

## PREPROCESADOR PARA ELVIRA

## PREPROCESSOR FOR ELVIRA

**Valentin Coluccio<sup>a</sup>, Felipe D. Paladea<sup>a</sup>, Ana L. Maldonado<sup>a</sup> y Elvio A. Heidenreich<sup>a</sup>**

<sup>a</sup>*Universidad Nacional de Lomas de Zamora, Facultad de ingeniería. Lomas de Zamora, Argentina,  
eheidenreich@ingenieria.unlz.edu.ar, <https://ingenieria.unlz.edu.ar/>*

**Palabras clave:** Elvira, SALOME, Preprocesado, Interfaz gráfica.

**Resumen.** Este trabajo presenta una herramienta de interfaz gráfica orientada a la configuración de archivos de simulación para el programa Elvira, desarrollada e integrada dentro del entorno de preprocesado SALOME. El módulo permite realizar de forma sistemática y validada cada sección del archivo, incluyendo la importación de mallas, la definición de materiales, la creación de propiedades específicas tanto para nodos como para elementos, las cuales pueden ser asignadas a grupos definidos por el usuario. Además, permite la configuración intuitiva de los modos de reinicio, la incorporación de máscaras para parámetros nodales, la selección del esquema de integración y del solucionador numérico (solver), así como la definición de estímulos aplicados sobre grupos específicos. El sistema permite establecer las diversas modalidades de escritura sobre nodos y elementos, así como la exportación de un único archivo con toda la información de la simulación o múltiples archivos organizados por categorías (nodos, elementos, fibras, etc.) en un directorio. De este modo, la herramienta permite optimizar sustancialmente los tiempos y minimizar la necesidad de recursos adicionales en la preparación de archivos de simulación.

**Keywords:** Elvira, SALOME, Preprocesor, Graphical Interface.

**Abstract.** This work presents a graphical interface tool for configuring simulations files for the Elvira program, developed and integrated within the SALOME preprocessing environment. The module allows for systematic and validated configuration of each section of the file, including mesh import, material definition, and creation of specific properties for both nodes and elements, which can be assigned to user-defined groups. Furthermore, it allows for intuitive configuration of restart modes, the incorporation of masks for nodal parameters, the selection of the integration scheme and the numerical solver, and the definition of stimuli applied to specific groups. The system allows for different writing modes for nodes and elements, as well as the export of a single file with all the simulation information or multiple files organized by categories (nodes, elements, fibers, etc.) in a directory. In this way, the tool substantially optimizes time and minimizes the need for additional resources in the preparation of simulation files.

## 1 INTRODUCCIÓN

El modelado computacional de la actividad eléctrica cardíaca es una herramienta clave en el estudio de enfermedades del corazón, desarrollo de fármacos y en investigaciones académicas. Elvira es un simulador desarrollado específicamente para dichos fines, permitiendo simular la propagación del potencial de acción en tejidos cardíacos a partir de modelos numéricos.

Elvira es un software de simulación desarrollado por Elvio Heidenreich en la universidad de Zaragoza para su tesis doctoral ([Heidenreich Elvio A., 2009](#)), orientado al estudio de la electrofisiología cardíaca mediante el método de elementos finitos. El programa resuelve numéricamente el modelo de reacción-difusión de tipo monodominio para modelar la propagación del potencial de acción a través de tejido cardíaco. Permite la incorporación de geometrías complejas, la asignación de propiedades heterogéneas de materiales (representando diferentes tipos de tejido o condiciones patológicas) y la integración de modelos iónicos celulares detallados, como los de Ten Tusscher, Bueno-Orovio o Grandi, etc. Su objetivo principal es servir como plataforma flexible para la investigación “in silico” de arritmias cardíacas, efectos de fármacos y terapias de resincronización.

