

Grado en Ciencia e Ingeniería de Datos

Trabajo Fin de Grado

Mallas adaptativas en redes neuronales de grafos para la predicción de variables oceanográficas

José Gabriel Reyes Rodríguez

Tutores

Javier Sánchez Pérez

Giovanny Alejandro Cuervo Londoño

Las Palmas de Gran Canaria

11 de julio de 2025

SOLICITUD DE DEFENSA DE TRABAJO DE FIN DE TÍTULO

D./D^a **José Gabriel Reyes Rodríguez**, autor del trabajo **Mallas adaptativas en redes neuronales de grafos para la predicción de variables oceanográficas**, correspondiente a la titulación **Grado en Ciencia e Ingeniería de Datos**.

SOLICITA

que se inicie el procedimiento de defensa del mismo, para lo que se adjunta la documentación requerida, haciendo constar que

☒ se autoriza / ☐ no se autoriza la grabación en audio de la exposición y turno de preguntas.

Asimismo, con respecto al registro de la propiedad intelectual/industrial del TFT, declara que:

☐ Se ha iniciado o hay intención de iniciarlo (defensa no pública).

☒ No está previsto.

Y para que así conste firma la presente. (fecha en firma electrónica)

El/La estudiante

Fdo. José Gabriel Reyes Rodríguez

A rellenar y firmar **obligatoriamente** por el/la/los/las tutores

En relación con la presente solicitud, se informa: (firmar donde corresponda)

Positivamente (en caso de detección de copia, esta firma quedará invalidada)

Fdo. Javier Sánchez Pérez, Giovanni
Alejandro Cuervo Londoño

Negativamente (justificación en TFT05)

Fdo. Javier Sánchez Pérez, Giovanni
Alejandro Cuervo Londoño

DIRECTORA DE LA ESCUELA DE INGENIERÍA INFORMÁTICA

Mallas adaptativas en redes neuronales de grafos para la predicción de variables oceanográficas

Trabajo Fin de Grado

José Gabriel Reyes Rodríguez

Tutores

Javier Sánchez Pérez

Giovanny Alejandro Cuervo Londoño

Grado en Ciencia e Ingeniería de Datos

Escuela de Ingeniería Informática

Universidad de Las Palmas de Gran Canaria

Las Palmas de Gran Canaria

11 de julio de 2025

Índice general

Índice de figuras	v
Índice de tablas	vii
Agradecimientos	viii
Resumen	ix
Abstract	x
1. Introducción	1
1.1. Motivaciones	3
1.2. Objetivos	4
1.3. Organización del documento	5
2. Estado del arte	7
2.1. Modelado físico del océano	7
2.2. Modelado oceánico con aprendizaje profundo	8
2.2.1. Redes neuronales convolucionales	9
2.2.2. Redes recurrentes	10
2.2.3. Modelos basados en redes neuronales de grafos	11
2.2.4. Aprendizaje profundo guiado por la física	13
3. Metodología y planificación del trabajo	15
3.1. Metodología	15
3.2. Planificación	17
3.2.1. Planificación inicial	17
3.2.2. Tareas realizadas	18
3.3. Recursos empleados	18
4. Conjuntos de datos	21
4.1. Fuentes de datos	21
4.1.1. Copernicus Marine Service	21
4.1.2. Copernicus Climate Change Service	22
4.2. Proceso de descarga de los datos	22
4.3. Procesado de los datos	23
4.4. Representación de la zona de estudio	24
4.5. Creación de las máscaras de la zona de estudio	25
5. Redes neuronales empleadas	31

