

Refinamiento de Mallas para Simulaciones en SOFA Framework: Equilibrio entre Comportamiento Dinámico y Eficiencia Computacional

Lipa, G.^{1,*}, Rodríguez-Sanz, A.¹, Muñoz, J.¹, Martínez, S.¹, Monje, C. A.¹

RoboticsLab, Dpto. de Ingeniería de Sistemas y Automática, Universidad Carlos III de Madrid, Av. de la Universidad 30, 28911 Leganés, España.

Resumen

Este trabajo investiga el impacto de la resolución de malla en la precisión y eficiencia de las simulaciones de plataformas robóticas blandas utilizando el marco SOFA, ampliamente empleado para simular sistemas físicos y que ofrece flexibilidad para integrar múltiples modelos. Definir la resolución de malla óptima sigue siendo un desafío, ya que las mallas de mayor resolución mejoran la precisión pero aumentan los costes computacionales, por lo que resulta esencial encontrar un equilibrio adecuado entre ambos factores. Para abordar este problema, es necesario determinar la configuración de malla óptima que permita obtener un comportamiento adecuado en la simulación sin comprometer su viabilidad. En este estudio, se comparan diferentes métodos de mado disponibles en Gmsh: `FRONTAL`, `INITIALMESHONLY`, `HXT`, `MMG3D` y `DELAUNAY`. Los resultados obtenidos proporcionan pautas clave para lograr un balance entre la calidad del mado y la eficiencia de las simulaciones en plataformas de robótica blanda.

Palabras clave: Robótica blanda, Robótica, Optimización

Mesh Refinement for Simulations in the SOFA Framework: Balancing Dynamic Behavior and Computational Efficiency

Abstract

This work explores the mesh resolution impact on the accuracy and efficiency of simulations for soft robotic platforms using the SOFA framework, a widely used tool for physical system simulation due to its flexibility in integrating multiple models. Determining the optimal mesh resolution remains a challenge, as higher-resolution meshes improve accuracy but also increase computational costs. Finding a suitable balance between these factors is essential. To address this, we evaluate various meshing methods available in Gmsh, such as `FRONTAL`, `INITIALMESHONLY`, `HXT`, `MMG3D`, and `DELAUNAY`, to identify mesh configurations that provide realistic simulation behavior without compromising feasibility. The results offer valuable guidance for achieving an effective trade-off between mesh quality and simulation performance in soft robotics.

Keywords: Soft robotics, Robotics, Optimization

1. Introducción

En la robótica, se distinguen enfoques basados en estructuras rígidas, orientados a tareas precisas y controladas, y la robótica blanda, que emplea materiales deformables para lograr movimientos más seguros y adaptativos en entornos dinámicos. Esta naturaleza no lineal y continua de los sistemas blandos plantea desafíos únicos tanto en su modelado como en su control, haciendo que la simulación precisa sea una herramienta esencial para su desarrollo. Esta clase de sistemas complejos se simulan por medio del análisis de elementos finitos (FEA),

mediante los cuales se modelan las geometrías tridimensionales en múltiples elementos finitos de menor tamaño (FEM). Entre los entornos de simulación más avanzados para el análisis de elementos finitos destacan COMSOL, ABAQUS, ANSYS y SOFA Framework (Collins et al., 2021). Todos estos programas son paquetes profesionales enfocados al ámbito industrial cuyas licencias de uso tienen precios muy elevados, a excepción de SOFA (Simulation Open Framework Architecture), que se ha consolidado como una alternativa de código abierto muy efectiva para la simulación de robots blandos y sus interacciones con el entorno. En este contexto, destacan varios trabajos rele-

*Autor para correspondencia: glipa@ing.uc3m.es

vantes (Duriez, 2013; Duriez et al., 2016; Coevoet et al., 2017a) que demuestran la utilidad y funcionalidad del framework para modelar dinámicas complejas, validar diseños y facilitar el desarrollo de plataformas blandas sin necesidad de prototipos físicos.

Los FEA siguen un flujo de trabajo determinado que abarca las siguientes fases: definición volumétrica del modelo, configuración del material y definición de las restricciones, configuración de los parámetros del análisis, solución de los sistemas de ecuaciones y post-procesado. De estas fases, la primera es crítica para establecer una simulación del sistema precisa y estable a expensas del coste computacional. En el simulador SOFA, la definición del modelo se estructura en torno a cuatro componentes: la topología, el modelo físico, el método numérico, y la visualización y control.