ELVIRA ha sido ampliamente adoptado en entornos académicos y de investigación, lo que respalda su relevancia dentro del modelado cardíaco computacional. En la Universitat Politècnica de València (UPV) se ha utilizado en simulaciones tridimensionales de arritmias ventriculares y estudios personalizados de taquicardia postinfarto, donde sirvió como solucionador principal para resolver el modelo de monodominio con el método de elementos finitos ([López-Pérez et al., 2019](#); [López-Pérez, Sebastián y Ferrero, 2015](#)). En la Universidad de Valencia se empleó ELVIRA para realizar simulaciones electrofisiológicas para tesis doctoral, documentando la discretización espacial, temporal y el hardware utilizado ([Godoy, 2017](#)). Por su parte, en la Universidad de Zaragoza (UNIZAR) se ha aplicado ELVIRA en simulaciones de fibras cardíacas y propagación eléctrica en tejido realista, consolidando su uso en trabajos de posgrado y proyectos de investigación ([Fernández, 2019](#)). Asimismo, en el marco del congreso Computing in Cardiology, [Serra et al. \(2021\)](#) compararon los resultados de su herramienta Arrhythmic3D con los obtenidos mediante ELVIRA, validando su precisión y eficiencia en mallas de millones de elementos. Estas aplicaciones institucionales y científicas demuestran que ELVIRA constituye una plataforma consolidada y validada para la simulación biofísica cardíaca, con un amplio reconocimiento en la comunidad científica internacional.

Sin embargo, preparar los datos de entrada para Elvira puede convertirse en una tarea ardua, ya que requiere una combinación de herramientas como editores de malla, códigos personalizados y una correcta interpretación de la estructura del archivo de entrada. Con el objetivo de simplificar dicho proceso, se desarrolló un módulo de preprocesamiento integrado en SALOME 9.13 ([Salome project, 2025](#)). Este módulo centraliza todas las etapas necesarias para la creación del archive listo para la simulación.

Este artículo describe el diseño, implementación y validación de dicho modulo, destacando su utilidad en contextos académicos e investigativos, y su potencial de extensión futura.

## 2 ENTORNO DE EJECUCION Y LENGUAJE DE PROGRAMACION

Para iniciar un análisis computacional, es fundamental generar una malla del objeto de estudio. Por ello, se optó por implementar el preproceso en un programa de mallado que admitiera módulos externos y fuera de código abierto, favoreciendo la colaboración y el desarrollo futuro por parte de otros profesionales. Como alternativas principales surgieron SALOME PLATFORM y Gmsh, cuyas características se resumen cualitativamente en la [Tabla 1](#).

| Característica   | Gmsh                           | SALOME   |
|------------------|--------------------------------|--|
| Interfaz gráfica | Simple y ligera                | Completa, Estilo CAD                                       |
| CAD              | Básico (Lenguaje propio: .geo) | Completo (GEOM, SHAPER)                                    |
| Control de malla | Mediante código                | Visual y mediante código                                   |
| Postproceso      | Resultados numéricos ligeros   | Resultados avanzados gracias a su integración con ParaView |

Tabla 1: Comparación cualitativa entre el software Gmsh y SALOME.

Luego de analizar ambas opciones, se optó por la selección de SALOME. Tomando como puntos sobresalientes su sistema CAD completo, que permite crear las geometrías y las mallas de manera completa y la visualización de cada paso; la posibilidad de programar en Python, lo que permite utilizar un lenguaje que va en ascenso y que no depende de constantes compilaciones; y la disponibilidad de un módulo que funcione como referencia y orientación para la creación de un módulo similar.

### 3 DISEÑO DEL MODULO

El módulo fue desarrollado como un conjunto de funciones y clases en PyQt5, que es una biblioteca de Python que permite crear aplicaciones con interfaces gráficas de usuario (GUI) con la cual trabaja SALOME; accesibles a través de un panel dedicado, donde cada ícono representa una funcionalidad independiente. El orden de los botones en la interfaz refleja la secuencia de las secciones en el archivo de salida, garantizando un flujo de trabajo coherente y facilitando la ejecución de los pasos según lo previsto por el software Elvira. Las herramientas incluyen: importación de malla, creación de materiales y propiedades, asignación a nodos y a elementos, configuración de reinicio, resolutor, estímulos, salidas y exportación. (Figura 1, Figura 2). Esta organización también mejora la claridad y trazabilidad, permitiendo identificar con rapidez el origen de errores y compartimentar el proceso para depuración y validación.

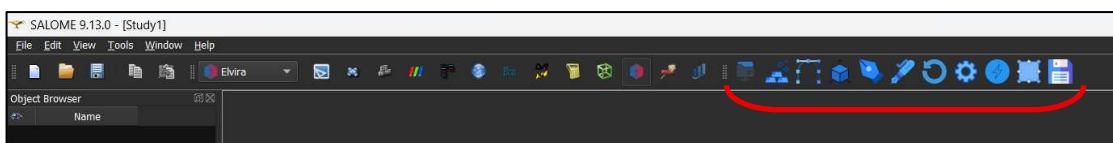


Figura 1: Interfaz del módulo de preprocesado para ELVIRA integrada en SALOME 9.13.