5.1.	Las redes neuronales de grafos	31
5.2.	El modelo GraphCast	34
5.2.1.	Codificación	34
5.2.2.	Procesamiento	37
5.2.3.	Decodificación	39
5.3.	El modelo Hi-LAM	40
5.3.1.	Procesamiento	45
5.3.2.	Decodificación	47
5.3.3.	Predicción final	48
5.3.4.	Resultados y beneficios	49
5.4.	El modelo SeaCast	50
6.	Creación de las mallas	53
6.1.	Mecanismo de creación de las mallas	53
6.2.	Mallas uniformes	54
6.2.1.	Mallas uniformes estructuradas	55
6.2.2.	Mallas uniformes no estructuradas	57
6.3.	Mallas adaptativas	61
6.3.1.	Mallas adaptativas estructuradas	61
6.3.2.	Mallas adaptativas no estructuradas	61
7.	Configuración de experimentos y resultados	67
7.1.	Hiperparámetros y estrategia de entrenamiento	67
7.1.1.	Función de pérdida	68
7.1.2.	Optimizador	68
7.2.	Evaluación de la influencia estructural de la malla	69
7.3.	Impacto de la reducción de resolución de la malla	75
7.4.	Comparativa de mallas con resolución optimizada	84
8.	Conclusiones y trabajo futuro	91
8.1.	Conclusiones	91
8.2.	Trabajo futuro	92
A.	Competencias específicas cubiertas	95
B.	Objetivos de desarrollo sostenible	97
	Bibliografía	105

Índice de figuras

3.1. Flujo de trabajo de la metodología CRISP-DM. Fuente: IBM Corporation [2024]	16
4.1. Representación de la zona de estudio dentro de la zona IBI	25
4.2. Representación de la zona de estudio	26
4.3. Temperatura superficial del mar (K)	27
4.4. Máscara tierra-agua	28
4.5. Máscara de batimetría	29
4.6. Pesos de la grilla	29
5.1. Estructura de la malla de GraphCast con detalles sobre como los distintos procesos dentro de la arquitectura <i>encode-process-decode</i> se llevan a cabo dentro de la malla. Fuente: [Lam et al., 2022]	35
5.2. Representación del problema de los artefactos encontrados por Oskarsson, donde los puntos blancos se corresponden con nodos de la malla. Fuente: Oskarsson et al. [2023].	41
5.3. Conectividad de un nodo en su propio nivel. Fuente: Oskarsson et al. [2023].	43
5.4. Posicionamiento de nodos entre niveles. Fuente: Oskarsson et al. [2023]. . .	43
5.5. Comparativa de las estructuras de mallas de GC-LAM y Hi-LAM. Fuente: Oskarsson et al. [2023].	44
5.6. Representación del proceso de predicción con aplicación del forzamiento de contorno. Fuente: [Oskarsson et al., 2023]	49
5.7. Imagen representativa del dominio del modelo SeaCast. El color marrón indica el forzante de contorno, el verde la capa superficial del mar y el azul representa las zonas de mayor profundidad. Fuente: [Holmberg et al., 2024]	51
6.1. Malla uniforme estructurada cuadrangular con aristas diagonales cruzadas de 30x30 de resolución	55
6.2. Malla uniforme estructurada cuadrangular con arista diagonal por triangulación de Delaunay de 30x30 de resolución	56
6.3. Malla generada mediante muestreo de nodos aleatorios	59
6.4. Malla generada mediante muestreo del punto más lejano	60
6.5. Comparación de la evolución de las probabilidades asignadas según valor batimétrico normalizado de las funciones de densidad propuestas.	64

