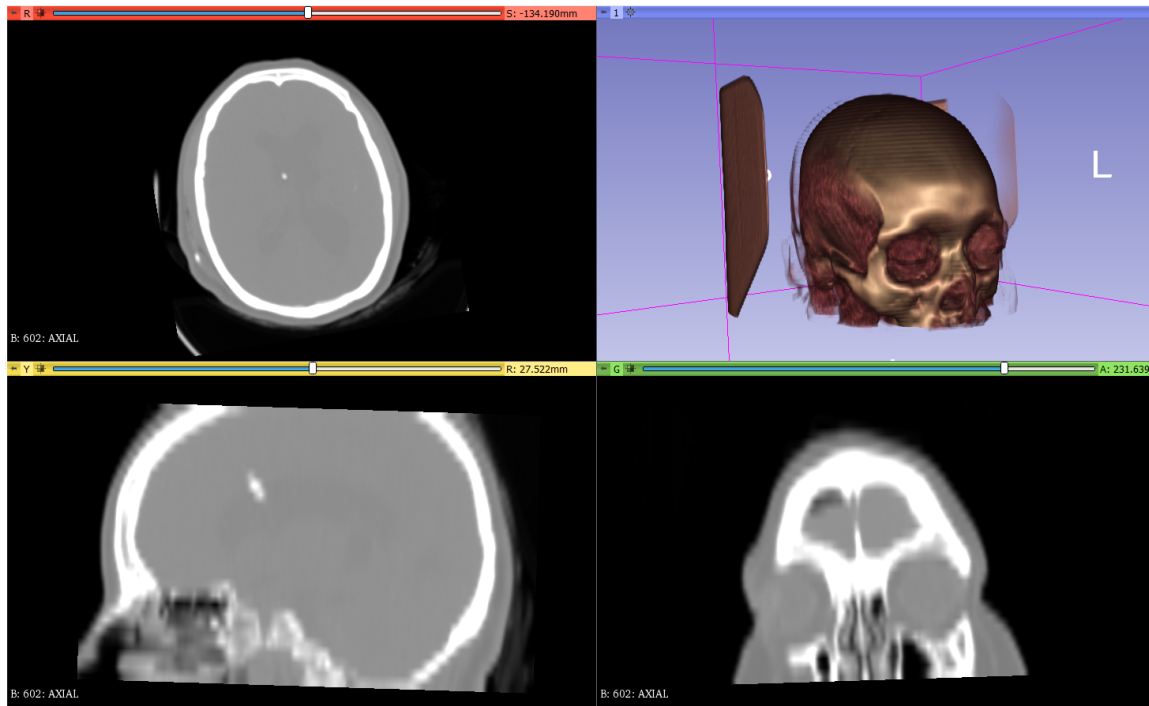


Figura 6.13: Software de segmentación de estructuras anatómicas a partir de imágenes de tomografía tomadas de un caso clínico.

A partir de los modelos obtenidos, se generó la escena de simulación en el framework SOFA y en esta sección se describen los modelos utilizados en el caso de estudio presentado.

Modelo Visual

Los modelos visuales utilizados en esta simulación corresponden a los modelos obtenidos de la segmentación presentados en la sección 6.2.2. En la figura 6.14a se muestra el modelo visual que simula la piel del paciente, en la cual se implementó un material para tener una mejor apariencia del tejido cutáneo; en la figura 6.14b se muestra el modelo que simula el cráneo del paciente, en éste se mapeó una textura para dar la apariencia del tejido óseo, las suturas del cráneo también fueron modeladas en Blender; éstos modelos no tienen ningún mapeo de deformaciones, por lo que son modelo estáticos, mientras que el modelo del tejido ventricular de la figura 6.14c se mapea con el modelo de colisiones y a su vez, con el modelo mecánico para simular la deformación. Asimismo, en la figura 6.15 se muestra el modelo del catéter

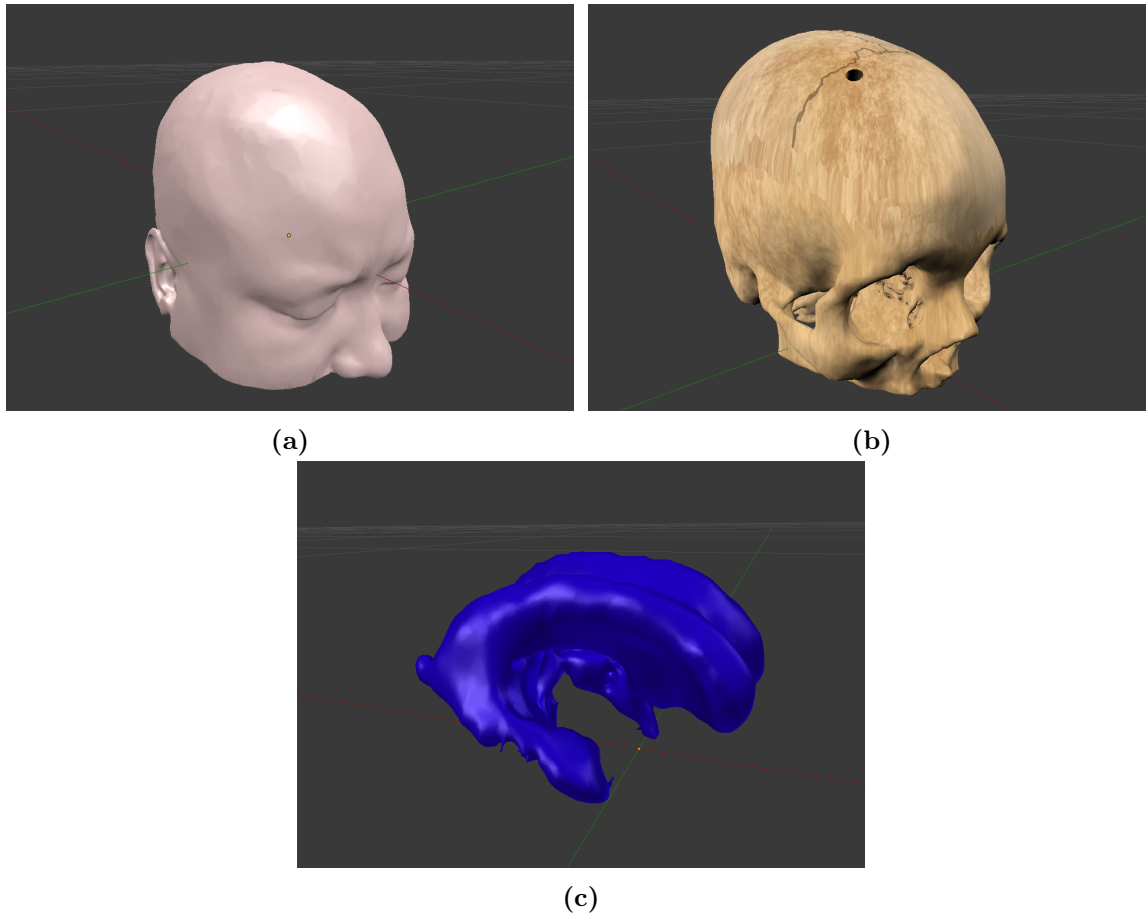


Figura 6.14: Resultado de la segmentación y preprocesamiento de las estructuras anatómicas obtenidas a partir de imágenes de tomografía. (a) Modelo visual de la piel. (b) Modelo visual del tejido óseo. (c) Modelo visual de las estructuras ventriculares.

ventricular que se acopló con el modelo de colisiones mostrado en la figura 6.3b y que está vinculado con el dispositivo háptico para la interacción con el entorno de simulación.

Otra característica que se implementó en la simulación, fue incorporar como modelos visuales tres líneas de referencia o líneas guía que sirven como referencia para los practicantes a la hora de llevar a cabo la punción. La ubicación de las líneas de referencia se obtuvo de acuerdo con la retroalimentación proporcionada por neurocirujanos. Adicionalmente, se agregó una línea que se proyecta a partir de la punta del instrumento virtual con la finalidad de proporcionar al usuario de mayor referencia

6. CASOS DE ESTUDIO

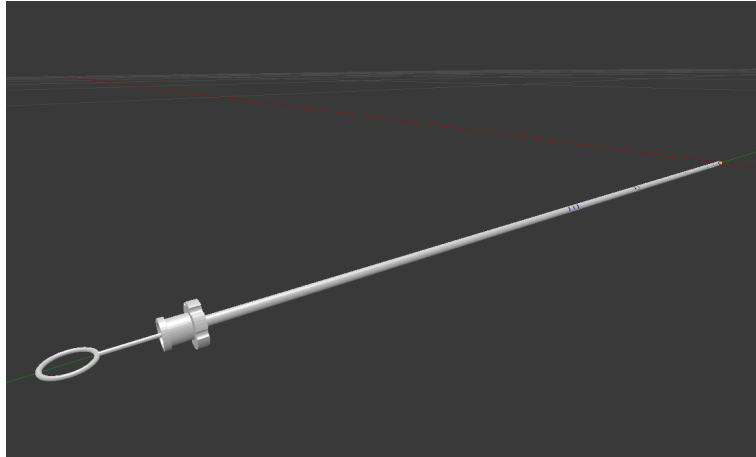


Figura 6.15: Catéter ventricular

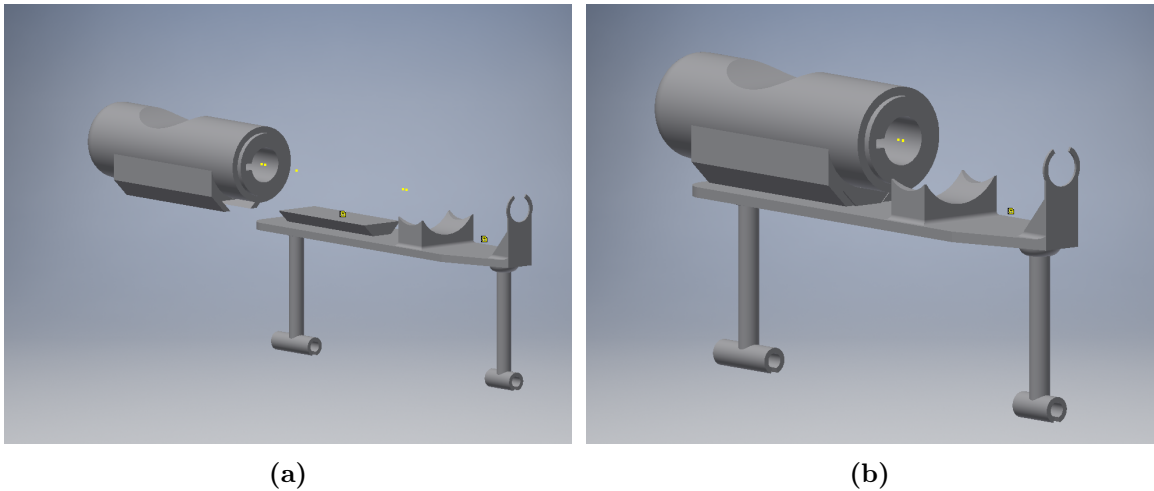


Figura 6.16: Diseño de acople de catéter ventricular para el dispositivo háptico Phantom Touch. (a) Modelo desmontable de dos piezas. (b) Modelo ensamblado.

de la posición con respecto al ambiente virtual.