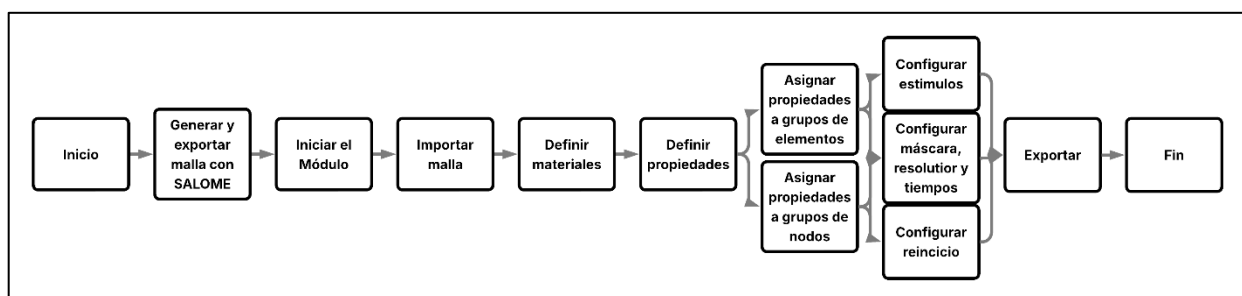


Figura 2: Diagrama de flujo de interacción entre el usuario y el módulo de preprocesado para ELVIRA

### 3.1 Configuración de entrada (Importación de malla, materiales, propiedades, asignación)

La configuración de entrada reúne las herramientas iniciales del módulo, orientadas a preparar los datos fundamentales que alimentarán la simulación. Desde la importación de la malla hasta la asignación de materiales y propiedades, estas funciones garantizan la coherencia estructural y física del modelo antes de su exportación final.

En SALOME, se genera la geometría y la malla según las necesidades del análisis, y posteriormente se exporta en formato Universal File Format (.unv) para habilitar el módulo e importar la malla. La elección de este formato se fundamenta en la riqueza de información que proporciona: todas las propiedades de la malla se registran de forma modular, incluyendo la identificación de nodos y elementos, sus coordenadas y los grupos a los que pertenecen. La importación del archivo se realiza mediante el explorador de archivos en Windows o Linux, guardando el archivo seleccionado y su ruta en variables independientes.

A continuación, la definición de materiales se efectúa ingresando su nombre, conductancia, la relación entre conductancia transversal y longitudinal, y la capacitancia. La interfaz permite visualizar cada material junto con sus propiedades y eliminarlo en caso de error. La información se almacena en un diccionario, en el que las claves externas corresponden al nombre de cada material y están asociadas a diccionarios individuales que contienen los valores de sus tres propiedades.

Posteriormente, la creación de propiedades tanto para los nodos como para los elementos consisten en partes similares. Ambas necesitan el ingreso del nombre de la propiedad, el material previamente creado y la posibilidad de especificar la dirección de las fibras, en caso de que sea aplicable. Las opciones son: cero (0) para dejarlo por defecto, dos (2) para direcciones en el plano y tres (3) para direcciones en el espacio. A su vez, en las propiedades de los nodos debemos especificar el modelo celular con el cual se trabajará. Actualmente, se dispone de los modelos incluidos en la versión base de Elvira, entre los que se encuentran “Roger y McCulloch”, “tenTusscher ENDO, MID y EPI”, “Luo Rudy ENDO, MID y EPI”, “Bueno-Orovio ENDO, MID y EPI”, “Nygren”, “Purkinje Stewart”, “Grandi ENDO y EPI”, “CRLP ENDO y EPI”, “Grandi\_HF ENDO y EPI”, “MacCanell Fibroblast”, “Malekar” y “Ohara ENDO, MID y EPI”. Ambas propiedades se guardan en diccionarios con una estructura similar a la utilizada para los materiales.