6.6.	Mallas adaptativas generadas donde: (a) malla generada según las probabilidades obtenidas de la distribución basada en profundidad inversa suavizada, (b) malla generada según las probabilidades obtenidas de la distribución sigmoide mixto ($\alpha = 0,7$), (c) malla generada según muestreo del punto más lejano ponderado por la distribución basada en profundidad inversa suavizada, (d) malla generada según muestreo del punto más lejano ponderado por la distribución sigmoide mixto ($\alpha = 0,7$)	65
7.1.	Visualización comparativa de las tres mallas jerárquicas generadas (U-malla, R-malla y B-malla) en sus respectivos tres niveles de resolución. Se muestran las estructuras nodales y conexiones para cada configuración evaluada en el experimento sobre la influencia estructural de la malla. . . .	71
7.2.	Evolución del RMSE medio a lo largo del horizonte de predicción de 15 días para cada una de las configuraciones de malla consideradas (U-malla, R-malla y B-malla). Se presentan los errores promedio en tres dominios distintos: (a) toda la región de estudio, (b) zona costera africana y (c) mar abierto. Estos resultados permiten comparar el rendimiento global y local de cada tipo de malla.	72
7.3.	Distribución espacial del RMSE en distintos horizontes de predicción (1, 5, 10 y 15 días) para cada tipo de malla evaluada. Se observa cómo la U-malla presenta errores más localizados en zonas costeras, mientras que la B-malla y la R-malla ofrecen patrones más homogéneos.	73
7.4.	Comparación detallada entre las tres configuraciones de malla (U-malla, R-malla y B-malla) en el último día de predicción. Se muestran tres mapas por cada caso: el RMSE espacial, los valores predichos por el modelo y los valores objetivos del conjunto de test. Esta visualización permite detectar errores localizados y evaluar la fidelidad espacial de las predicciones. . . .	74
7.5.	Comparativa visual (parte 1) de las mallas con mayor resolución utilizadas en el experimento sobre la reducción progresiva de la malla. Cada fila muestra una configuración concreta y cada columna representa un nivel jerárquico (del 1 al 3).	76
7.6.	Comparativa visual (parte 2) de las mallas con mayor resolución utilizadas en el experimento sobre la reducción progresiva de la malla. Esta segunda parte incluye las configuraciones de menor tamaño evaluadas.	77
7.7.	Comparativa visual de las mallas con resoluciones degradadas utilizadas en el mismo experimento. Esta figura complementa la anterior, mostrando configuraciones más compactas. Cada fila representa una malla específica y cada columna, su correspondiente nivel de resolución.	78
7.8.	Evolución del error cuadrático medio (RMSE) a lo largo del horizonte de predicción de 15 días para las configuraciones <code>nce_81_27_9</code> y <code>ce_81_27_9</code> . Se muestran los resultados globales (a), en la región costera (b) y en el mar abierto (c). La comparación permite evaluar el impacto del cruce de aristas diagonales sobre la precisión de las predicciones.	81
7.9.	Distribución espacial del RMSE a distintos días del horizonte de predicción (1, 5, 10 y 15) para las configuraciones <code>nce_81_27_9</code> y <code>ce_81_27_9</code> . Esta representación permite observar diferencias en el comportamiento espacial del error, especialmente en zonas costeras.	82

7.10. Comparativa del error cuadrático medio (RMSE) en el último día del horizonte de predicción para las distintas configuraciones de malla degradadas jerárquicamente. Se muestra la media del RMSE ordenada de menor a mayor en tres regiones: (a) toda la zona de estudio, (b) región costera afectada por el fenómeno de afloramiento, y (c) mar abierto. Esta comparación permite evaluar el impacto de la reducción de resolución en distintos contextos geográficos.	83
7.11. Comparativa visual de las cuatro configuraciones de malla utilizadas en la comparativa de mallas con resolución optimizada. Cada fila representa un modelo diferente y cada columna un nivel de resolución.	85
7.12. Evolución del error cuadrático medio (RMSE) a lo largo del horizonte de predicción de 15 días para las distintas configuraciones de malla con resolución optimizada. Se presentan resultados en tres ámbitos espaciales: (a) toda la región de estudio, (b) la zona costera afectada por el fenómeno de afloramiento, y (c) el mar abierto. Esta representación permite evaluar la capacidad de generalización de cada configuración a lo largo del tiempo y en distintas regiones del dominio.	88
7.13. Distribución espacial del RMSE para los días 1, 5, 10 y 15 del horizonte de predicción, comparando las configuraciones de malla <code>nce</code> , <code>fps</code> , <code>fps_base</code> y <code>fps_mixed_sigmoid</code> . Donde se puede apreciar la eliminación de la presencia de artefactos en las zonas costeras.	89
7.14. Comparativa espacial entre modelos de malla en el último día del horizonte de predicción. Para cada configuración (<code>nce</code> , <code>fps</code> , <code>fps_base</code> y <code>fps_sigmoide mixto</code>) se muestran tres mapas: el RMSE espacial, la predicción del modelo y los datos objetivos. Esta representación permite evaluar visualmente el ajuste del modelo y las diferencias en la distribución del error en función del diseño de la malla.	90