Modelo Mecánico

El modelo biomecánico utilizado para la simulación de los tejidos suaves fue el modelo elástico lineal. Se asume que el modelo cumple con las características de isotropía, homogeneidad y linealidad, por lo que se utilizó el modelo de la solución presentado en el apéndice A y se consideró una masa uniforme. Se utilizó la imple-

6.2 Simulador Háptico-Virtual de Ventriculostomía

mentación de SOFA del método de elemento finito y para la solución temporal se utilizó el método de Euler implícito, descrito en la sección A.2.1.

Además, se utilizó el componente *SparseGridRamification* del framework SOFA para la generación de la malla tridimensional, ésta es generada mediante hexaedros que aproximan la geometría de la malla de los ventrículos. La visualización del modelo mecánico se muestra en la figura 6.17.

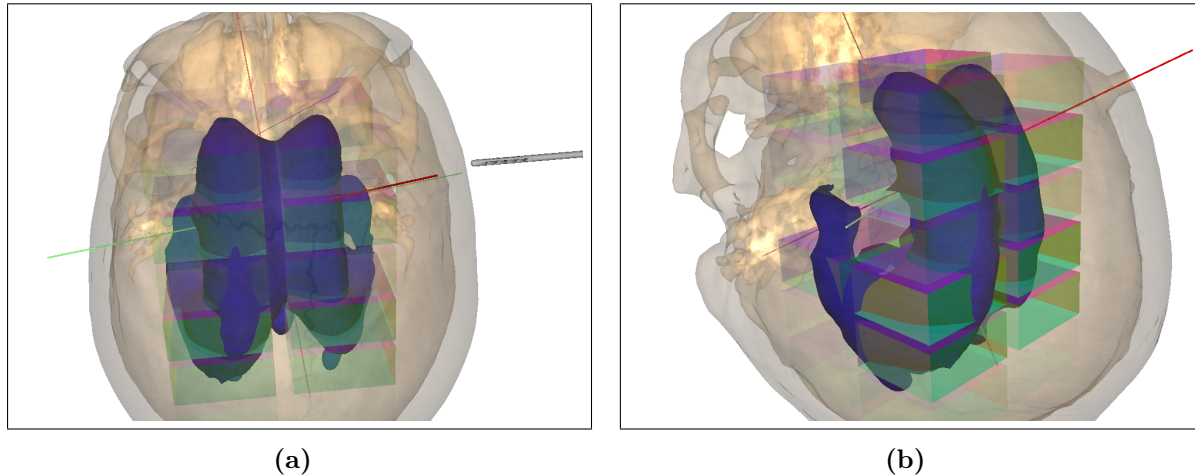


Figura 6.17: Visualización de la malla de hexaedros en el framework SOFA. (a) Visualización en el plano transversal. (b) Detalle de la malla de hexaedros.

Modelo de Colisiones

La malla de colisiones utilizada para la interacción con el tejido ventricular fue la misma que la malla de visualización y se muestra en la figura 6.18. Por otro lado, la malla de colisiones del catéter fue la misma que la utilizada en el caso anterior y se muestra en la figura 6.3b.

6.2.4 Resultados

Una vez que se tienen los modelos y se ha creado la escena de simulación, en la figura 6.19 se muestra el resultado del entorno de simulación desarrollado en el que se incorporó el modelo de punción propuesto. En las figuras 6.19a y 6.19b se muestra la escena de simulación como lo vería el practicante, se incluyen las líneas que sirven

6. CASOS DE ESTUDIO

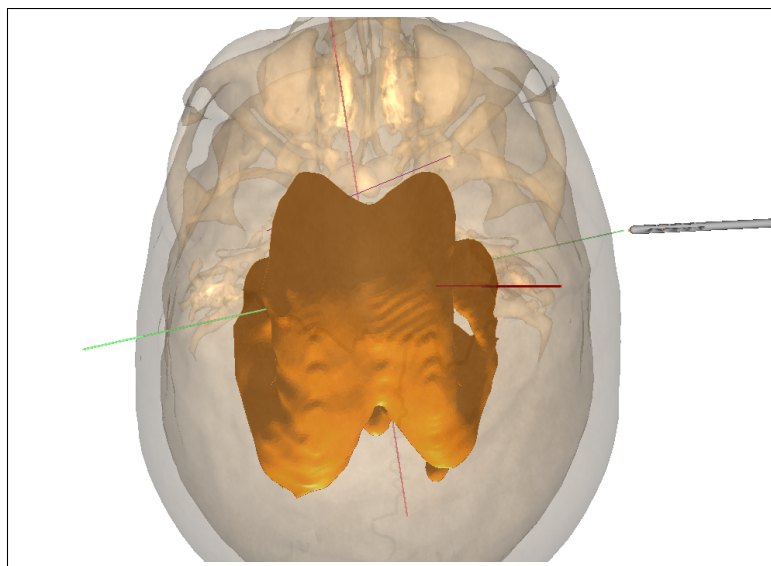


Figura 6.18: Visualización de la malla de colisiones utilizadas.

como referencia para determinar la orientación adecuada del instrumento. Además, es posible modificar el material de las mallas para visualizar los modelos en transparencia como se muestra en las figuras 6.19c y 6.19d, ésto con la finalidad de ofrecer al practicante una herramienta que le permita tener una visualización con mayor referencia sobre las estructuras anatómicas que podrían servir para complementar el entrenamiento de los especialistas.

Por otro lado, en la figura 6.20 se muestran las líneas guía que se incorporaron al ambiente de simulación, así como de la línea de referencia del instrumento virtual de acuerdo a la asesoría de un neurocirujano experto.

Con la finalidad de proporcionar al practicante una experiencia táctil aún más cercana a la realidad, en la figura 6.21 se muestra el montaje de la pieza manufacturada para acoplar el catéter ventricular que se utiliza en la práctica clínica.

Como se mencionó anteriormente, se requieren definir diversos parámetros para la ejecución de la simulación. En la tabla 6.4 se resumen los parámetros utilizados en el modelo de punción y en la tabla 6.5 se muestran los parámetros utilizados en la escena de simulación.

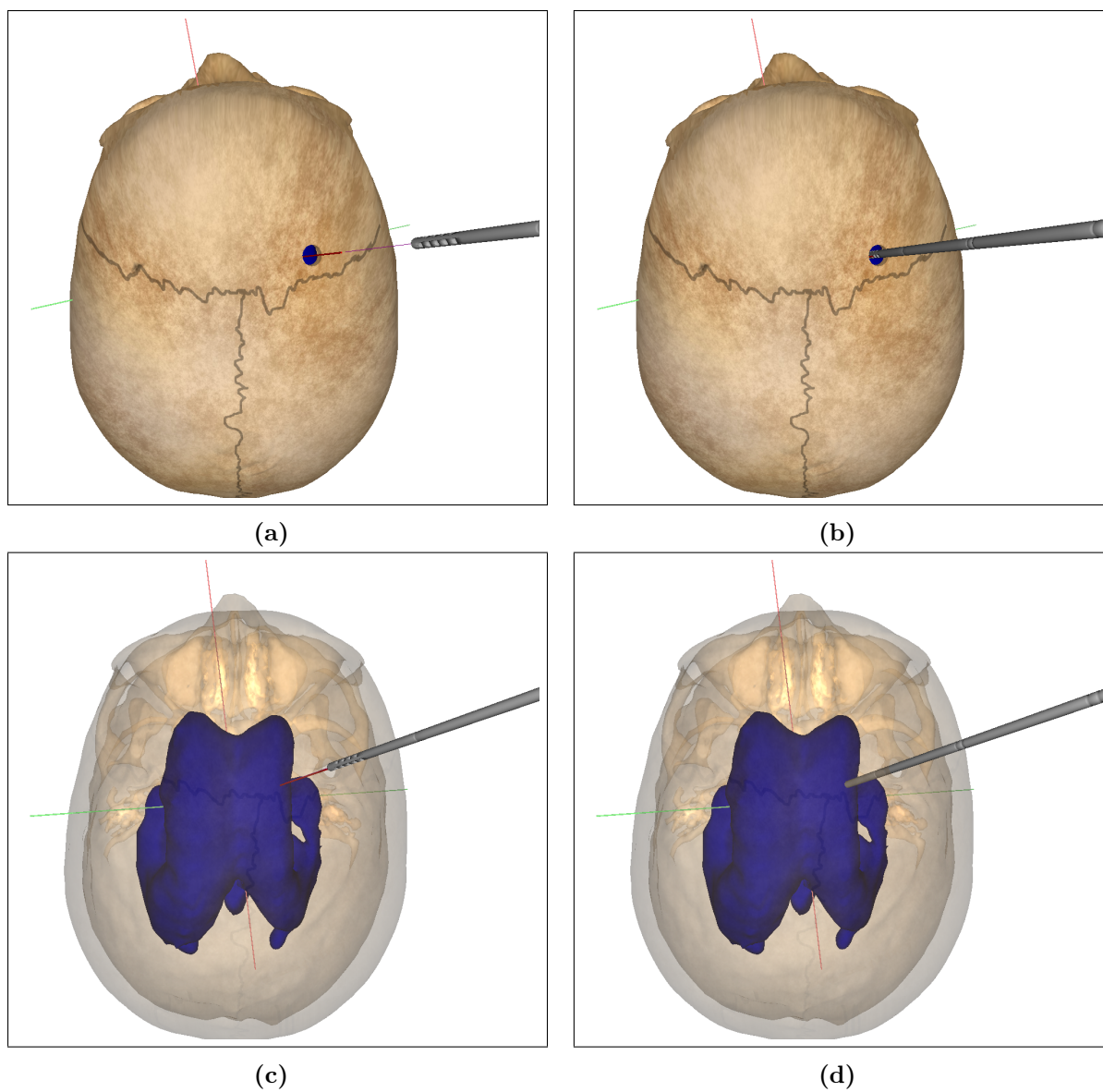


Figura 6.19: Escena de simulación para la práctica de ventriculostomía. (a), (b) Inserción del catéter ventricular. (c), (d) Visualización en transparencia del tejido óseo.

Las gráficas mostradas en esta sección son resultado de los datos obtenidos de la simulación al realizar un procedimiento de punción.