Finalmente, para asignar propiedades a nodos y elementos, estos deben estar agrupados al exportar la malla. Los grupos pueden recibir cualquier nombre y se muestra al acceder a las ventanas de asignación de propiedades. Se despliega una lista con todas las propiedades disponibles, permitiendo al usuario seleccionar la correspondiente; en ausencia de grupos, se crea uno que contenga todos los nodos o elementos. Para los bordes entre propiedades, se recomienda formar un grupo con los nodos fronterizos y asignarles una propiedad específica que combine las características de cada caso. La información se almacena en dos diccionarios, cuyas claves son los nombres de los grupos y cuyos valores corresponden al número de propiedad, asignado según su orden de creación. Esta estructura facilita los pasos posteriores para generar el archivo final.

### 3.2 Configuración de simulación (reinicios, máscara, solucionador, estímulos)

La configuración de simulación agrupa las herramientas destinadas a definir el comportamiento dinámico del modelo, incluyendo los modos de reinicio, la aplicación de máscaras, la selección del solucionador numérico y la creación de estímulos eléctricos. Estas funciones permiten adaptar el entorno de ejecución a distintos escenarios experimentales, garantizando coherencia entre parámetros temporales y condiciones iniciales.

En primer lugar, Elvira dispone de seis tipos de reinicio, numerados del menos uno (-1) al cuatro (4), que se seleccionan mediante un desplegable que muestra el número asociado y una breve descripción. Cada opción requiere información específica, la cual se habilita en la ventana según la selección del usuario una vez que la elige. La opción por defecto no genera archivo de reinicio por lo cual no necesita datos adicionales, manteniendo la ventana sin cambios. La opción de leer un archivo de reinicio, con el fin de restaurar un estudio en proceso, requiere la ruta de un archivo previamente creado, la cual se almacena mediante un botón de selección. La generación de uno o varios archivos de reinicio permite establecer los pasos en que se crearán, ingresando primero la cantidad de archivos y luego los pasos correspondientes a través de celdas dinámicas. La opción combinada de leer y generar archivos permite partir de un archivo existente y producir uno nuevo posteriormente. Asimismo, se puede generar archivos de reinicio cada  $n$  pasos de tiempo, ingresando el valor de los pasos en una celda, o combinar esta funcionalidad con la lectura de un archivo existente. Toda la información de reinicio se guarda en un diccionario, cuyas claves incluyen la opción seleccionada, la ruta del archivo (si corresponde), los pasos de creación de los archivos (si corresponde, en forma de lista) y el incremento de pasos de tiempo para generar archivos (si aplica).

A continuación, la ventana de configuración permite establecer la máscara y definir los parámetros del solucionador. La máscara se carga desde un archivo previamente generado mediante un botón, y su ruta se registra en una variable. El solucionador requiere seleccionar primero el esquema de integración, identificado por un número del cero (0) al tres (3), mediante un desplegable que incluye una breve descripción; el valor seleccionado se guarda en una variable. La elección del tipo de solucionador se realiza de manera similar, con números del uno (1) al seis (6) y su correspondiente descripción, almacenando el valor seleccionado. Finalmente, los parámetros temporales se configuran mediante tres celdas en las que se ingresan el incremento inicial, el incremento mínimo y el tiempo de simulación, todos en milisegundos, y se guardan en una lista.

Finalmente, en la sección de estímulos se crean y asocian los mismos, siendo aplicables tanto a nodos individuales como a grupos de nodos. Primero, se asigna un nombre al estímulo y se determina su aplicación: a un nodo mediante un número de identificación válido o a un grupo de nodos seleccionado desde un desplegable que queda disponible al escoger esta opción. A continuación, se cargan las propiedades del estímulo, organizadas en grupos de tres (tiempo de inicio, duración y corriente). Cada estímulo puede contener múltiples grupos; por ello, se ingresa primero la cantidad de grupos y luego se habilitan las celdas necesarias para introducir los valores correspondientes. Los estímulos se almacenan en diccionarios, donde las claves externas son los nombres de los estímulos y cada uno contiene un diccionario asociado con los nodos de aplicación y los parámetros definidos.

### 3.3 Configuración de salida (Frecuencias de escritura y exportación)

La configuración de salida agrupa las herramientas destinadas a definir cómo se registran y exportan los resultados de la simulación. Incluye tanto la especificación de las frecuencias de escritura de datos como la generación de los archivos finales requeridos por Elvira.