Índice de tablas

3.1.	Fases y tareas desarrolladas en el proyecto	20
4.1.	Rango de coordenadas geográficas	24
5.1.	Detalles de refinamiento de la malla multinivel de GraphCast.	38
7.1.	Resumen estructural de las mallas utilizadas por tipo y nivel jerárquico. Se indica el número total de nodos y aristas en cada uno de los tres niveles para los tres tipos de malla considerados: U-malla (uniforme), R-malla (aleatoria) y B-malla (basada en profundidad).	70
7.2.	Comparativa del tiempo de entrenamiento (en horas y minutos) requerido por cada configuración de malla, así como la mejora porcentual obtenida respecto a la U-malla. Estos resultados reflejan la eficiencia computacional alcanzada por las mallas alternativas sin comprometer la calidad predictiva.	70
7.3.	Características de las mallas por nivel y tipo en el experimento sobre el impacto de la reducción de resolución de la malla.	76
7.4.	Comparativa de los tiempos de entrenamiento (en horas y minutos) y mejora porcentual obtenida respecto a la U-malla original y la malla con aristas cruzadas ce_81_27_9 , en el experimento sobre reducción de resolución.	80
7.5.	Características estructurales de las mallas por nivel utilizadas en el experimento de resolución optimizada.	84
7.6.	Comparativa del RMSE y mejora porcentual por zonas para distintas configuraciones de malla respecto a la malla nce . Se presentan los valores medios del RMSE en el último día de predicción, así como las mejoras relativas (en porcentaje) alcanzadas en distintas subregiones: cabos Blanco, Ghir y Juby, franja costera africana y mar abierto.	87
7.7.	Tiempos de entrenamiento requeridos por cada modelo de malla en la comparativa de configuraciones con resolución optimizada. La duración total (en horas y minutos) refleja que las diferencias entre configuraciones son mínimas, lo que sugiere una elevada estabilidad computacional incluso al introducir mecanismos de densificación espacial.	87
B.1.	Tabla con la valoración del grado de relación del TFG con los ODS, según una escala de 0 a 3 (no procede, bajo, medio, alto).	98

Agradecimientos

Agradecimientos

La finalización de este Trabajo de Fin de Grado no habría sido posible sin el apoyo constante, la guía y el aliento de muchas personas a lo largo de este proceso académico y personal.

En primer lugar, deseo expresar mi más profundo agradecimiento a mis padres, Ángeles y Miguel, por su dedicación incondicional, su confianza y el respaldo constante que me han brindado durante estos años. Su ejemplo y compromiso han sido una fuente constante de inspiración.

Quisiera agradecer también a mis tutores, Javier Sánchez Pérez y Giovanny Alejandro Cuervo Londoño, por su valiosa orientación, su disponibilidad continua y por enriquecer este trabajo con sus sugerencias y experiencia. Su acompañamiento ha sido clave en cada etapa del proyecto.

Agradezco, asimismo, a mis amistades más cercanas por su apoyo emocional, por los momentos de desconexión tan necesarios, y por recordarme la importancia del equilibrio durante esta etapa académica.

A mis compañeros y compañeras de promoción, con quienes he compartido no solo clases y trabajos, sino también un ambiente de compañerismo y colaboración que ha contribuido a hacer de estos años una experiencia enriquecedora e inolvidable.

Por último, a todas aquellas personas que, de una forma u otra, han contribuido a mi desarrollo personal y profesional, mi más sincero agradecimiento.

Resumen

Este Trabajo de Fin de Grado está centrado principalmente en la mejora de la predicción de variables oceanográficas a partir de redes neuronales basadas en grafos, donde nuestro punto de interés se centra en descubrir el impacto que tiene sobre la predicción diferentes tipos de mallas.