Finalmente, una propuesta para la evaluación de este procedimiento es una métri-

6. CASOS DE ESTUDIO

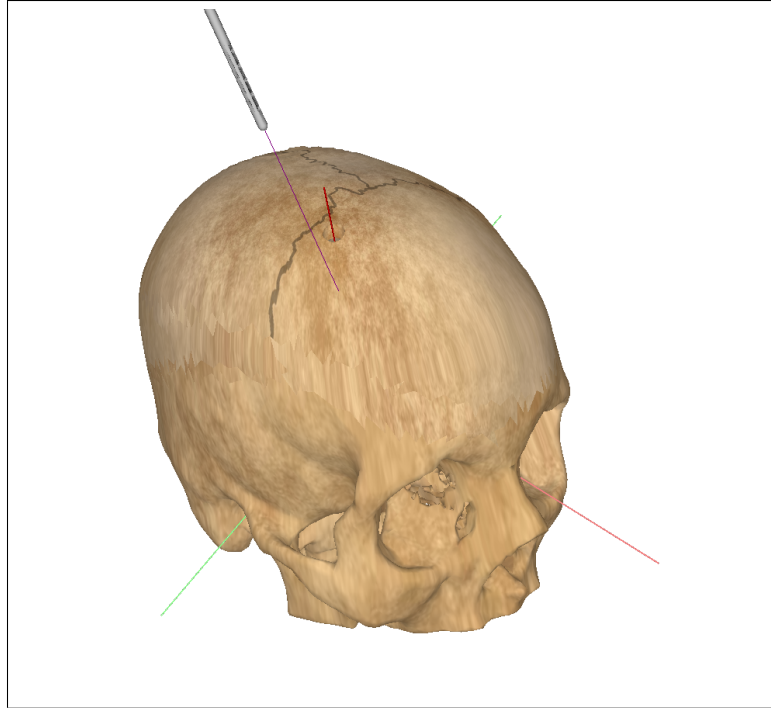


Figura 6.20: Líneas de referencia incorporadas al ambiente de simulación del procedimiento de ventriculostomía.

ca del error calculado como la distancia entre la posición de la punta del instrumento y el punto más cercano sobre la recta de trayectoria ideal. La ecuación de la distancia al punto más cercano de una recta en el espacio está dada por

$$d = \frac{|(\mathbf{x}_0 - \mathbf{x}_1) \times (\mathbf{x}_0 - \mathbf{x}_2)|}{|\mathbf{x}_2 - \mathbf{x}_1|} \quad (6.1)$$

donde $\mathbf{x}_0 = (x_0, y_0, z_0)$ es un punto en el espacio; mientras que $\mathbf{x}_1 = (x_1, y_1, z_1)$ y $\mathbf{x}_2 = (x_2, y_2, z_2)$ son dos puntos sobre la recta dada por la ecuación paramétrica

$$v = \begin{cases} x_1 + t(x_2 - x_1) \\ y_1 + t(y_2 - y_1) \\ z_1 + t(z_2 - z_1) \end{cases} \quad (6.2)$$

donde $t \in \mathbb{R}$. En la figura 6.26 se muestra la gráfica del error durante toda la simulación y en en la figura 6.27 se muestra el intervalo en el cual se llevó a acabo la punción. El valor promedio del error durante este intervalo fue de 0.46 cm con un error máximo de 0.89 cm.

6.2 Simulador Háptico-Virtual de Ventriculostomía

Parámetro	Valor
Umbral de punción del ventrículo	0.4 N
Peso virtual del instrumento	1 g
k	5.0
C_p	0.1 N
b_p	3.0 N
D_p	0.001 N
Δv	0.0001 m/s
Fuerza de corte (f_c)	0.05 N
Profundidad de penetración (d_1)	7 cm
f_{max}	2 N
f_1	0.2 N
d_2	7.3 cm

Tabla 6.4: Parámetros utilizados en el modelo de punción en el simulador de ventriculostomía.

Parámetro	Valor
Paso de tiempo de simulación	0.02 s
Número de hexaedros	83
Módulo de Young	0.001 MPa
Relación de Poisson	0.45
Masa de Rayleigh	0.1
Rigidez de Rayleigh	0.1
Masa Uniforme	5 g

Tabla 6.5: Parámetros utilizados en la escena de simulación de la práctica de ventriculostomía.

6. CASOS DE ESTUDIO



(a)



(b)

Figura 6.21: (a) Detalle de la pieza manufacturada para el acoplamiento del catéter ventricular. (b) Montaje en el dispositivo háptico Phantom Touch y se muestra el modelo en impresión 3D de un paciente real obtenido de la segmentación de imágenes de tomografía.

6.2 Simulador Háptico-Virtual de Ventriculostomía

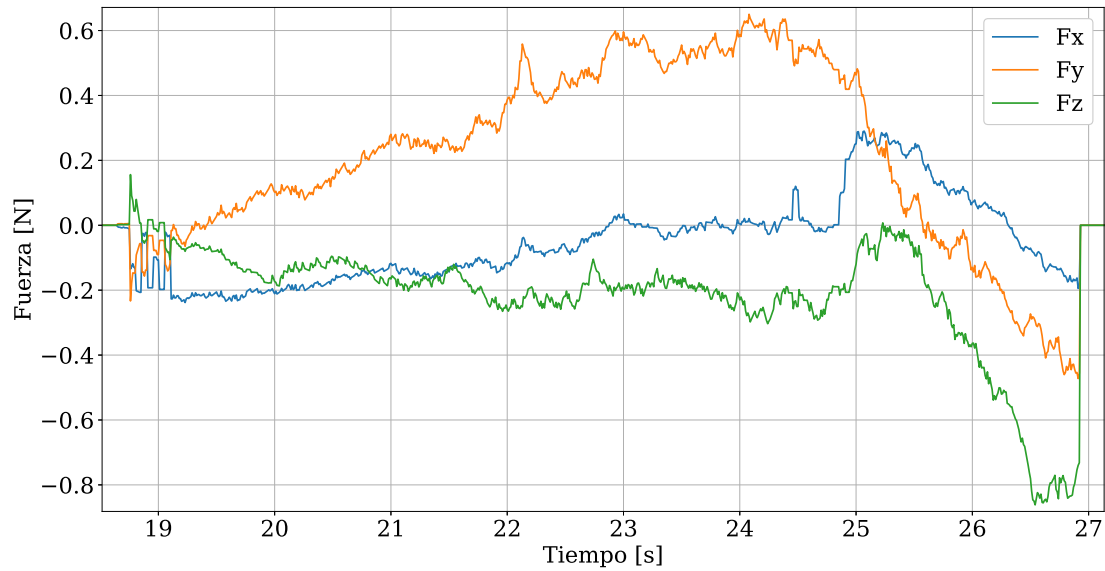


Figura 6.22: Gráfica de fuerza generada por el dispositivo háptico durante la simulación de punción ventricular.

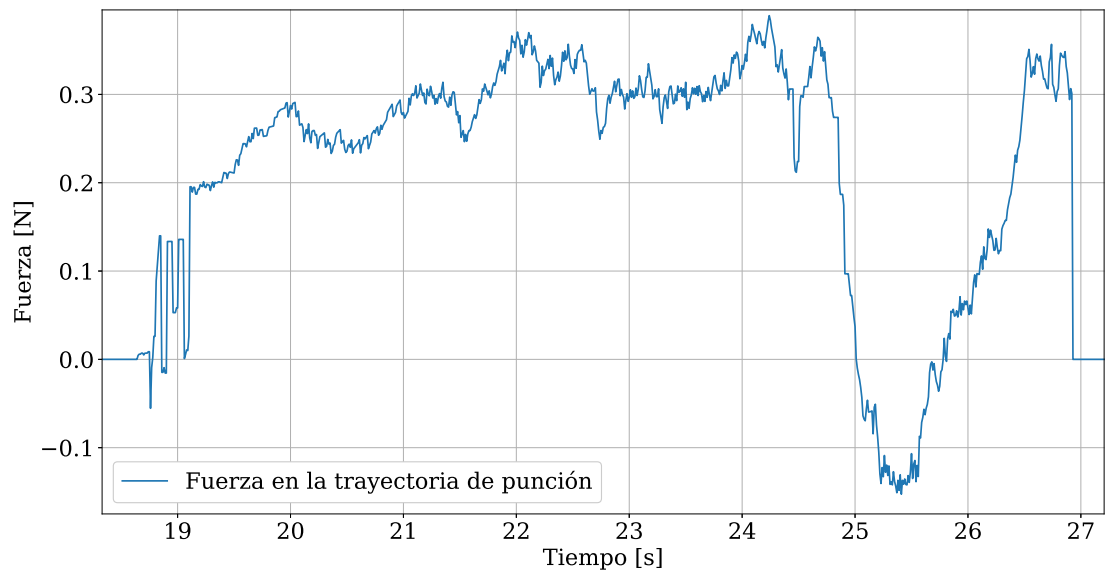


Figura 6.23: Gráfica de fuerza generada por el dispositivo háptico durante la simulación de punción ventricular en la dirección de punción.

6. CASOS DE ESTUDIO

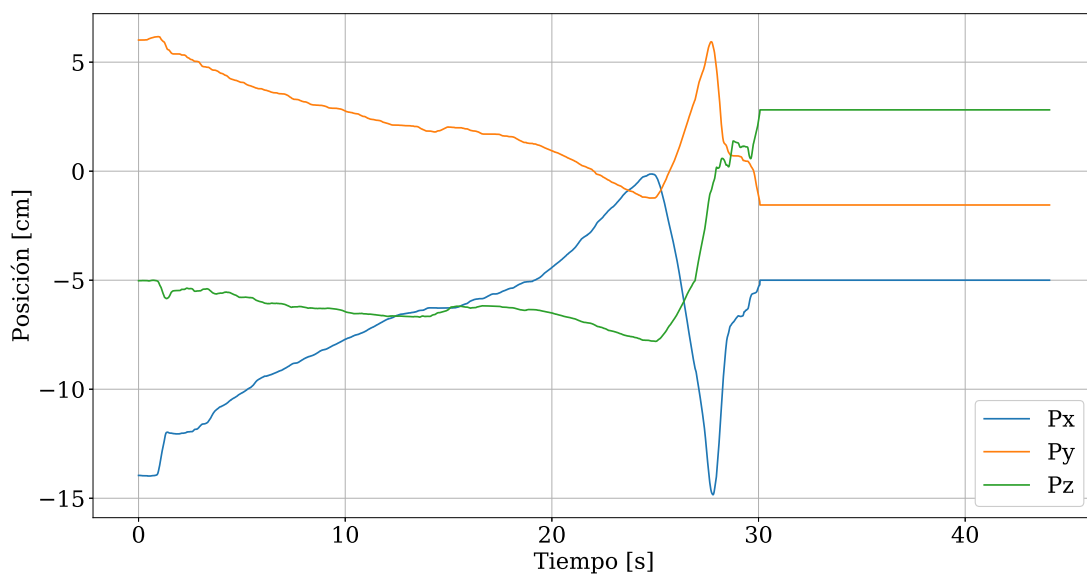


Figura 6.24: Gráfica de la posición registrada por el dispositivo háptico durante la simulación de punción ventricular.