El último parámetro por configurar corresponde a las frecuencias de escritura de los resultados del análisis, aplicables a nodos y/o elementos. La ventana permite seleccionar un grupo de nodos mediante un desplegable y definir, mediante dos celdas numéricas, la frecuencia de escritura y la bandera de postproceso. Mediante una casilla de verificación, se habilita la misma configuración para un grupo de elementos. A su vez, el usuario puede elegir, mediante dos casillas de verificación, si desea registrar en la salida general el potencial, la corriente axial o ambos, cargando las frecuencias de escritura correspondientes. Todos los datos se almacenan en un diccionario, donde las claves indican la selección de las casillas mediante variables



booleanas y los valores corresponden a los datos numéricos ingresados.

Por último, el flujo de trabajo de la exportación del módulo de preprocesado para Elvira se inicia con la importación de mallas en formato Universal File Format (.unv), un estándar ampliamente adoptado para el intercambio de geometrías y discretizaciones entre distintas herramientas de preprocesado. Para ello, se implementa una rutina encargada de interpretar el contenido del archivo y reconstruir las estructuras de datos necesarias para el posterior procesamiento en Elvira, en adelante denominado “parser”. Este parser analiza el archivo línea por línea, identificando secciones mediante los códigos estándar de UNV, incluyendo la lectura de nodos y sus coordenadas tridimensionales, la lectura los elementos finitos con sus nodos asociados, y la interpretación de grupos de nodos y elementos que posteriormente se emplean para la asignación de propiedades o la selección de dominios de estímulo y salida. Cada bloque se descompone en unidades individuales de información, como identificadores o coordenadas, y se convierte al tipo de dato correspondiente, organizándose en estructuras de alto nivel que mantienen la jerarquía lógica del archivo y permiten un acceso directo desde la interfaz gráfica. Asimismo, el parser está diseñado para tolerar líneas en blanco, variaciones menores en el espaciado y formatos de exportación heterogéneos provenientes de distintas versiones de SALOME u otros preprocesadores, asegurando una importación confiable incluso con mallas de diversa procedencia.

El proceso de escritura de datos constituye la etapa central del módulo, ya que genera los archivos de entrada requeridos por Elvira para la ejecución de las simulaciones. Esta funcionalidad se encuentra implementada en una función que articula de manera sistemática la estructura de los distintos bloques de información, garantizando tanto la compatibilidad con el simulador como la consistencia interna de los datos. El procedimiento se organiza de manera jerárquica y secuencial, respetando el orden en que los parámetros son configurados desde la interfaz gráfica, lo que permite al usuario mantener una relación directa entre las acciones realizadas en la GUI y la estructura final del archivo exportado, facilitando la depuración y el análisis en contextos colaborativos o de validación cruzada.

La exportación contempla dos modalidades complementarias. En la modalidad consolidada se genera un único archivo que integra la totalidad de los bloques requeridos por Elvira, lo que resulta adecuado para simulaciones de prueba o de pequeña escala, simplificando la organización de los insumos y minimizando la manipulación de archivos intermedios. En la modalidad distributiva, cada bloque de información se escribe en un archivo independiente dentro de un directorio estructurado, con un archivo central que actúa como eje, favoreciendo simulaciones de mayor complejidad, la reutilización de secciones previamente validadas, la sustitución selectiva de parámetros y la integración de flujos de trabajo que requieren pre o posprocesamiento externo.

En ambos casos, el flujo de escritura sigue una secuencia que comienza con la inicialización de los archivos de destino e inserción de títulos y encabezados de control. A continuación, se vuelcan los datos de los materiales y las propiedades específicas de nodos y elementos, asociadas a los grupos definidos por el usuario y al material correspondiente. Posteriormente se registran los nodos y elementos con sus identificadores, coordenadas y propiedades asignadas. Se escriben luego los parámetros de simulación, incluyendo configuraciones de reinicio, máscaras, esquema de integración del solucionador, tipo de solucionador y tiempos de simulación. Los estímulos definidos por el usuario se contabilizan y organizan en grupos de parámetros, registrándose en el formato requerido por Elvira junto con los nodos a los que están vinculados. Finalmente, se consignan de manera condicional las instrucciones para la escritura de potencial y corriente, así como la selección de nodos y/o elementos a exportar, y se cierra el archivo mediante rutinas controladas que aseguran la consistencia y accesibilidad de los datos. La secuencia general del proceso se resume en el diagrama de la [Figura 3](#).

En conjunto, esta metodología garantiza la correcta preparación de los datos de entrada y proporciona flexibilidad para adaptarse a distintos escenarios de investigación, permitiendo al usuario optar entre rapidez y simplicidad en la modalidad consolidada o modularidad y escalabilidad en la modalidad distributiva, según los requerimientos del caso de estudio.