En la creación de los diferentes tipos de mallas, nos centraremos principalmente en cambios en la distribución de los nodos así como en explorar además una estrategia de densificación basada en la batimetría de la zona de estudio, donde, previamente, se hará un estudio del impacto que tienen distintas cantidades de nodos en las mallas así como el impacto de cambiar el número de aristas entre niveles.

Abstract

This Bachelor's Thesis is primarily focused on improving the prediction of oceanographic variables using graph-based neural networks, with our main interest centered on discovering the impact that different types of meshes have on prediction accuracy.

In the creation of the various types of meshes, we will mainly focus on changes in the distribution of the nodes, exploring a densification strategy based on the bathymetry of the study area. Before this, a study will be conducted on the impact of varying the number of nodes in the meshes, as well as the effect of changing the number of edges between levels.

Capítulo 1

Introducción

En las últimas décadas, el estudio, monitoreo y predicción de variables oceanográficas se ha convertido en un área de creciente relevancia para múltiples disciplinas, que van desde la oceanografía y la climatología hasta la gestión sostenible de recursos marinos, la navegación o la mitigación de desastres naturales. Variables como la temperatura superficial del mar (SST), la salinidad, las corrientes o el nivel del mar no solo permiten caracterizar el estado dinámico del océano, sino que también juegan un papel clave en la comprensión del sistema climático global y su interacción con los ecosistemas terrestres y marinos.

Por otra parte, la creciente preocupación por el cambio climático, junto con la intensificación de fenómenos extremos como huracanes, olas de calor oceánicas o el aumento del nivel del mar, ha incrementado la demanda de modelos predictivos precisos que permitan anticipar con antelación suficiente el comportamiento de estas variables, donde una predicción más precisa y localizada no solo facilita la toma de decisiones informadas en áreas como la pesca, el transporte marítimo o la conservación marina, sino que también es crucial para la planificación de políticas ambientales a largo plazo.

En el caso concreto de este trabajo, el foco se sitúa en el estudio del fenómeno de afloramiento de aguas profundas (conocido como *upwelling*), especialmente relevante en la región de las Islas Canarias y la costa noroccidental africana, debido a su alto impacto en la productividad biológica y en el equilibrio de los ecosistemas marinos. La aparición de bancos de peces asociados a este fenómeno representa, además, un recurso clave para el sector pesquero canario, por lo que disponer de herramientas que permitan predecir de forma precisa y anticipada los eventos de afloramiento podría resultar de gran valor estratégico para la gestión sostenible de esta actividad económica.

En la actualidad, gracias a los avances tecnológicos y científicos, disponemos de un volumen sin precedentes de datos oceánicos procedentes de sensores *in situ*, satélites, boyas y plataformas de observación remota, donde iniciativas como el programa Copernicus [European Union - Copernicus, 2025] han democratizado el acceso a esta información, ofreciendo datos públicos y abiertos de alta resolución tanto espacial como temporal. Paralelamente, el desarrollo de técnicas de aprendizaje automático y, en particular, del aprendizaje profundo (*deep learning*), ha abierto nuevas posibilidades para modelar sistemas dinámicos complejos como el océano.

No obstante, a pesar de los avances recientes, la predicción de variables oceanográficas

sigue presentando importantes desafíos, siendo uno de ellos la representación del espacio oceánico, de tal manera que los modelos puedan ser capaces de captar tanto relaciones locales como patrones de largo alcance. En este contexto, el uso de redes neuronales sobre grafos (Graph Neural Networks, GNNs) ha emergido como una alternativa prometedora, permitiendo modelar interacciones espaciales más flexibles que las basadas en convoluciones convencionales propias de los modelos de redes neuronales convolucionales (Convolutional Neural Networks, CNN).