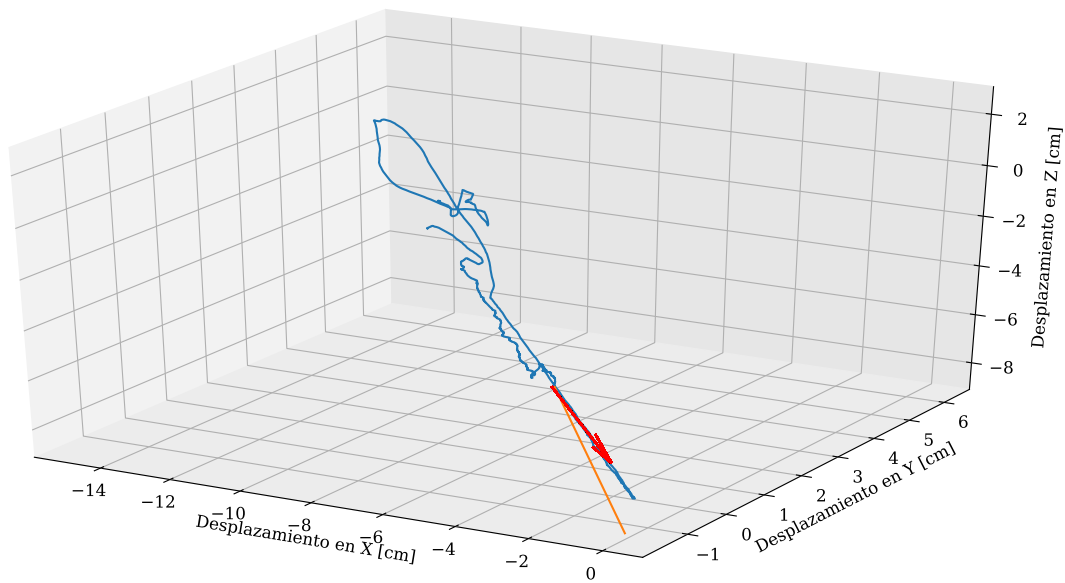


Figura 6.25: Gráfica de la trayectoria registrada por el dispositivo háptico durante la simulación de punción ventricular. En color rojo se muestra el vector de punción durante la simulación y en color anaranjado la recta que define una trayectoria ideal de acuerdo con las líneas de referencia.

6. CASOS DE ESTUDIO

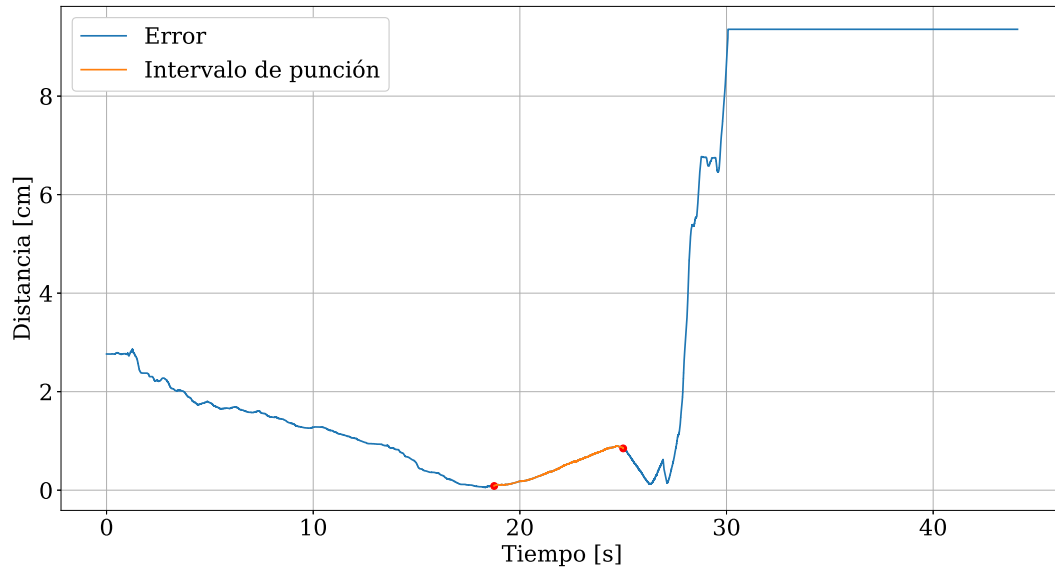


Figura 6.26: Gráfica del error de la posición del dispositivo háptico con respecto a la recta de referencia ideal. En color anaranjado se muestra el intervalo en que ocurre el procedimiento de punción.

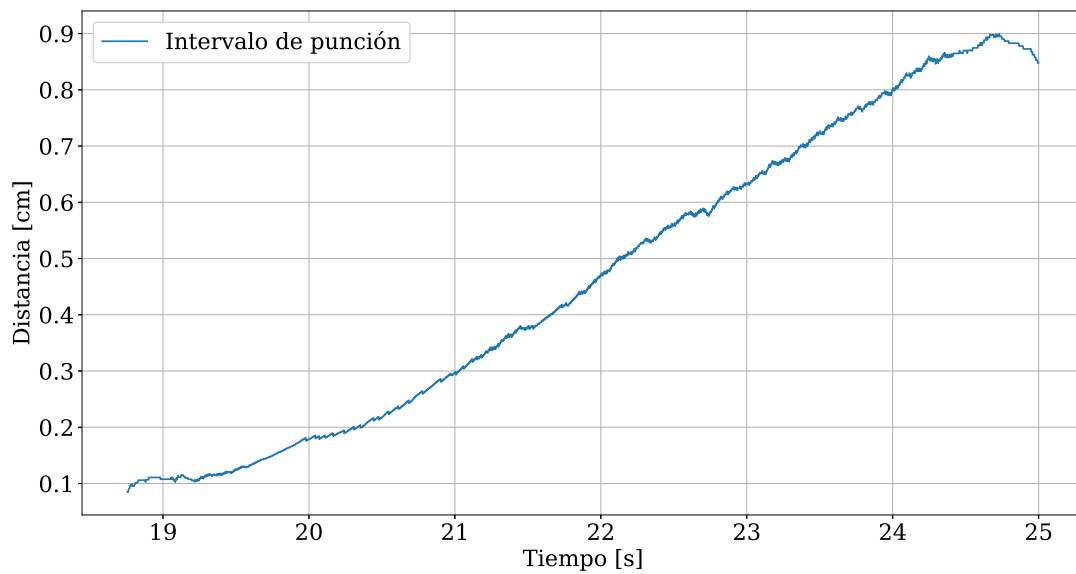


Figura 6.27: Gráfica del error de la posición del dispositivo háptico con respecto a la recta de referencia ideal en el intervalo de punción.

Conclusiones y Trabajo a Futuro

7.1 Conclusiones

En el trabajo presentado se desarrolló un modelo de punción háptica para sistemas de entrenamiento de procedimientos de punción en el área médica. El modelo fue implementado satisfactoriamente como un plugin en el framework de SOFA, lo cual permite la flexibilidad de incorporarlo en diversas escenas de simulación de procedimientos que involucren la punción de tejidos tal como se muestra en los casos de estudio descritos.

Además, el modelo propuesto permite ser ajustado a través de parámetros que caracterizan los tejidos con los cuales se interactúa en el entorno virtual, de modo que, a través de una evaluación experimental o a través de pruebas psicofísicas, el experto puede proveer información que permita estimar y ajustar estos parámetros para aproximarlos cada vez más a la experiencia real.

Se presentaron dos casos de estudio que ejemplifican la aplicación del modelo háptico propuesto, estos son: a) simulación de la práctica de venopunción y b) del procedimiento de ventriculostomía. Los casos de estudio presentados en esta tesis constituyen una herramienta más que es factible de utilizarse para complementar el proceso de entrenamiento de especialistas en el área médica, a través de la definición de métricas para obtener un análisis cuantitativo sobre la destreza del practicante.

7. CONCLUSIONES Y TRABAJO A FUTURO

De la evaluación realizada se obtuvieron resultados prometedores que indican la factibilidad de llevar este tipo de simuladores a los centros de formación y para incrementar las destrezas y habilidades de los practicantes. Se mencionó que la percepción háptica sí se aproxima a la sensación que se experimenta al realizar el procedimiento en un paciente.

De los resultados obtenidos se puede concluir que el uso de simuladores virtuales para la práctica médica pueden servir para complementar el proceso de entrenamiento de los estudiantes y, a su vez, pueden ser útiles para incrementar las destrezas de los médicos especialistas en diferentes ramas de la medicina.

7.2 Trabajo a Futuro

Algunas propuestas que caben destacar para dar continuidad a este trabajo son:

- Alinear los modelos virtuales con modelos físicos y crear un ambiente de realidad aumentada para tener una experiencia cada vez más inmersiva.
- Caracterizar los tejidos para determinar los valores de los parámetros de los modelos de punción háptica y para el modelo biomecánico.
- Realizar más pruebas con médicos especialistas para evaluar la factibilidad de utilizar el sistema como una herramienta de entrenamiento de médicos u otros practicantes de la salud.
- Es recomendable utilizar un dispositivo háptico que provea retroalimentación de par (6 grados de libertad) para proveer al usuario de mayor sensación táctil de los tejidos durante la interacción háptica.
- Diseñar protocolos a partir de la definición y análisis de métricas adecuadas para evaluar de manera objetiva el procedimiento realizado por un practicante, para que de esta forma se pueda cuantificar la expertiz del usuario.

Solución Numérica del Modelo Biomecánico

La simulación computacional requiere de algoritmos desarrollados para solucionar modelos de fenómenos físicos. Actualmente existen diversos métodos numéricos para la simulación de tejidos deformables, entre ellos se encuentra la aproximación de la solución de ecuaciones diferenciales parciales a partir de la teoría mecánica del medio continuo [42]. Entre los métodos más utilizados en la simulación biomecánica de tejidos suaves se encuentra el método de elemento finito y en este capítulo se describe la solución numérica del modelo presentado en el capítulo 2 utilizando este método.

A.1 Método de Elemento Finito

En esta sección se describe la aproximación de la solución del modelo elástico lineal en tres dimensiones utilizando la formulación por el método de elemento finito.