La topología define la estructura discreta del dominio de simulación, usualmente representada mediante una malla volumétrica compuesta por elementos como tetraedros o hexaedros. El modelo físico establece las leyes físicas que rigen el comportamiento del sistema, como la elasticidad lineal, modelos viscoelásticos o interacción fluido-estructura. Cada uno de los nodos que componen la malla volumétrica se establece como un grado de libertad que debe poseer las propiedades del material. Por otra parte, el método numérico traduce el modelo físico en un sistema de ecuaciones algebraicas, utilizando técnicas como integración explícita o implícita, y métodos de resolución directa o iterativa. Por último, la visualización y control representan el estado del sistema y permite la interacción con dispositivos hápticos o condiciones de frontera dinámicas.

Por otro lado, Gmsh es una herramienta de código abierto que permite crear o importar geometrías complejas y generar mallas estructuradas o no estructuradas con diferentes algoritmos. Ofrece funciones como refinamiento local, suavizado de nodos y control del tamaño y orientación de los elementos, lo que permite ajustar la malla en zonas clave sin aumentar demasiado su complejidad.

Dada su flexibilidad, Gmsh ofrece múltiples estrategias de mallado que pueden influir notablemente en el comportamiento de una simulación, y cuya elección depende del problema al que este enfocado. En este trabajo se estudiarán distintos algoritmos de mallado disponibles en Gmsh, evaluando cómo cada uno afecta a aspectos como la precisión de los resultados, el tiempo de cómputo y la estabilidad numérica. A continuación se describen los principales métodos analizados:

- **INITIALMESHONLY (IMO):** Genera una malla volumétrica inicial rápida al omitir el refinamiento interno mediante la inserción de puntos. Prioriza la definición del contorno y la extensión espacial de la estructura, lo que conlleva a la creación de una malla con elementos internos grandes y de baja definición, presentando una calidad y precisión variables a lo largo del volumen.
- **DELAUNAY:** Utiliza triangulación de Delaunay para generar mallas volumétricas. Este algoritmo busca evitar la creación de triángulos alargados o estrechos en las caras tetraédricas minimizando elementos degenerados, y mejorando la calidad numérica de la malla.

- **FRONTAL (ADVANCING FRONT):** Genera la malla a partir de las superficies, extendiéndose hacia el interior del volumen. Tiende a producir elementos bien alineados con la geometría y de alta calidad local, aunque su costo computacional es mayor.
- **MMG3D(MMG IN 3D):** Técnica de remallado adaptativo basada en métricas anisotrópicas. Permite modificar una malla existente en función de campos escalares (por ejemplo, tensiones o errores numéricos), mejorando la resolución solo donde es necesario.
- **HIGH-QUALITY TETRAHEDRAL EXTRUSION (HTX):** Combina estrategias de Delaunay y frontal para lograr mallas de alta calidad, especialmente útiles para geometrías complejas. Es capaz de preservar detalles finos y generar elementos más uniformes en zonas curvas.

Este trabajo¹ presenta una revisión y comparación de los algoritmos presentes en la herramienta Gmsh para la creación de mallas volumétricas y su optimización con el objetivo de estudiar su impacto en la simulación a nivel de precisión de los resultados, el tiempo de cómputo y la estabilidad numérica. Los resultados obtenidos permitirán optimizar la simulación de articulaciones blandas para el robot humanoide TEO.

2. Descripción del método experimental

A continuación, se describen los pasos a realizar durante la optimización del mallado. Para facilitar la descripción, se utilizarán a lo largo del artículo el caso de uso propuesto, un tobillo blando (Figura 1) actuado por cuatro tendones dispuestos ortogonalmente alrededor del eje central, los cuales se extienden axialmente a lo largo de toda la estructura, similar al funcionamiento empleado en otras articulaciones blandas del robot TEO (Nagua et al., 2021; Continelli et al., 2023). Los tendones se han descrito usando el modelo **CABLECONSTRAINT**, basado en restricciones con multiplicadores de Lagrange, incluido en el plugin **SOFTROBOTS** desarrollado en (Coevoet et al., 2017b).