Teniendo esto en cuenta, el presente proyecto se sitúa dentro del marco del modelado y la predicción de variables oceanográficas mediante técnicas de aprendizaje profundo, donde el objetivo principal consiste en explorar cómo diferentes representaciones espaciales del océano, basadas en distintas mallas (regulares e irregulares), influyen en el rendimiento de modelos de predicción basados en GNNs. Para ello, hemos desarrollado un modelo de redes neuronales basado en la arquitectura expuesta en GraphCast, donde el foco de la investigación estará centrado en la remodelación de las mallas de grafos subyacentes de tal forma que podamos observar el comportamiento del modelo en base a diferentes mallas, así como analizar cómo diferentes aproximaciones estructurales en la creación de los grafos son capaces de influir en los resultados obtenidos, midiendo el impacto que diferentes variaciones en dichas mallas pueden causar sobre la predicción final de la variable objetivo escogida para este trabajo, siendo esta la SST, donde su elección se debe principalmente a limitaciones en los recursos computacionales disponibles, a los datos disponibles y a la alta importancia de esta variable en la predicción de la aparición de los afloramientos.

A lo largo de esta memoria, se describirán los pasos que han sido necesarios realizar, así como las decisiones tomadas en el desarrollo de este Trabajo Fin de Grado, donde se realizará un estudio del estado del arte a día actual, donde se podrá apreciar la evolución en las técnicas que se han empleado para la predicción de la SST en diferentes regiones del planeta con diferentes enfoques. También se detallará el proceso de obtención y procesamiento de los datos requeridos para la consecución de este trabajo, así como un análisis de los modelos en los que el trabajo estará basado en el modelaje de los datos, donde posteriormente cambiaremos la generación de las mallas por procesos propios tal que imiten la estructura de archivos y formato esperados por el modelo con tal de que la integración sea completa.

En materia del contenido realizado para la culminación de este trabajo, hemos tenido que ser especialmente cuidadosos en la selección del conjunto de datos a utilizar, donde la resolución satelital de los mismos iba a ser de vital importancia para la obtención de unos resultados prometedores. Este fue, de hecho, uno de los primeros retos encontrados durante el desarrollo del proyecto, ya que, debido a la situación geográfica de las Islas Canarias, la disponibilidad de datos de calidad se ha visto notablemente reducida en comparación con otras zonas de interés, como el mar Mediterráneo. Al tratarse de una iniciativa europea, el programa Copernicus ofrece datos con mayor resolución espacial sobre regiones próximas al continente europeo, mientras que en regiones más alejadas, como el Atlántico subtropical, dicha resolución tiende a disminuir.

No obstante, con el fin de entrenar un modelo que representara de forma fiable la dinámica oceánica en la zona de estudio, se llevó a cabo un riguroso proceso de adquisición y tratamiento de datos. Para ello, se emplearon productos satelitales procedentes del servicio marino y del servicio climático del programa Copernicus, incluyendo variables clave como la SST y los componentes zonal y meridional del viento a 10 metros. Los datos

se organizaron en archivos diarios y se estructuraron en secuencias temporales diseñadas para evaluar tanto predicciones a corto como a largo plazo. Adicionalmente, se delimitó una región geográfica concreta que abarca las Islas Canarias, Madeira y parte de la costa noroccidental africana, y se construyeron diversas máscaras auxiliares que incorporan información sobre batimetría, separación tierra-agua y ponderación espacial según la latitud. Este procesamiento cuidadoso permitió adaptar los datos a las necesidades del modelo, garantizando una representación coherente y precisa de los fenómenos oceánicos de interés.

En cuanto al modelado de los datos, se ha optado por seguir la estructura propuesta por Google DeepMind en su modelo GraphCast [Lam et al., 2022], combinada con el enfoque presentado en el modelo Hi-LAM [Oskarsson et al., 2023], el cual introduce una arquitectura basada en múltiples niveles de conectividad espacial. Este último propone el uso de mallas con resoluciones espaciales diferenciadas, ajustando el alcance de las aristas en función del nivel jerárquico, lo que permite capturar de forma más efectiva las interacciones de corto y largo alcance en el entorno oceánico. La combinación de ambas estrategias facilita la integración de la influencia espacial del océano en distintas escalas de interacción, aportando mayor flexibilidad y precisión al modelado.