A.1.1 Formulación Débil de las Ecuaciones de Equilibrio

El objetivo es determinar la formulación débil de las ecuaciones diferenciales de equilibrio que se resumen en la sección 2.3. Para ello consideramos el vector arbitrario \mathbf{v}

A. SOLUCIÓN NUMÉRICA DEL MODELO BIOMECÁNICO

$$\mathbf{v} = \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} \quad (\text{A.1})$$

De acuerdo con la relación cinemática 2.21, se tiene que

$$\tilde{\nabla} \mathbf{v} = \begin{bmatrix} \frac{\partial v_x}{\partial x} \\ \frac{\partial v_y}{\partial y} \\ \frac{\partial v_z}{\partial z} \\ \frac{\partial v_x}{\partial y} + \frac{\partial v_y}{\partial x} \\ \frac{\partial v_x}{\partial z} + \frac{\partial v_z}{\partial x} \\ \frac{\partial v_y}{\partial z} + \frac{\partial v_z}{\partial y} \end{bmatrix} \quad (\text{A.2})$$

entonces

$$\begin{aligned} (\tilde{\nabla} \mathbf{v})^T \boldsymbol{\sigma} &= \frac{\partial v_x}{\partial x} \sigma_{xx} + \frac{\partial v_y}{\partial y} \sigma_{yy} + \frac{\partial v_z}{\partial z} \sigma_{zz} + \left(\frac{\partial v_x}{\partial y} + \frac{\partial v_y}{\partial x} \right) \sigma_{xy} \\ &+ \left(\frac{\partial v_x}{\partial z} + \frac{\partial v_z}{\partial x} \right) \sigma_{xz} + \left(\frac{\partial v_y}{\partial z} + \frac{\partial v_z}{\partial y} \right) \sigma_{yz} \end{aligned} \quad (\text{A.3})$$

Para derivar la formulación débil de la ecuación 2.13 se multiplica la primera ecuación por la función arbitraria v_x y se integra sobre el volumen V para obtener

$$\int_V v_x \frac{\partial \sigma_{xx}}{\partial x} dV + \int_V v_x \frac{\partial \sigma_{xy}}{\partial y} dV + \int_V v_x \frac{\partial \sigma_{xz}}{\partial z} dV + \int_V v_x b_x dV = 0 \quad (\text{A.4})$$

Mediante integración por partes utilizando el teorema de Green-Gauss en la ecuación anterior, se obtiene la ecuación

$$\begin{aligned} \int_S v_x \sigma_{xx} n_x dS - \int_V \frac{\partial v_x}{\partial x} \sigma_{xx} dV + \int_S v_x \sigma_{xy} n_y dS - \int_V \frac{\partial v_x}{\partial y} \sigma_{xy} dV \\ + \int_S v_x \sigma_{xz} n_z dS - \int_V \frac{\partial v_x}{\partial z} \sigma_{xz} dV + \int_V v_x b_x dV = 0 \end{aligned} \quad (\text{A.5})$$

La componente t_x del vector de tracción \mathbf{t} está dado por 2.4, por lo que la expresión anterior se reduce a

$$\int_S v_x t_x dS - \int_V \left(\frac{\partial v_x}{\partial x} \sigma_{xx} + \frac{\partial v_x}{\partial y} \sigma_{xy} + \frac{\partial v_x}{\partial z} \sigma_{xz} \right) dV + \int_V v_x b_x dV = 0 \quad (\text{A.6})$$

De manera similar para las otras dos ecuaciones de 2.13 se multiplican por las funciones arbitrarias v_y y v_z , respectivamente, de esta forma se tiene que

$$\int_S v_y t_y dS - \int_V \left(\frac{\partial v_y}{\partial x} \sigma_{yx} + \frac{\partial v_y}{\partial y} \sigma_{yy} + \frac{\partial v_y}{\partial z} \sigma_{yz} \right) dV + \int_V v_y b_y dV = 0 \quad (\text{A.7})$$

$$\int_S v_z t_z dS - \int_V \left(\frac{\partial v_z}{\partial x} \sigma_{zx} + \frac{\partial v_z}{\partial y} \sigma_{zy} + \frac{\partial v_z}{\partial z} \sigma_{zz} \right) dV + \int_V v_z b_z dV = 0 \quad (\text{A.8})$$

De la suma de las ecuaciones A.6 - A.8 resulta que

$$\begin{aligned} & \int_S (v_x t_x + v_y t_y + v_z t_z) dS + \int_V (v_x b_x + v_y b_y + v_z b_z) dV \\ & - \int_V \left[\frac{\partial v_x}{\partial x} \sigma_{xx} + \frac{\partial v_y}{\partial y} \sigma_{yy} + \frac{\partial v_z}{\partial z} \sigma_{zz} + \left(\frac{\partial v_x}{\partial y} + \frac{\partial v_y}{\partial x} \right) \sigma_{xy} \right. \\ & \quad \left. + \left(\frac{\partial v_x}{\partial z} + \frac{\partial v_z}{\partial x} \right) \sigma_{xz} + \left(\frac{\partial v_y}{\partial z} + \frac{\partial v_z}{\partial y} \right) \sigma_{yz} \right] dV = 0 \end{aligned} \quad (\text{A.9})$$

Finalmente, utilizando A.3 se obtiene la formulación débil

$$\int_V (\tilde{\nabla} \mathbf{v})^T \boldsymbol{\sigma} dV = \int_S \mathbf{v}^T \mathbf{t} dS + \int_V \mathbf{v}^T \mathbf{b} dV \quad (\text{A.10})$$

Esta ecuación, conocida como ecuación de trabajo virtual, es válida para cualquier relación constitutiva y para cualquier vector de peso \mathbf{v} .

A.1.2 Formulación Elástica por el Método de Elemento Finito

A partir de la formulación débil A.10, se pueden derivar las ecuaciones de elemento finito para el modelo elástico. Para lograr esto, se aproxima el vector de desplazamiento \mathbf{u} como

A. SOLUCIÓN NUMÉRICA DEL MODELO BIOMECÁNICO

$$\mathbf{u} = \mathbf{N}\mathbf{a} \quad (\text{A.11})$$

donde \mathbf{N} es una función de interpolación denominada función de forma dada por

$$\mathbf{N} = \begin{bmatrix} N_1 & 0 & 0 & N_2 & 0 & 0 & \dots & N_n & 0 & 0 \\ 0 & N_1 & 0 & 0 & N_2 & 0 & \dots & 0 & N_n & 0 \\ 0 & 0 & N_1 & 0 & 0 & N_2 & \dots & 0 & 0 & N_n \end{bmatrix}; \quad \mathbf{a} = \begin{bmatrix} u_{x1} \\ u_{y1} \\ u_{z1} \\ u_{x2} \\ u_{y2} \\ u_{z2} \\ \vdots \\ u_{xn} \\ u_{yn} \\ u_{zn} \end{bmatrix} \quad (\text{A.12})$$

y donde n es el número de nodos del cuerpo completo. El método de Galerkin significa que el vector de peso \mathbf{v} se elige de acuerdo con

$$\mathbf{v} = \mathbf{N}\mathbf{c} \quad (\text{A.13})$$

Como el vector \mathbf{v} es arbitrario, \mathbf{c} es un vector arbitrario. Entonces de la ecuación anterior se obtiene que

$$\tilde{\nabla}\mathbf{v} = \mathbf{B}\mathbf{c} \quad \text{donde} \quad \mathbf{B} = \tilde{\nabla}\mathbf{N} \quad (\text{A.14})$$

Sustituyendo A.13 y A.14 en la ecuación de la formulación débil A.10, y como \mathbf{c} es independiente del sistema de coordenadas, se obtiene que

$$\mathbf{c}^T \left(\int_V \mathbf{B}^T \boldsymbol{\sigma} dV - \int_S \mathbf{N}^T \mathbf{t} dS - \int_V \mathbf{N}^T \mathbf{b} dV \right) = 0 \quad (\text{A.15})$$

y como la matriz \mathbf{c} es arbitraria, se concluye que

$$\int_V \mathbf{B}^T \boldsymbol{\sigma} dV = \int_S \mathbf{N}^T \mathbf{t} dS + \int_V \mathbf{N}^T \mathbf{b} dV \quad (\text{A.16})$$

Esta ecuación satisface cualquier modelo constitutivo.

Considerando el modelo elástico lineal, incluimos el modelo constitutivo correspondiente 2.22

$$\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\epsilon} \quad (\text{A.17})$$

La relación cinemática 2.21 está dada por

$$\boldsymbol{\epsilon} = \tilde{\nabla}\mathbf{u} \quad (\text{A.18})$$

De las ecuaciones A.11 y A.14 se obtiene la relación

$$\boldsymbol{\epsilon} = \mathbf{B}\mathbf{a} \quad (\text{A.19})$$

entonces la ecuación A.18 queda como

$$\boldsymbol{\sigma} = \mathbf{D}\mathbf{B}\mathbf{a} \quad (\text{A.20})$$

Con la ecuación anterior, la ecuación A.16 toma la forma

$$\left(\int_V \mathbf{B}^T \mathbf{D} \mathbf{B} dV \right) \mathbf{a} = \int_S \mathbf{N}^T \mathbf{t} dS + \int_V \mathbf{N}^T \mathbf{b} dV \quad (\text{A.21})$$

Considerando las condiciones de frontera

$$\begin{aligned} \mathbf{t} &= \mathbf{S}\mathbf{n} = \mathbf{h} & \text{en } S_h \\ \mathbf{u} &= \mathbf{g} & \text{en } S_g \end{aligned} \quad (\text{A.22})$$

donde \mathbf{h} y \mathbf{g} son vectores conocidos. Con esta formulación la ecuación A.21 toma la forma siguiente

$$\left(\int_V \mathbf{B}^T \mathbf{D} \mathbf{B} dV \right) \mathbf{a} = \int_{S_h} \mathbf{N}^T \mathbf{h} dS + \int_{S_g} \mathbf{N}^T \mathbf{t} dS + \int_V \mathbf{N}^T \mathbf{b} dV \quad (\text{A.23})$$

que es la formulación por elemento finito que se busca. Para escribir esta formulación de manera más compacta, se definen las siguientes matrices

A. SOLUCIÓN NUMÉRICA DEL MODELO BIOMECÁNICO

$$\begin{aligned}
 \mathbf{K} &= \int_V \mathbf{B}^T \mathbf{D} \mathbf{B} dV \\
 \mathbf{f}_b &= \int_{S_h} \mathbf{N}^T \mathbf{t} dS + \int_{S_g} \mathbf{N}^T \mathbf{t} dS \\
 \mathbf{f}_l &= \int_V \mathbf{N}^T \mathbf{b} dV
 \end{aligned} \tag{A.24}$$

donde \mathbf{K} es la matriz de rigidez, \mathbf{f}_b es el vector de frontera y \mathbf{f}_l el vector de cargas. Utilizando las definiciones A.24, la ecuación A.23 se puede escribir como

$$\mathbf{K} \mathbf{a} = \mathbf{f}_b + \mathbf{f}_l \tag{A.25}$$

Además, si se define el vector de fuerzas \mathbf{f} como

$$\mathbf{f} = \mathbf{f}_b + \mathbf{f}_l \tag{A.26}$$

Se obtiene la formulación estándar de elemento finito

$$\mathbf{K} \mathbf{a} = \mathbf{f} \tag{A.27}$$

como \mathbf{D} es simétrica y definida positiva, la matriz \mathbf{K} también es simétrica y semi-definida positiva ya que $\boldsymbol{\epsilon} = \mathbf{B} \mathbf{a} = \mathbf{0}$ cuando \mathbf{a} corresponde a movimientos de cuerpo rígido, si se cuenta con las condiciones de frontera suficientes los movimientos de cuerpo rígido se eliminan obteniendo así una solución única.