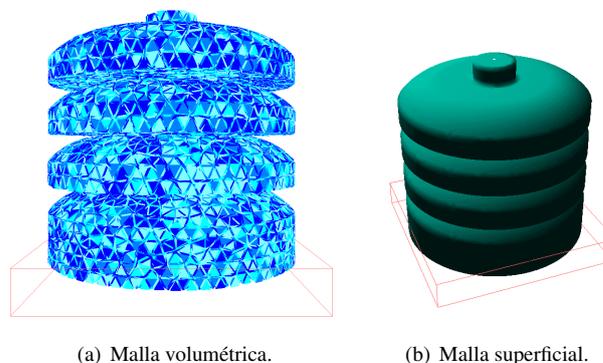


Figura 1: Representación gráfica del modelo del tobillo blando en una escena de SOFA.

¹El código de este trabajo se puede encontrar en: <https://gitlab.com/softroboticslab/sofa-platforms.git>

2.1. Metodología empleada en las comparaciones

Para la evaluación de las métricas seleccionadas en un escenario de simulación activa, se optó por modificar la longitud de uno de los tendones mediante una entrada en desplazamiento definida por una señal cosenoidal. El movimiento continuo y periódico generado con esta señal es ideal para analizar el comportamiento dinámico del sistema bajo estímulos controlados, como vibraciones o ciclos biomecánicos (Figura 2).

La simulación se llevó a cabo durante un intervalo total de 10 segundos, empleando un paso de tiempo constante de $\Delta t = 0,1$ segundos (100 iteraciones), registrándose la evolución temporal del ángulo de interés. La señal utilizada para actuar sobre el tendón se describe en la ecuación (1):

$$f(t) = -4 \cos\left(\frac{2\pi}{5}t\right) + 6 \quad (1)$$

donde t representa el tiempo de simulación. A partir de esta metodología, se generó una base de datos para cada técnica de mallado considerada, asociando el número total de tetraedros de la malla con la amplitud del movimiento simulado. Esto permite evaluar la influencia de la resolución espacial de la malla y del método de generación empleado sobre la precisión y la fiabilidad de la simulación.

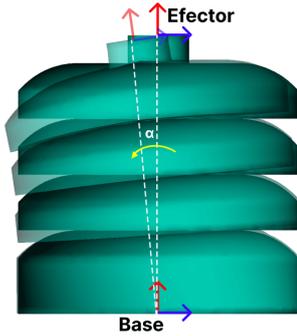


Figura 2: Actuación del tobillo variando la longitud de un cable.

2.2. Evaluación de la precisión dinámica de la malla

A continuación, al igual que en (Ferrentino et al., 2024), para evaluar el impacto del número de elementos en el desempeño dinámico de la simulación, se llevaron a cabo una serie de pruebas de distintos mallados a través de la herramienta CGAL. En dicho artículo se proponen cuatro mallas volumétricas construidas con el teorema de Delaunay (4500, 9300, 16200 y 36000 tetraedros) con las que se busca una convergencia en la amplitud obtenida. En nuestro caso se utilizó la misma metodología para comparar el punto de convergencia obtenido en el modelo del tobillo con todos los algoritmos disponibles en Gmsh. Para ello se realizaron simulaciones consecutivas para cada algoritmo variando el número de tetraedros (Figura 3) en el rango de 500 a 100000 tetraedros.

2.3. Evaluación de calidad de malla

Además del estudio de precisión dinámica en función del número de elementos, se consideró relevante realizar un análisis comparativo de la calidad geométrica de las mallas generadas por cada método. Para ello, se seleccionó una resolución representativa de aproximadamente 40000 tetraedros para comparar

los métodos, ya que a esta escala, la malla presenta sus principales características estructurales sin que ello implique un coste computacional elevado.

La calidad de la malla se puede analizar mediante las siguientes métricas: la relación de aspecto, la distorsión angular y la valencia. La relación de aspecto o ASPECT RATIO (AR) indica la relación entre las longitudes de los lados del tetraedro, donde valores cercanos a 1 indican buena calidad. La distorsión angular o SKEWNESS (SK), por el contrario, mide la distorsión geométrica del elemento. Por último, la VALENCIA (V) marca el número de conexiones por nodo, indicativo de la regularidad topológica.