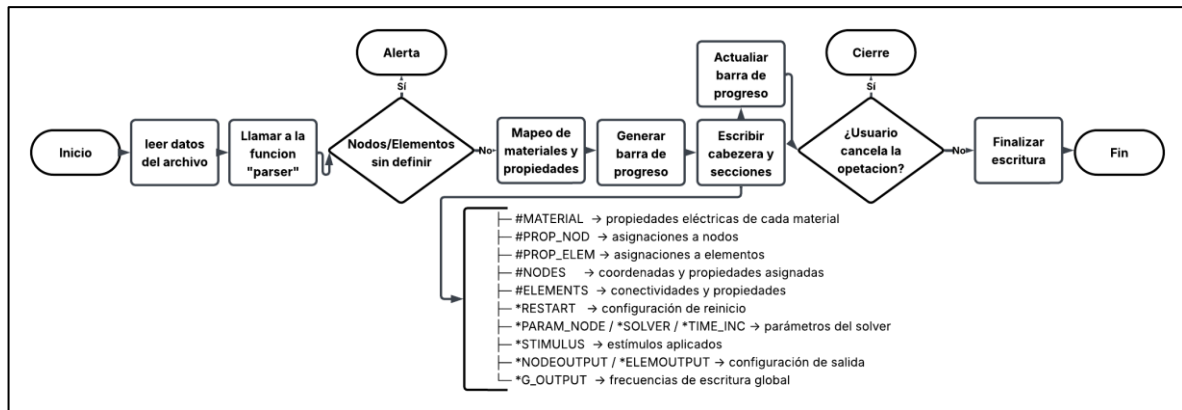


Figura 3: Diagrama de flujo del proceso de exportación de archivos de simulación en el módulo de preprocesado ELVIRA

### 3.4 Validaciones

Con el objetivo de garantizar la confiabilidad y reproducibilidad de los archivos generados, el módulo incorpora un sistema de validaciones internas que opera en distintos niveles de la configuración. Este enfoque evita que se produzcan archivos inconsistentes que podrían comprometer la ejecución de las simulaciones en Elvira. En primer lugar, se asegura que cada material definido cuente con el conjunto completo de propiedades físicas, como conductancia, capacitancia y razón, verificando que los valores sean numéricos, positivos o nulos cuando corresponda, lo que previene inconsistencias dimensionales que invaliden el modelo eléctrico. Asimismo, cada propiedad asignada a nodos o elementos se valida para confirmar que corresponda a un grupo existente, verificando la coherencia entre los identificadores y los grupos definidos en la malla, así como la consistencia de los valores con el dominio físico, evitando asignaciones erróneas o superposiciones no deseadas.

El sistema también comprueba que las referencias realizadas durante la configuración correspondan efectivamente a entidades presentes en la malla importada desde SALOME, ya sea que se trate de la introducción manual de identificadores o de la selección de grupos completos, reduciendo errores comunes de tipeo o de correspondencia entre versiones distintas de la malla. En cuanto a los estímulos definidos por el usuario, se verifica que cada uno tenga un nombre único para prevenir duplicaciones, y que los parámetros asociados sean valores reales y positivos, organizados en tercias de variables específicas ( $A_i$ ,  $B_i$ ,  $C_i$ ), asegurando consistencia matemática y física. Además, antes de generar los archivos de salida, se valida la ruta seleccionada y los permisos de escritura en el directorio correspondiente, evitando fallos en la etapa final del flujo de trabajo.

Finalmente, el sistema comprueba la coherencia de parámetros de configuración como los reinicios, la selección de esquemas de integración, los solucionadores y las frecuencias de salida, asegurando que los valores sean enteros positivos, se encuentren dentro de los rangos esperados y que la combinación de opciones seleccionadas no genere conflictos lógicos en la simulación. Para mantener al usuario informado, se incorporan cuadros de diálogo que notifican en tiempo real cualquier inconsistencia detectada, cumpliendo un doble propósito: prevenir la escritura de archivos inválidos y educar al usuario sobre las restricciones impuestas por el

modelo fisiológico de Elvira. En conjunto, este sistema de validaciones no solo funciona como un mecanismo de control de calidad, sino que también constituye un soporte pedagógico que guía al usuario en la construcción de modelos, reduciendo la necesidad de correcciones posteriores y asegurando la integridad de las simulaciones generadas.