A.1.3 Discretización del Modelo Dinámico

Haciendo un análisis similar al modelo en estado estacionario, a continuación se expresa el modelo discretizado del modelo dinámico 2.34.

Considerando la ecuación A.11 reemplazando \mathbf{a} por $\tilde{\mathbf{u}}$ y sustituyendo \mathbf{f}_l en la ecuación A.24 por

$$\mathbf{f}_l = - \int_V \mathbf{N}^T \bar{\mathbf{b}} dV + \left(\int_V \mathbf{N}^T \rho \mathbf{N} dV \right) \frac{d^2 \tilde{\mathbf{u}}}{dt^2} + \left(\int_V \mathbf{N}^T \boldsymbol{\mu} \mathbf{N} dV \right) \frac{d \tilde{\mathbf{u}}}{dt} \tag{A.28}$$

que es la discretización de la ecuación 2.33. De forma análoga a la ecuación A.27 sustituyendo la ecuación A.28 en las ecuaciones generales de equilibrio se obtiene finalmente, después de ensamblar las matrices (suma de las matrices de cada elemento), la ecuación diferencial matricial

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} + \mathbf{f} = \mathbf{0} \quad (\text{A.29})$$

donde \mathbf{K} y \mathbf{f} son las matrices de rigidez y fuerzas externas, esfuerzos iniciales, etc., como se describió anteriormente. Las matrices \mathbf{C} y \mathbf{M} están dadas por

$$\begin{aligned} \mathbf{C} &= \int_V \mathbf{N}^T \boldsymbol{\mu} \mathbf{N} dV \\ \mathbf{M} &= \int_V \mathbf{N}^T \rho \mathbf{N} dV \end{aligned} \quad (\text{A.30})$$

La matriz \mathbf{M} se denomina matriz de masa y la matriz \mathbf{C} es conocida como matriz de amortiguamiento del sistema.

En la práctica, es difícil determinar la matriz \mathbf{C} debido a que no siempre se conocen los coeficientes para determinar $\boldsymbol{\mu}$. Generalmente, se asume que la matriz de amortiguamiento es una combinación lineal de la matriz de rigidez y la matriz de masa dada por

$$\mathbf{C} = \alpha \mathbf{M} + \beta \mathbf{K} \quad (\text{A.31})$$

Los parámetros α y β se determinan de forma experimental. Este modelo se conoce como amortiguamiento de Rayleigh que además proporciona estabilidad numérica en la solución.

Finalmente, para resolver la ecuación A.29, ésta se puede reescribir como

$$\ddot{\mathbf{x}} = \mathbf{M}^{-1} \mathbf{f}(\mathbf{x}, \mathbf{v}) \quad (\text{A.32})$$

donde $\mathbf{x} = \mathbf{u}$, $\mathbf{v} = \dot{\mathbf{u}}$ y $\ddot{\mathbf{x}} = \ddot{\mathbf{u}}$ representan la posición, la velocidad y la aceleración respectivamente, mientras que $\mathbf{f}(\mathbf{x}, \mathbf{v}) = \mathbf{f}^{ext} - \mathbf{K}\mathbf{x} - \mathbf{C}\mathbf{v}$.

A.2 Solución del Modelo Dinámico

Los métodos de integración numérica temporales se utilizan para la solución numérica de problemas de ecuaciones diferenciales dependientes del tiempo con valores iniciales.

Los problemas de valor inicial tienen la forma

$$u'(t) = f(u(t), t) \quad \text{para } t > t_0 \quad (\text{A.33})$$

con valores iniciales

$$u(t_0) = \nu. \quad (\text{A.34})$$

En la simulación en tiempo real es importante tener la solución del sistema de ecuaciones en un tiempo definido.

A.2.1 Método de Euler Implícito

También se conoce como método de Euler hacia atrás, ya que para discretizar la ecuación diferencial A.33 se utiliza el método de diferencias finitas hacia atrás de modo que, para una ecuación diferencial ordinaria, se obtiene la ecuación

$$\frac{u^{n+1} - u^n}{h} = f(u^{n+1}) \quad (\text{A.35})$$

o

$$u^{n+1} = u^n + hf(u^{n+1}) \quad (\text{A.36})$$

Con este método se tiene que resolver un sistema de ecuaciones en cada paso de tiempo; sin embargo, se tiene la ventaja de que es un método incondicionalmente estable, esto implica que se pueden tener pasos de tiempo más grandes y así alcanzar el tiempo requerido para la simulación en tiempo real sin perder estabilidad en la solución.

A.2 Solución del Modelo Dinámico

Para resolver la ecuación dinámica de segundo grado se utiliza el siguiente algoritmo propuesto por Baraff y Witkin [43].

Se tiene la ecuación que al ser discretizada se resuelve como una ecuación diferencial ordinaria A.32

$$\ddot{\mathbf{x}} = \mathbf{M}^{-1}\mathbf{f}(\mathbf{x}, \mathbf{v}) \quad (\text{A.37})$$

Dada la posición $\mathbf{x}(t_0)$ y la velocidad $\dot{\mathbf{x}}(t_0)$ del sistema en un tiempo t_0 , el objetivo es determinar la posición $\mathbf{x}(t_0 + h)$ y la velocidad $\dot{\mathbf{x}}(t_0 + h)$ en el tiempo $t_0 + h$. Para calcular el estado siguiente y la velocidad utilizando el método implícito, primero transformamos la ecuación A.37 en un sistema de ecuaciones diferenciales de primer orden. Esto se obtiene al definir la velocidad del sistema \mathbf{v} como $\mathbf{v} = \dot{\mathbf{x}}$ y entonces

$$\frac{d}{dt} \begin{pmatrix} \mathbf{x} \\ \dot{\mathbf{x}} \end{pmatrix} = \frac{d}{dt} \begin{pmatrix} \mathbf{x} \\ \mathbf{v} \end{pmatrix} = \begin{pmatrix} \mathbf{v} \\ \mathbf{M}^{-1}\mathbf{f}(\mathbf{x}, \mathbf{v}) \end{pmatrix} \quad (\text{A.38})$$

Para simplificar la notación definimos $\mathbf{x}_0 = \mathbf{x}(t_0)$, $\mathbf{v}_0 = \mathbf{v}(t_0)$, $\Delta\mathbf{x} = \mathbf{x}(t_0 + h) - \mathbf{x}(t_0)$ y $\Delta\mathbf{v} = \mathbf{v}(t_0 + h) - \mathbf{v}(t_0)$.

Con el método implícito de Euler hacia atrás, la ecuación A.38 se puede expresar como

$$\begin{pmatrix} \Delta\mathbf{x} \\ \Delta\mathbf{v} \end{pmatrix} = h \begin{pmatrix} \mathbf{v}_0 + \Delta\mathbf{v} \\ \mathbf{M}^{-1}\mathbf{f}(\mathbf{x}_0 + \Delta\mathbf{x}, \mathbf{v}_0 + \Delta\mathbf{v}) \end{pmatrix} \quad (\text{A.39})$$

En este método se requiere resolver para obtener valores de $\Delta\mathbf{x}$ y $\Delta\mathbf{v}$ que satisfagan la ecuación A.39. La ecuación A.39 es una ecuación no lineal y en lugar de resolverla de manera exacta utilizando un algoritmo iterativo para este tipo de ecuaciones, se utiliza una aproximación de primer orden utilizando series de Taylor al término \mathbf{f}

$$\mathbf{f}(\mathbf{x}_0 + \Delta\mathbf{x}, \mathbf{v}_0 + \Delta\mathbf{v}) = \mathbf{f}_0 + \frac{\partial\mathbf{f}}{\partial\mathbf{x}}\Delta\mathbf{x} + \frac{\partial\mathbf{f}}{\partial\mathbf{v}}\Delta\mathbf{v} \quad (\text{A.40})$$

En esta ecuación la derivada $\partial\mathbf{f}/\partial\mathbf{x}$ se evalúa en $(\mathbf{x}_0, \mathbf{v}_0)$ y de manera similar para $\partial\mathbf{f}/\partial\mathbf{v}$. Se define $\mathbf{f}_0 = \mathbf{f}(\mathbf{x}_0, \mathbf{v}_0)$. Sustituyendo esta aproximación en A.39 se obtiene el sistema lineal

A. SOLUCIÓN NUMÉRICA DEL MODELO BIOMECÁNICO

$$\begin{pmatrix} \Delta \mathbf{x} \\ \Delta \mathbf{v} \end{pmatrix} = h \begin{pmatrix} \mathbf{v}_0 + \Delta \mathbf{v} \\ \mathbf{M}^{-1}(\mathbf{f}_0 + \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \Delta \mathbf{x} + \frac{\partial \mathbf{f}}{\partial \mathbf{v}} \Delta \mathbf{v}) \end{pmatrix} \quad (\text{A.41})$$

Tomando la fila inferior de la ecuación A.41 y sustituyendo $\Delta \mathbf{x} = h(\mathbf{v}_0 + \Delta \mathbf{v})$ se obtiene

$$\Delta \mathbf{v} = h \mathbf{M}^{-1} \left(\mathbf{f}_0 + \frac{\partial \mathbf{f}}{\partial \mathbf{x}} h(\mathbf{v}_0 + \Delta \mathbf{v}) + \frac{\partial \mathbf{f}}{\partial \mathbf{v}} \Delta \mathbf{v} \right) \quad (\text{A.42})$$

Sea \mathbf{I} la matriz identidad, y reagrupando términos se obtiene

$$\left(\mathbf{I} - h \mathbf{M}^{-1} \frac{\partial \mathbf{f}}{\partial \mathbf{v}} - h^2 \mathbf{M}^{-1} \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right) \Delta \mathbf{v} = h \mathbf{M}^{-1} \left(\mathbf{f}_0 + h \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \mathbf{v}_0 \right) \quad (\text{A.43})$$

para la cual se obtiene una solución para $\Delta \mathbf{v}$. Una vez que se obtiene $\Delta \mathbf{v}$, calculamos $\Delta \mathbf{x} = h(\mathbf{v}_0 + \Delta \mathbf{v})$. Finalmente, esta ecuación forma un sistema de ecuaciones lineales.

De manera general, el algoritmo de Euler hacia atrás consiste en evaluar \mathbf{f}_0 , $\partial \mathbf{f} / \partial \mathbf{x}$ y $\partial \mathbf{f} / \partial \mathbf{v}$, formando el sistema de la ecuación A.43; se obtiene la solución para $\Delta \mathbf{v}$; después se actualiza \mathbf{x} y \mathbf{v} . Para solucionar el sistema de ecuaciones lineales obtenido se utiliza un algoritmo como el que se describe en la sección A.3.