2.4. Evaluación de la eficiencia computacional del mallado

El mallado influye directamente en los tiempos de simulación al definir la estructura del sistema de ecuaciones a resolver. En esta sección, nos centramos en el impacto del mallado sobre el rendimiento computacional, es decir, en cómo diferentes estrategias de generación de malla afectan a la complejidad del sistema numérico subyacente.

Este impacto está determinado por las *relaciones nodales*, es decir, por la cantidad y distribución de conexiones entre nodos. Una mayor conectividad genera matrices de rigidez más densas (es decir, relacionadas entre sí), lo que incrementa significativamente el coste de resolución. Este fenómeno se refleja en indicadores como la valencia media de los nodos o el número de elementos no nulos de la matriz.

El análisis se enfoca, por tanto, en los procesos computacionales más sensibles a la estructura de la malla, en particular aquellos asociados a la construcción y resolución del sistema de ecuaciones que modela el comportamiento físico de la plataforma. Uno de los componentes más relevantes en este aspecto es la factorización numérica del sistema lineal, que depende directamente de la estructura y propiedades de la matriz de rigidez generada a partir del mallado.

Para evaluar este impacto, se consideran dos indicadores clave: el tamaño de la matriz y el número de elementos no nulos. El tamaño de la matriz (N) define la dimensión del sistema de ecuaciones, directamente relacionado con el número de nodos. Por otra parte, el número de elementos no nulos (L_{nnz}) indica la cantidad de entradas distintas de cero en la parte triangular inferior de la matriz, lo que refleja su grado de conectividad y densidad.

Dentro de SOFA, el tiempo total de simulación se reparte en distintos bloques computacionales. De todos estos bloques, los más sensibles a la estructura del mallado pertenecen al denominado MBK, que agrupa las operaciones asociadas a la ecuación del sistema físico $M\ddot{x} + B\dot{x} + Kx = f$, donde M , B y K son las matrices de masa, amortiguamiento y rigidez, respectivamente.

El proceso MBKBUILD es la etapa en la que se ensamblan las matrices del sistema, especialmente la de rigidez. Incluye procesos como BuildStiffnessMatrix y ProjectedMapMatrix, que son sensibles a la topología del mallado.

El proceso MBKSOLVE es la etapa de resolución del sistema lineal, centrada en la NumericFactorization. Su coste depende fuertemente de la densidad de la matriz, estimada mediante el cociente L_{nnz}/n^2 , donde n es el tamaño de la matriz global.

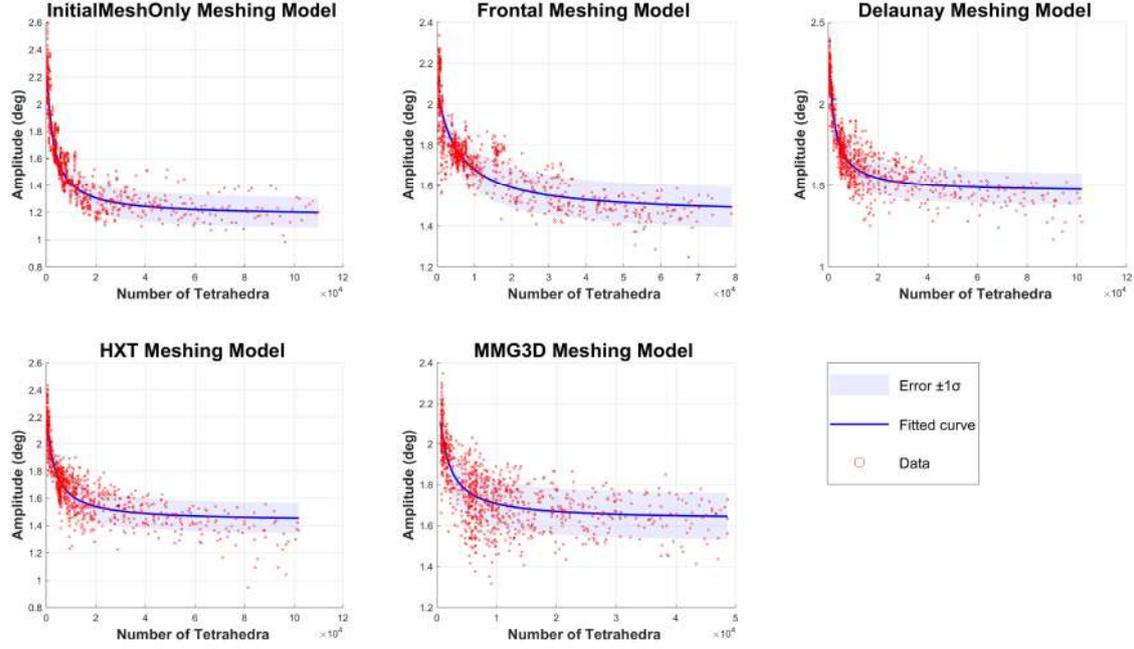


Figura 3: Gráficas de dispersión del ángulo alcanzado frente al número de tetraedros para cada tipo de mallado.