#### 4 RESULTADOS

Finalmente, el módulo desarrollado logró integrarse exitosamente en el entorno SALOME 9.13 como un componente de preprocesado especializado para la preparación de simulaciones en Elvira, aunque sin tener comunicación directa con el mismo. La herramienta fue evaluada mediante diferentes casos de pruebas en dos (2) dimensiones, verificando la correcta ejecución de cada etapa y la coherencia entre los datos configurados en la interfaz gráfica y el archivo final de salida (.dat).

Los resultados obtenidos pueden resumirse en la [Tabla 2](#).

| Funcionalidad  | Estado de implementación | Observaciones   |
|--|--------------------------|---|
| Importación de mallas en formato .unv                  | Validado                 | Lectura correcta de nodos y elementos, sin pérdidas de información.   |
| Gestión de materiales                                  | Validado                 | Creación de materiales con libertad de eliminación.   |
| Gestión de propiedades                                 | Validado                 | Creación de propiedades con vinculación a materiales ya creados y posibilidad de eliminación.                     |
| Asignación de propiedades                              | Validado                 | Asociación de propiedades a grupos de nodos y elementos definidos por el usuario.                                 |
| Modos de reinicio                                      | Validado                 | Selección de los diferentes modos de reinicio para experimentos a largo plazo.                                    |
| Carga de mascara                                       | Validado                 | Carga de dirección de archivo de máscara para mayor configuración de la simulación.                               |
| Configuración de esquema de integración y solucionador | Validado                 | Selección intuitiva de esquemas de integración y solucionador disponibles.  |
| Definición de estímulos                                | Validado                 | Permite crear y asignar estímulos a nodos individuales o a grupos completos creados por el usuario                |
| Gestión de salidas                                     | Validado                 | Selección de escritura de potencial y/o corriente axial sobre nodos y/o elementos, con frecuencias configurables. |
| Exportación de archivos (.dat)                         | Validado                 | Soporta exportación única o múltiple organizada en directorios.   |



|                       |          |  |
|-----------------------|----------|--|
| Validaciones internas | Validado | Alertas sobre campos incompletos, parámetros fuera de rango o erróneos, archivos faltantes y nodos o elementos sin pertenencia a ningún grupo. |
|-----------------------|----------|--|

Tabla 2: Funcionalidades del módulo

Además, como parte del proceso de evaluación del módulo, se realizó una instancia de validación colaborativa junto con miembros del equipo de investigación que desarrollaba un estudio sobre reentrada isquémica en tejido ventricular humano, utilizando el modelo de O'Hara-Rudy y una formulación monodominio para la propagación del potencial de acción. Estos investigadores habían efectuado previamente la configuración de sus simulaciones mediante diversos scripts y herramientas complementarias, lo que permitió efectuar una comparación cualitativa entre ambos enfoques. Tras repetir la configuración con el módulo propuesto, el equipo señaló una disminución estimada del orden del 60 %–70 % en el tiempo de preparación, atribuida a la integración de todas las etapas en una única interfaz que evita alternar entre diferentes códigos y aplicaciones. Asimismo, destacaron que la posibilidad de modificar parámetros y rehacer configuraciones de manera inmediata ante errores o cambios en el modelo representó una mejora sustancial en la agilidad del flujo de trabajo. Aunque esta valoración no surge de una medición cronometrada formalmente, constituye una apreciación cualitativa respaldada por la experiencia de usuarios con conocimiento previo de Elvira, y refuerza la evidencia sobre la mejora en eficiencia y usabilidad aportada por la herramienta desarrollada.

Adicionalmente, cabe destacar que, para nuevos usuarios, el ahorro de tiempo resulta incluso más significativo e incalculable, dado que, en ausencia de esta herramienta, deberían primero comprender en detalle la estructura interna de Elvira, sus formatos de entrada y el modo de preparación de scripts, lo que representa una barrera inicial considerable.

Finalmente, se resalta que el presente trabajo continúa en desarrollo, con el objetivo de incorporar nuevas funcionalidades y optimizaciones. Actualmente, el módulo está siendo probado en entornos académicos por un grupo de cinco doctorandos en España y por colaboradores en Italia, lo que permitirá ampliar la validación y recopilar retroalimentación adicional para futuras mejoras.