A.3 Algoritmo de Gradiente Conjugado

El algoritmo de Gradiente Conjugado es un método del subespacio de Krylov que, aunque originalmente pensado como un algoritmo directo [44], actualmente es muy utilizado como un algoritmo iterativo para solucionar grandes sistemas de ecuaciones lineales. Para utilizar este método, el sistema de ecuaciones debe cumplir con las características de ser un sistema disperso con entradas no nulas que ocurren con un patrón predecible. Además, cuando este sistema se representa como una matriz, la matriz debe de ser simétrica y definida positiva. Como es el caso estos tipos de sistemas generalmente aparecen en la solución numérica de ecuaciones diferenciales con condiciones de valor en la frontera [45].

A.3 Algoritmo de Gradiente Conjugado

Cómo se mencionó se utiliza ampliamente para resolver sistemas de ecuaciones lineales del tipo:

$$\mathbf{Ax} = \mathbf{b} \quad (\text{A.44})$$

donde A es una matriz simétrica definida positiva, por lo que se cumple que

$$\begin{aligned} \mathbf{A} &= \mathbf{A}^T \\ \mathbf{x}^T \mathbf{Ax} &> 0 \quad \forall \mathbf{x} \neq 0 \end{aligned} \quad (\text{A.45})$$

El objetivo es encontrar un vector \mathbf{x} tal que $\|\mathbf{b} - \mathbf{Ax}\|_2 \leq \epsilon \|\mathbf{b}\|_2$, para una ϵ dada. En el algoritmo mostrado a continuación, la entrada es el vector de valores inicial de x , que se sobrescribe en cada iteración, el vector del lado derecho de la ecuación b , y una subrutina que se encarga del producto matriz por vector. Se limita el número de operaciones a $kmax$ y devuelve una solución x , ϵ evalúa la precisión de la solución [46]. El algoritmo de gradiente conjugado es el siguiente:

Algoritmo. Método de Gradiente Conjugado

Función $cg(x, b, A, \epsilon, kmax)$

```
r = b - Ax
ρ0 = ||r||22
k = 1
while √ρk-1 > ε||b||2 and k < kmax do
  if k = 1 then
    | p = r
  else
    | β = ρk-1/ρk-2
    | p = r + βp
  end
  w = Ap
  α = ρk-1 / (pT w)
  x = x + αp
  r = r - αw
  ρk = ||r||22
  k = k + 1
end
```

donde $\|x\|_2 = \sqrt{x_1^2 + x_2^2 + \dots + x_n^2}$.

Para mayor información sobre el método de gradiente conjugado se puede consultar [46].

A. SOLUCIÓN NUMÉRICA DEL MODELO BIOMECÁNICO

Desarrollo de un Plugin en SOFA

La biblioteca SOFA es un conjunto de componentes que pueden ser integrados en una escena de simulación y una forma de agregar un componente al entorno de SOFA es a través del desarrollo de un plugin. Un plugin es una biblioteca dinámica (en Windows se denomina DLL, por sus siglas en inglés) que respeta ciertas convenciones para que las aplicaciones basadas en SOFA puedan cargarla en tiempo de ejecución y acceder a los componentes que proporciona. Ya que SOFA está basada en la herramienta de compilación CMake, es posible consultar la documentación correspondiente en la página de esta herramienta¹.

En este capítulo se describe, a través de un ejemplo, que archivos se requieren para integrar un plugin a la biblioteca de SOFA y poder ser utilizado en una escena de simulación.

B.1 Ejemplo de desarrollo de un plugin en SOFA

En esta sección se describen los requerimientos mínimos que se necesitan para desarrollar un plugin a partir de la documentación de la página de SOFA².

¹<https://cmake.org/documentation>

²<https://www.sofa-framework.org/community/doc/programming-with-sofa/start-coding/create-your-plugin/>

B. DESARROLLO DE UN PLUGIN EN SOFA

La forma más conveniente para escribir un plugin es colocar los archivos en el árbol fuente de SOFA, en la ruta `/applications/plugins/`. Por ejemplo, aquí se muestra la integración del plugin llamado `MyPlugin`, que se coloca en la carpeta `/applications/plugins/MyPlugin`. Esta carpeta contendrá los siguientes archivos:

- Los archivos de interfase del plugin: `initMyPlugin.h` y `initMyPlugin.cpp`;
- el archivo `CMakeList.txt` que describe como compilar el plugin.

B.1.1 Archivo `initMyPlugin.h`

Este archivo contiene las definiciones macros `DLL export/ DLL import` que se utilizan en Windows para indicar qué símbolos deben de ser visibles fuera del alcance de la DLL ya compilada. Para mayor detalle se puede consultar la documentación de Microsoft MSDN correspondiente a los atributos `dllexport` y `dllimport`³.

```
#ifndef INITMYPLUGIN_H
#define INITMYPLUGIN_H

#include <sofa/helper/system/config.h>

#ifdef SOFA_BUILD_MYPLUGIN
#define SOFA_MyPlugin_API SOFA_EXPORT_DYNAMIC_LIBRARY
#else
#define SOFA_MyPlugin_API SOFA_IMPORT_DYNAMIC_LIBRARY
#endif

/** Pagina principal
  Esta es la pagina principal de documentacion
  de Doxygen para MyPlugin.
 */

#endif
```

Listing B.1: `initMyPlugin.h`

³<https://docs.microsoft.com/en-us/cpp/cpp/dllexport-dllimport?view=vs-2017>

B.1.2 Archivo `initMyPlugin.cpp`

Este archivo contiene las definiciones de funciones que SOFA buscará cuando se intente cargar el plugin, estas son:

- `initExternalModule()`,
- `getModuleName()`,
- `getModuleVersion()`,
- `getModuleLicense()`,
- `getModuleDescription()`,
- `getModuleComponentList()`.

```
#include "initMyPlugin.h"

extern "C" {
    void initExternalModule ()
    {
        // Aqui se puede escribir el codigo de instalacion
        // que sera ejecutado antes de la creacion de cualquier componente
    }

    const char* getModuleName ()
    {
        return "MyPlugin";
    }

    const char* getModuleVersion ()
    {
        return "0.1 ";
    }

    const char* getModuleLicense ()
    {
        return "LGPL";
    }
}
```

B. DESARROLLO DE UN PLUGIN EN SOFA

```
    const char* getModuleDescription()
    {
        return "MyPlugin_provides_nothing_for_now.";
    }

    const char* getModuleComponentList()
    {
        // Comma-separated list of the components
        // in this plugin, empty for now
        return "";
    }
}
```

Listing B.2: initMyPlugin.cpp

B.1.3 Archivo CMakeLists.txt

Para integrar el plugin al sistema de compilación de SOFA, el archivo CMakeList.txt debe de tener la siguiente estructura:

```
cmake_minimum_required(VERSION 3.1)

#replace here with your project name and version number
project(MyPlugin VERSION X.X)

#look for all packages on which your project depends
find_package(SofaFramework REQUIRED)

set(HEADER_FILES
    MyHeaderFile1.h
    MyHeaderFile2.h
    MyHeaderFile3.h
)

set(SOURCE_FILES
    MySourceFile1.cpp
    MySourceFile2.cpp
    MySourceFile3.cpp
)
```

B.1 Ejemplo de desarrollo de un plugin en SOFA

```
add_library(${PROJECT_NAME}
            SHARED ${HEADER_FILES} ${SOURCE_FILES})
target_link_libraries(${PROJECT_NAME} SofaCore)
set_target_properties(${PROJECT_NAME}
                      PROPERTIES COMPILE_FLAGS "-DSOFA_BUILD_MYPLUGIN")
```

MyPlugin corresponde al nombre del plugin. El repositorio del plugin también debe de tener el mismo nombre. En este archivo se enlistan los archivos fuente y cabeceras que se desean incluir en el plugin para que puedan ser compilados.

Las dependencias requeridas se deben especificar como *REQUIRED*.

El plugin debe de ser agregado al proyecto como una nueva biblioteca incluyendo los archivos fuente y de cabecera. Para ello, se utiliza el comando de CMake *add_library*.

Por último, las dependencias se especifican utilizando el comando *target_link_libraries*

B.1.4 Instalación en CMake

Para instalar el plugin, el siguiente comando se tiene que añadir al final del archivo CMakeList.txt:

```
install(TARGETS MyPlugin
        RUNTIME DESTINATION bin
        LIBRARY DESTINATION lib
        ARCHIVE DESTINATION lib)
```

Y el siguiente comando con el nombre del plugin se tiene que agregar al archivo `/applications/plugins/CMakeLists.txt`

```
sofa_add_plugin(MyPlugin MyPlugin)
```

Finalmente, se compila el conjunto de bibliotecas de SOFA, como se describe en la documentación⁴ tomando en cuenta la activación del plugin correspondiente en la herramienta CMake.

⁴<https://www.sofa-framework.org/community/doc/getting-started/build/windows/>

B. DESARROLLO DE UN PLUGIN EN SOFA

Bibliografía

- [1] C. G. Corrêa, F. L. Nunes, E. Ranzini, R. Nakamura, and R. Tori, “Haptic interaction for needle insertion training in medical applications: The state-of-the-art,” *Medical Engineering & Physics*, nov 2018. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S135045331830167X>
- [2] S. Lee, J. Lee, A. Lee, N. Park, S. Lee, S. Song, A. Seo, H. Lee, J.-I. Kim, and K. Eom, “Augmented reality intravenous injection simulator based 3D medical imaging for veterinary medicine,” *The Veterinary Journal*, vol. 196, no. 2, pp. 197–202, may 2013. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1090023312004005>
- [3] N. Stobbs and B. N. Kumar, “The role of simulation in surgical training,” *Otorhinolaryngologist*, vol. 7, no. 3, pp. 169–172, 2014.
- [4] G. C. Burdea, *Force and Touch Feedback for Virtual Reality*. New York, NY, USA: John Wiley & Sons, Inc., 1996.
- [5] D. Feygin, M. Keehner, and R. Tendick, “Haptic guidance: experimental evaluation of a haptic training method for a perceptual motor skill,” in *Proceedings 10th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems. HAPTICS 2002*, 2002, pp. 40–47.
- [6] S. T. Vite, C. D. Velasco, A. F. H. Valencia, J. S. P. Lomelí, and M. Á. P. Castañeda, “Virtual Simulation of Brain Sylvian Fissure Exploration and Aneurysm Clipping with Haptic Feedback for Neurosurgical Training,” in *Augmented Reality, Virtual Reality, and Computer Graphics*, L. T. De Paolis and P. Bourdot, Eds. Cham: Springer International Publishing, 2018, pp. 230–238.
- [7] Robert L Frank, “Peripheral venous access in adults,” 2018. [Online]. Available: <https://www.uptodate.com/contents/peripheral-venous-access-in-adults>