Estos indicadores servirán más adelante como base para el análisis comparativo entre los distintos métodos de mallado, permitiendo evaluar cómo influye cada estrategia en la eficiencia de la simulación. En esta sección se presentan únicamente los conceptos necesarios para contextualizar y motivar dicho análisis.

3. Experimentos de Simulación y Resultados

Una vez planteado el método, se aplicó al caso de uso propuesto para determinar el mallado óptimo en base a las especificaciones. Los experimentos de simulación realizados fueron los siguientes:

3.1. Gráficas de ajuste y dispersión

En la Figura 3 se presentan los distintos gráficos de dispersión obtenidos para cada modelo de mallado, donde cada punto representa el resultado de una simulación. Se observa una tendencia en los datos que replica una función hiperbólica, la cual se modela mediante la siguiente ecuación:

$$f(x) = \frac{a}{x + b} + c \quad (2)$$

donde x es el número de tetraedros, $f(x)$ representa la amplitud del movimiento (ángulo alcanzado), a indica el escalado vertical de la hipérbola, b la asíntota vertical y c la asíntota horizontal. Este último parámetro es el más importante para determinar el valor límite de amplitud al que convergen los métodos, el cual se obtiene mediante la ecuación (3).

$$\lim_{x \rightarrow \infty} f(x) = c \quad (3)$$

Para comparar cuantitativamente los resultados obtenidos con cada método, se analizaron la asíntota horizontal (c), la raíz del error cuadrático medio (RMSE) y el coeficiente de determinación (R^2). Los valores obtenidos de las curvas de ajuste y estas métricas quedan recogidos en la Tabla 1.

Tabla 1: Resumen estadístico de los ajustes por tipo de mallado.

Methods	a	b	c	R^2	RMSE
IMO	3025.2	2530.2	1.17	0.99	0.024
HXT	2506.2	2946.8	1.43	0.80	0.112
MMG3D	899.3	1271.0	1.63	0.58	0.113
Delaunay	1922.7	2090.7	1.46	0.844	0.098
Frontal	3.5729	5.7013	1.4514	0.9678	0.0356

Como se puede observar en la Tabla 1, en los métodos FRONTAL, HXT y DELAUNAY, el coeficiente c se sitúa en torno a una amplitud de 1.45° , lo que sugiere una tendencia de convergencia consistente. En cambio, INITIALMESHONLY y MMG3D alcanzan valores de 1.1728 y 1.6269 , respectivamente, alejándose del comportamiento observado en los demás métodos.

En cuanto al R^2 , el método FRONTAL alcanza el valor más alto (0.9944), seguido de cerca por INITIALMESHONLY (0.9941). En contraste, MMG3D presenta el valor más bajo (0.5801), lo cual evidencia una menor capacidad de ajuste y mayor dispersión en los datos.

Por otro lado, el método FRONTAL presenta el RMSE más bajo (0.0151), seguido por INITIALMESHONLY (0.0240). Por otro lado, HXT y MMG3D superan los 0.11, indicando menor precisión.

Si bien FRONTAL e INITIALMESHONLY muestran buenos resultados en cuanto a ajuste y precisión, es necesario considerar

el contexto físico del modelo. En particular, el método INITIAL-MESHONLY, a pesar de su bajo error y alta estabilidad, converge hacia un valor que no representa adecuadamente el comportamiento esperado del sistema. Esto se debe a que se trata de una malla sin procesos de refinamiento ni densificación, lo que puede limitar su capacidad de describir correctamente la física del problema.