## 5 CONCLUSIONES

El desarrollo del módulo de preprocesado para Elvira representa una contribución relevante para la comunidad de investigadores en electrofisiología cardíaca computacional. Su integración en SALOME lo convierte en una herramienta de código abierto, accesible, modular y extensible, pensada para usuarios con conocimientos básicos en simulación numérica, pero sin experiencia en programación. Entre sus principales aportes se incluyen la reducción significativa del tiempo de preparación de simulaciones al eliminar la dependencia de herramientas externas o scripts ad hoc, es decir, programas diseñados específicamente para un caso particular y no reutilizables; la validación interna de parámetros para minimizar errores frecuentes en la preparación manual de archivos, la flexibilidad en la exportación de resultados que permite adaptarse a distintos esquemas experimentales, y la correspondencia directa entre la interfaz gráfica y la estructura del archivo final, garantizando trazabilidad y transparencia en los experimentos. Además, este trabajo establece una base para ampliar el ecosistema de Elvira mediante módulos adicionales; en particular, la futura incorporación de la generación de fibras de Purkinje y la posibilidad de ejecutar simulaciones directamente desde SALOME constituyen

líneas claras de desarrollo a mediano plazo.

## 6 TRABAJOS FUTUROS

El presente desarrollo representa un primer avance hacia un entorno unificado de preprocesado para simulaciones en Elvira. Aun así, se identifican varias líneas de mejora orientadas a ampliar su usabilidad y alcance. Entre ellas, destaca la futura integración completa de Elvira dentro de SALOME, permitiendo ejecutar simulaciones desde la misma plataforma y centralizar el flujo de trabajo. También se prevé habilitar el acceso directo a las mallas cargadas en SALOME, eliminando la exportación intermedia y reduciendo errores y tiempos de transferencia. En paralelo, la optimización del proceso de escritura mediante flujos en memoria o escritura paralela podría disminuir los tiempos de exportación en simulaciones de gran escala. Además, se planea incorporar la representación de fibras de Purkinje para lograr modelos electrofisiológicos más detallados, y extender las validaciones automáticas con verificaciones cruzadas que aseguren la coherencia entre materiales, propiedades y condiciones de estímulo.

## 7 AGRADECIMIENTOS

Los autores agradecen a la Universidad Nacional de Lomas de Zamora y a su Instituto de Investigación por el apoyo brindado para la realización de este trabajo. Asimismo, se reconoce la infraestructura y el entorno de colaboración que hicieron posible el desarrollo del módulo presentado.

## 8 REFERENCIAS

- Fernández, J.C. *New Methodologies for the Development and Validation of Electrophysiological Models*. Universidad de Zaragoza, Zaragoza, España, 2019. <https://zaguan.unizar.es/record/78866/files/TESIS-2019-067.pdf>
- Godoy, E.J. *A Computational Based Approach for Non-invasive Localization of Atrial ectopic foci*. Universitat de València, Valencia, España, 2020. <https://www.uv.es/rasea3/docs/GodoyPhDSmall.pdf>
- Heidenreich, E.A. *Algoritmos para ecuaciones de reacción difusión aplicados a electrofisiología*. Ph.D. thesis, Universidad de Zaragoza, Zaragoza, España, 2009. <https://repositorio.unlz.edu.ar/bitstream/handle/123456789/171/Algor%C3%ADtmos%20para%20ecuaciones%20de%20reacci%C3%B3n%20difusi%C3%B3n.pdf?sequence=1&isAllowed=y>
- López-Pérez, A., Sebastián, R., Ferrero, J.M. *Three-dimensional cardiac computational modelling: methods, features and applications*. BioMedical Engineering OnLine, 14(35):1-31, 2015. <https://doi.org/10.1186/s12938-015-0033-5>
- López-Pérez, A. et al. *Personalized Cardiac Computational Models: From Clinical Data to Simulation of Infarct-Related Ventricular Tachycardia*. Frontiers in Physiology, 10, 2019. <https://doi.org/10.3389/fphys.2019.00580>
- SALOME Platform. The open-source integration platform for numerical simulation, 2025. <https://www.salome-platform.org/>
- Serra, D. et al. *Arrhythmic3D: a fast automata-based tool to simulate and assess arrhythmia risk in 3D ventricular models*. Computing in Cardiology, 48:1-4, 2021. <https://doi.org/10.23919/CinC53138.2021.9662767>