BIBLIOGRAFÍA

- [8] T. R. Coles, D. Meglan, and N. W. John, “The Role of Haptics in Medical Training Simulators: A Survey of the State of the Art,” *IEEE Transactions on Haptics*, vol. 4, no. 1, pp. 51–66, 2011.
- [9] S. Shin, W. Park, H. Cho, S. Park, and L. Kim, “Needle insertion simulator with haptic feedback,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, J. A. Jacko, Ed., vol. 6762 LNCS, no. PART 2. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 119–124.
- [10] P. Xia and A. Sourin, “Design and implementation of a haptics-based virtual venepuncture simulation and training system,” *Proceedings of the 11th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and its Applications in Industry - VRCAI '12*, p. 25, 2012. [Online]. Available: <http://dl.acm.org/citation.cfm?doid=2407516.2407523>
- [11] P.-a. Heng, T.-t. Wong, R. Yang, and Y.-p. Chui, “Intelligent Inferencing and Haptic Simulation for Chinese Acupuncture Learning and Training ,” vol. 10, no. 1, pp. 28–41, 2006.
- [12] S. Teodoro Vite, C. F. Domínguez Velasco, S. Muscatello, M. Á. Padilla Castañeda, and L. T. De Paolis, “An Augmented Reality Platform for Preoperative Surgical Planning,” in *6th International Conference on Advancements of Medicine and Health Care through Technology; 17–20 October 2018, Cluj-Napoca, Romania*, S. Vlad and N. M. Roman, Eds. Singapore: Springer Singapore, 2019, pp. 177–182.
- [13] S. A. Heredia-Pérez, K. Harada, M. A. Padilla-Castañeda, M. Marques-Marinho, J. A. Márquez-Flores, and M. Mitsuishi, “Virtual reality simulation of robotic transsphenoidal brain tumor resection: Evaluating dynamic motion scaling in a master-slave system,” *The International Journal of Medical Robotics and Computer Assisted Surgery*, vol. 15, no. 1, p. e1953, 2019. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1002/rcs.1953>
- [14] A. Alaraj, *Comprehensive Healthcare Simulation: Neurosurgery*, 2016. [Online]. Available: <http://link.springer.com/10.1007/978-3-319-24187-6>
- [15] A. Alaraj, M. Lemole, J. Finkle, R. Yudkowsky, A. Wallace, C. Luciano, P. Bannerjee, S. Rizzi, and F. Charbel, “Virtual reality training in neurosurgery: Review of current status and future applications,” *Surgical Neurology International*, vol. 2, no. 1, p. 52, 2011.

-
- [16] S. Chan, F. Conti, K. Salisbury, and N. H. Blevins, “Virtual Reality Simulation in Neurosurgery: Technologies and Evolution,” *Neurosurgery*, vol. 72, no. suppl_1, pp. A154–A164, 2013. [Online]. Available: <https://doi.org/10.1227/NEU.0b013e3182750d26>
- [17] J. Lemole G. Michael, P. P. Banerjee, C. Luciano, S. Neckrysh, and F. T. Charbel, “Virtual Reality in Neurosurgical Education: Part-task Ventriculostomy Simulation with Dynamic Visual and Haptic Feedback,” *Neurosurgery*, vol. 61, no. 1, pp. 142–149, 2007. [Online]. Available: <https://doi.org/10.1227/01.neu.0000279734.22931.21>
- [18] K. G. V. James D. Westwood, Randy S. Haluck, Helene M. Hoffman, Greg T. Mogel, Roger Phillips, Richard A. Robb, *Studies in Health Technology and Informatics*, 2005, vol. 119. [Online]. Available: %3CGo%0Ato
- [19] F. Tendick, M. Downes, T. Goktekin, M. C. Cavusoglu, D. Feygin, X. Wu, R. Eyal, M. Hegarty, and L. W. Way, “A virtual environment testbed for training laparoscopic surgical skills,” *Presence*, vol. 9, no. 3, pp. 236–255, June 2000.
- [20] K. G. V. James D. Westwood, Randy S. Haluck, Helene M. Hoffman, Greg T. Mogel, Roger Phillips, Richard A. Robb, *Studies in Health Technology and Informatics*, 2005, vol. 111.
- [21] H. Çakmak, H. Maaß, C. Trantakis, G. Strauß, E. Nowatius, and U. Kühnapfel, “Haptic ventriculostomy simulation in a grid environment,” *Computer Animation and Virtual Worlds*, vol. 20, no. 1, pp. 25–38, 2009. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1002/cav.272>
- [22] T. R. Coles, N. W. John, D. Gould, and D. G. Caldwell, “Integrating Haptics with Augmented Reality in a Femoral Palpation and Needle Insertion Training Simulation,” *IEEE Transactions on Haptics*, vol. 4, no. 3, pp. 199–209, 2011.
- [23] C. M. Schirmer, J. B. Elder, B. Roitberg, and D. A. Lobel, “Virtual reality-based simulation training for ventriculostomy: An Evidence-based approach,” *Neurosurgery*, vol. 73, no. SUPPL. 4, pp. 66–73, 2013.
- [24] C. Luciano, P. Banerjee, G. M. Lemole, and F. Charbel, “Second generation haptic ventriculostomy simulator using the ImmersiveTouch™ system,” *Studies in Health Technology and Informatics*, vol. 119, pp. 343–348, 2006.

BIBLIOGRAFÍA

- [25] M. I. P. H. Cobb, J. M. Taekman, A. R. Zomorodi, L. F. Gonzalez, and D. A. Turner, “Simulation in Neurosurgery—A Brief Review and Commentary,” *World Neurosurgery*, vol. 89, pp. 583–586, 2016. [Online]. Available: <http://dx.doi.org/10.1016/j.wneu.2015.11.068>
- [26] N. S. Ottosen and H. Petersson, *Introduction to the Finite Element Method*. Prentice-Hall, 1992.
- [27] O. Zienkiewicz, R. Taylor, and J. Zhu, *The Finite Element Method: Its Basis and Fundamentals*. Butterworth-Heinemann, 2005.
- [28] B. Siciliano and O. Khatib, *Springer Handbook of Robotics*, 2nd ed. Springer Publishing Company, Incorporated, 2016.
- [29] F. P. Vidal, N. W. John, A. E. Healey, and D. A. Gould, “Simulation of ultrasound guided needle puncture using patient specific data with 3D textures and volume haptics,” *Computer Animation and Virtual Worlds*, vol. 19, no. 2, pp. 111–127, may 2008. [Online]. Available: <http://doi.wiley.com/10.1002/cav.217>
- [30] C. Richard, M. R. Cutkosky, and K. MacLean, “Friction identification for haptic display,” *American Society of Mechanical Engineers, Dynamic Systems and Control Division (Publication) DSC*, vol. 67, pp. 327–334, 1999.
- [31] D. Karnopp, “Computer Simulation of Stick-Slip Friction in Mechanical Dynamic Systems,” *Journal of Dynamic Systems, Measurement, and Control*, vol. 107, no. 1, pp. 100–103, 1985. [Online]. Available: <https://doi.org/10.1115/1.3140698>
- [32] K. B. Shimoga, “A Survey of Perceptual Feedback Issues in Dexterous Telemanipulation. II. Finger Touch Feedback,” in *Proceedings of the 1993 IEEE Virtual Reality Annual International Symposium*, ser. VRAIS '93. Washington, DC, USA: IEEE Computer Society, 1993, pp. 271–279. [Online]. Available: <http://dx.doi.org/10.1109/VRAIS.1993.380769>
- [33] D. C. Ruspini, K. Kolarov, and O. Khatib, “The haptic display of complex graphical environments,” *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1997*, pp. 345–352, 1997.
- [34] W. Dangxiao, G. Yuan, L. Shiyi, Z. Yuru, X. Weiliang, and X. Jing, “Haptic display for virtual reality: progress and challenges,” *Virtual Reality & Intelligent Hardware*, vol. 1, no. 2, p. 136, 2019. [Online]. Available: <https://doi.org/10.3724/SP.J.2096-5796.2019.0008>

-
- [35] F. Faure, C. Duriez, H. Delingette, J. Allard, B. Gilles, S. Marchesseau, H. Talbot, H. Courtecuisse, G. Bousquet, I. Peterlik, and S. Cotin, *SOFA: A Multi-Model Framework for Interactive Physical Simulation*, 2012. [Online]. Available: http://link.springer.com/10.1007/8415_2012_125
- [36] E. G. Corzo Gómez, O. L. Gómez Díaz, S. Serrano Gómez, M. Saavedra Martínez, and L. C. Rojas Arenas, “Elementos morfológicos implicados en la enseñanza de las competencias para lograr una adecuada venopunción periférica,” *International Journal of Morphology*, vol. 36, no. 1, pp. 159–163, 2018.
- [37] A. Bodenham, S. Babu, J. Bennett, R. Binks, P. Fee, B. Fox, A. J. Johnston, A. A. Klein, J. A. Langton, H. Mclure, and S. Q. M. Tighe, “Safe vascular access 2016 guidelines,” *Anaesthesia*, no. May, pp. 1–13, 2016.
- [38] C. Geuzaine and J.-F. Remacle, “Gmsh: a three-dimensional finite element mesh generator with built-in pre-and post-processing facilities,” Tech. Rep., 2009.
- [39] K. Mostofi and R. K. Khouzani, “Surface anatomy for implantation of external ventricular drainage: Some surgical remarks,” *Surgical Neurology International*, vol. 7, no. 23, pp. S577–S580, 2016.
- [40] R. Muralidharan, “External ventricular drains: Management and complications,” *Surgical Neurology International*, vol. 6, no. 7, pp. S271–S274, 2015.
- [41] D. R. Huyette, B. J. Turnbow, C. Kaufman, D. F. Vaslow, B. B. Whiting, and M. Y. Oh, “Accuracy of the freehand pass technique for ventriculostomy catheter placement: Retrospective assessment using computed tomography scans,” *Journal of Neurosurgery*, vol. 108, no. 1, pp. 88–91, 2008.
- [42] S. T. Vite, “Modelado computacional de cortes de tejido blando,” Master’s thesis, Universidad Nacional Autónoma de México, 2012.
- [43] D. Baraff and A. Witkin, “Large steps in cloth simulation,” *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1998*, pp. 43–54, 1998.
- [44] M. Hestenes and E. Stiefel, “Methods of conjugate gradients for solving linear systems,” *Journal of Research of the National Bureau of Standards*, vol. 49, no. 6, p. 409, 1952.

BIBLIOGRAFÍA

- [45] R. L. Burden and J. D. Faires, *Numerical Analysis*, 4th ed., ser. The Prindle, Weber and Schmidt Series in Mathematics. Boston: PWS-Kent Publishing Company, 1989.
- [46] C. Kelley, S. for Industrial, and A. Mathematics, *Iterative Methods for Linear and Nonlinear Equations*, ser. Frontiers in Applied Mathematics. Society for Industrial and Applied Mathematics, 1995.