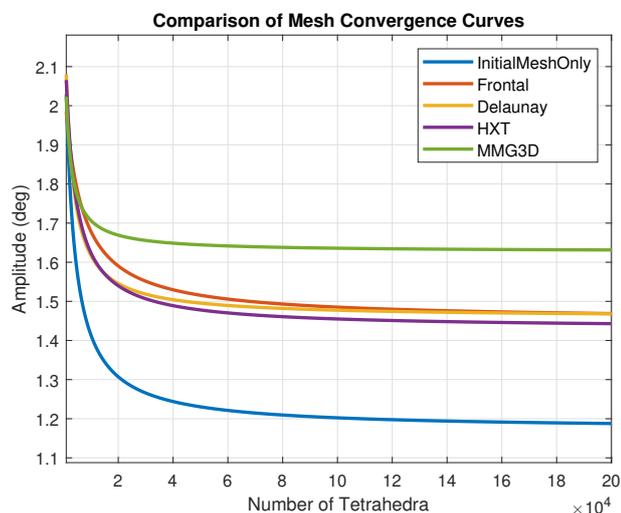


Figura 4: Comparación de las gráficas de ajuste resultantes para cada uno de los métodos de mallado.

Por su parte, MMG3D, aunque utiliza un enfoque adaptativo, depende fuertemente de las condiciones iniciales de la malla. Si éstas no permiten identificar correctamente las zonas relevantes para la simulación, el algoritmo puede fallar al distribuir adecuadamente la densidad de elementos.

La Figura 4 muestra de forma visual el comportamiento de cada método. Las curvas de FRONTAL, HXT y DELAUNAY presentan una tendencia clara de convergencia hacia valores estables, lo cual respalda su validez como modelos representativos. Aunque algunas de ellas, como HXT y DELAUNAY, presentan cierta dispersión, la tendencia general es coherente y razonable dentro del contexto del estudio.

La curva de INITIALMESHONLY es estable y presenta baja dispersión; sin embargo, converge hacia un valor que no es físicamente representativo, lo que limita su utilidad práctica en este contexto.

En conjunto, este análisis pone de manifiesto que no basta con obtener un buen ajuste estadístico; también es fundamental que el modelo converja hacia valores que tengan sentido físico. En este sentido, el método FRONTAL destaca como el más equilibrado, al combinar precisión en el ajuste con una convergencia coherente y un comportamiento adecuado.

3.2. Análisis crítico de las curvas de ajuste temporal

Además de un comportamiento adecuado en cuanto a convergencia, también es crucial la viabilidad de la simulación, para garantizar su aplicación en diversos contextos dentro de la robótica. Para ajustar los datos obtenidos, se ha seleccionado la curva de ajuste $y = a \cdot x^b$.

Aunque el ajuste temporal no proporciona una gran cantidad de información adicional por sí mismo, las discrepancias entre los tiempos de simulación para cada tipo de mallado, a pesar de utilizar el mismo número de tetraedros, son un indicativo claro de cómo el tipo de mallado influye significativamente en el rendimiento temporal de la simulación.

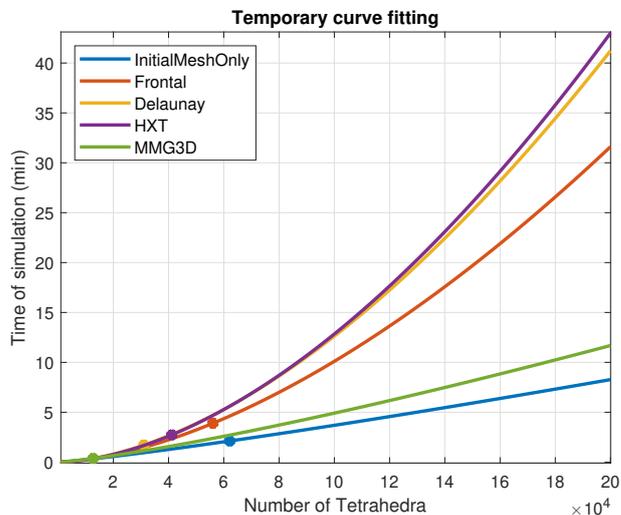


Figura 5: Gráficas de ajuste asociadas a los 5 métodos de mallado.

En la Figura 5 se observa que, para un mismo número de tetraedros, los tiempos de simulación varían considerablemente en función del método de mallado empleado. Además, en cada curva se ha señalado un punto umbral² que representa la convergencia más próxima a la asíntota, con un error del 4 % respecto al valor final.

Estos resultados permiten concluir que la resolución de la malla, por sí sola, no determina el rendimiento computacional. En este contexto, cobra relevancia un factor importante: la calidad geométrica del mallado.

3.3. Evaluación de la calidad de malla

Para estudiar cómo la calidad de la malla afecta el comportamiento de la simulación, se seleccionó una resolución fija que genera aproximadamente 40000 tetraedros por malla. Esta resolución permite comparar de forma justa los distintos métodos sin que el coste computacional sea excesivo. Los resultados obtenidos en esta simulación se resumen en la Tabla 2.

El análisis de la tabla muestra que el método INITIAL-MESHONLY tiene valores extremadamente altos en el ASPECT RATIO promedio (5.18) y máximo (117.99), lo cual indica una gran distorsión en los tetraedros. Esto se complementa con un SKEWNESS alto, lo que refleja una geometría muy deformada. Aunque el tiempo de paso es bajo (0.958s), la baja calidad de malla compromete la estabilidad y precisión de la simulación.

En términos de rendimiento computacional, si nos fijamos en los valores de performance, el tiempo invertido en el ensamblado de matrices ($t_{MBKBUILD}$) es relativamente similar entre los distintos métodos de mallado. Sin embargo, donde se observa una diferencia sustancial es en la resolución del sistema lineal

²Punto a partir del cual el número de tetraedros deja de tener relevancia en la precisión de la simulación

Tabla 2: Comparativa entre calidad geométrica del mallado para un numero de tetraedros 40000, características del sistema numérico y rendimiento computacional.

METHODS	GEOMETRY					NUMERIC SYSTEM			PERFORMANCE(SEC)		
	AR Prom	AR Max	Sk Prom	Sk Max	V Prom	L nnz	N	t_{step}	$t_{MBKBuild}$	$t_{MBKSolve}$	
IMO	5.1876	117.991	0.0929	0.9886	12.58	$3,65 \times 10^6$	32430	0.680	0.306	0.326	
MMG3D	0.8943	1.8510	0.0076	0.7433	14.92	$4,36 \times 10^6$	27174	0.845	0.296	0.515	
Frontal	0.9644	2.4412	0.0019	0.6033	17.31	$4,88 \times 10^6$	23853	0.994	0.248	0.725	
Delaunay	0.8719	1.7898	3.68e-5	0.1787	17.85	$5,82 \times 10^6$	23697	1.378	0.303	1.033	
HXT	0.8775	1.6071	0.0004	0.3649	17.89	$5,92 \times 10^6$	23724	1.418	0.241	1.148	

($t_{MBKSolve}$), donde aquellos métodos con una menor valencia promedio (V_{Prom}) presentan una menor complejidad numérica y, por tanto, tiempos de cómputo más reducidos.

La valencia promedio está intrínsecamente relacionada con la densidad de la matriz de rigidez (L_{nnz}), lo cual afecta directamente al rendimiento del solver durante cada iteración. A partir de los datos de la tabla, se puede establecer un orden de eficiencia computacional: IMO es el más eficiente, seguido de MMG3D, FRONTAL, DELAUNAY y, finalmente, HXT, que resulta el menos eficiente debido a su mayor densidad matricial.

Considerando conjuntamente la precisión, la calidad geométrica y la eficiencia computacional, el método de mallado más prometedor es el FRONTAL. Este logra un buen equilibrio entre la calidad de los elementos, que garantiza resultados numéricamente fiables, y una densidad moderada en la matriz de rigidez, lo que se traduce en un rendimiento computacional competitivo frente a métodos más densos como HXT o Delaunay.

4. Conclusiones

Este artículo ofrece un estudio comparativo detallado para el mallado en simulaciones por métodos de elementos finitos. Se ha demostrado que, si bien una mayor resolución de malla mejora la precisión en la simulación de plataformas robóticas blandas, también incrementa significativamente el coste computacional debido a la mayor densidad de la matriz de rigidez. Además, se presenta un caso de uso, en el cual se determina un algoritmo de mallado y un número de tetraedros óptimo para un tobillo blando de un robot humanoide (TEO). En este caso, el método Frontal se destacó frente a Delaunay y HXT al ofrecer un equilibrio entre la calidad geométrica y la eficiencia computacional, generando mallas bien estructuradas sin afectar drásticamente a los tiempos de simulación. En contraste, los métodos IMO y MMG3D presentaron ventajas en tiempos de cómputo, pero con limitaciones: en el caso de IMO, una baja calidad geométrica; y en el caso de HXT, la ausencia de condiciones iniciales adecuadas generó comportamientos inconsistentes frente al resto de los métodos. A partir de estos resultados, se ha determinado que el algoritmo Frontal es el más eficiente para el modelo propuesto.

Por último, cabe resaltar la importancia de seleccionar cuidadosamente la técnica de mallado según los requerimientos específicos de cada simulación, considerando tanto la precisión de los resultados como la viabilidad en términos de coste computacional, poniendo de manifiesto la importancia de herramientas como la propuesta en este artículo. Como líneas futuras, se propone investigar la influencia de la estructura del modelo en el desempeño numérico, avanzar en criterios sistemáticos para

el diseño de mallas óptimas y explorar herramientas como el plugin Model Order Reduction (Goury and Duriez, 2018) del framework SOFA para reducir el tiempo de simulación a expensas de una ligera reducción en la precisión.

Agradecimientos

Estos resultados de investigación se enmarcan dentro de los proyectos ADAPTA, con referencia PLEC2023- 010218, financiado por MICIU /AEI /10.13039/501100011033, SIROCO, con referencia PID2023-147343OB-I00, financiado por MICIU /AEI /10.13039/501100011033 y por FEDER, UE, y iRoboCity2030-CM, Robótica inteligente para ciudades sostenibles (TEC-2024/TEC-62), financiado por Programas de Actividades I+D en tecnologías de la Comunidad de Madrid.

Referencias

- Coevoet, E., Escande, A., Duriez, C., Feb. 2017a. Optimization-Based Inverse Model of Soft Robots With Contact Handling. *IEEE Robotics and Automation Letters*.
DOI: 10.1109/LRA.2017.2669367
- Coevoet, E., Morales Bieze, T., Largilliere, F., Zhang, Z., Thieffry, M., Sanz Lopez, M., Carrez, B., Marchal, D., Goury, O., Dequidt, J., Duriez, C., Nov. 2017b. Software toolkit for modeling, simulation and control of soft robots. *Advanced Robotics* 31, 1208–1224.
DOI: 10.1080/01691864.2017.1395362
- Collins, J., Chand, S., Vanderkop, A., Howard, D., 2021. A review of physics simulators for robotic applications. *IEEE Access* 9, 51416–51431.
DOI: 10.1109/ACCESS.2021.3068769
- Continelli, N. A., Nagua Cuenca, L. F., Monje, C. A., Balaguer, C., mar. 2023. Modelado de un cuello robótico blando mediante aprendizaje automático. *Revista Iberoamericana de Automática e Informática industrial* 20 (3), 282–292.
DOI: 10.4995/riai.2023.18752
- Duriez, C., 2013. Control of Elastic Soft Robots based on Real-Time Finite Element Method. In: *ICRA 2013 IEEE International Conference on Robotics and Automation*. Karlsruhe, France.
- Duriez, C., Coevoet, E., Largilliere, F., Bieze, T. M., Zhang, Z., Sanz Lopez, M., Carrez, B., Marchal, D., Goury, O., Dequidt, J., Dec. 2016. Framework for online simulation of soft robots with optimization-based inverse model. In: *Proceedings of SIMPAR 2016 conference*. Proceedings of SIMPAR 2016 conference. San Francisco, United States.
- Ferrentino, P., Roels, E., Brancart, J., Terryn, S., Van Assche, G., Vanderborght, B., 2024. Finite element analysis-based soft robotic modeling: Simulating a soft actuator in sofa. *IEEE Robotics & Automation Magazine* 31 (3), 97–105.
DOI: 10.1109/MRA.2022.3220536
- Goury, O., Duriez, C., 2018. Fast, generic and reliable control and simulation of soft robots using model order reduction. *IEEE Transactions on Robotics* 34 (6), 1565 – 1576.
DOI: 10.1109/TR0.2018.2861900
- Nagua, L., Relación, C., Monje, C. A., Balaguer, C., 2021. A new approach of soft joint based on a cable-driven parallel mechanism for robotic applications. *Mathematics* 9 (13).
DOI: 10.3390/math9131468