Para la validación de los resultados obtenidos por el modelo, se ha empleado la métrica de la raíz cuadrada del error cuadrático medio (Root Mean Square Error, RMSE), ampliamente reconocida y utilizada en el ámbito de la modelización numérica y la predicción de variables oceanográficas. Esta métrica permite cuantificar la desviación media entre las predicciones generadas por el modelo y los valores reales, proporcionando así una medida directa y objetiva de su precisión.

Por otra parte, para alcanzar los objetivos establecidos y desarrollar este proyecto de forma estructurada, se ha seguido la metodología Cross-Industry Standard Process for Data Mining (CRISP-DM), ampliamente utilizada en proyectos de ciencia e ingeniería de datos. Este enfoque metodológico propone una serie de etapas bien definidas, que han sido adaptadas a las necesidades del trabajo: en primer lugar, se llevó a cabo una comprensión profunda del problema; a continuación, se realizó el proceso de recopilación y preparación de los datos; posteriormente, se construyeron y entrenaron los modelos adecuados; y, finalmente, se evaluaron los resultados utilizando métricas alineadas con los objetivos planteados.

1.1. Motivaciones

Este trabajo parte de la necesidad de mejorar la capacidad de predicción de variables oceanográficas, tanto en términos de exactitud como de anticipación temporal, en regiones de alta relevancia estratégica como el entorno marino de las Islas Canarias. El océano constituye un componente fundamental del sistema climático global y desempeña funciones clave en el transporte de energía, la regulación atmosférica y el sostenimiento de la biodiversidad [Gattuso et al., 2018; Levin et al., 2020; IPCC, 2019]. Asimismo, representa un espacio esencial para el desarrollo de actividades humanas como la pesca, el transporte marítimo o la monitorización del cambio climático [Gattuso et al., 2018]. En este contexto, la predicción precisa de variables como la SST, las corrientes, la salinidad o el nivel del

mar resulta crítica tanto para la investigación científica como para la gestión operativa de los recursos oceánicos [Chassignet et al., 2007; Racault et al., 2012].

Si bien los modelos físicos tradicionales ofrecen un marco sólido para la simulación de estos fenómenos, también presentan limitaciones notables: requieren elevados recursos computacionales, dependen fuertemente de condiciones iniciales precisas y, en muchos casos, operan sobre mallas regulares o predefinidas que no reflejan adecuadamente la complejidad espacial del entorno marino [Blayo and Debreu, 1999; Pain et al., 2005].

En respuesta a estos desafíos, el aprendizaje automático —y en particular el aprendizaje profundo— ha emergido como una alternativa eficaz y escalable para modelar dinámicas oceanográficas complejas a partir de datos observacionales [Zhang et al., 2024a; Bolton and Zanna, 2019; Fablet et al., 2022]. Dentro de este contexto, las redes neuronales de grafos (Graph Neural Networks, GNNs) han demostrado ser especialmente adecuadas para procesar datos definidos sobre dominios no euclidianos, como mallas irregulares o representaciones espaciales heterogéneas del océano [Liu et al., 2023; Wang et al., 2025]. A diferencia de las redes convolucionales tradicionales, las GNNs permiten integrar explícitamente las relaciones espaciales entre puntos de observación mediante la estructura del grafo, ofreciendo una mayor flexibilidad en la modelización de fenómenos localizados o con geometrías complejas [Wang et al., 2025; Liu et al., 2023].

Sin embargo, la eficacia de estos modelos depende en gran medida del diseño de la malla sobre la cual se construye el grafo. En muchas aplicaciones, se recurre a mallas fijas y regulares que no necesariamente se ajustan a la topografía submarina ni a la distribución real de las masas de agua [Pain et al., 2005]. Esta falta de adaptabilidad puede comprometer tanto la precisión de las predicciones como la eficiencia computacional del modelo [Blayo and Debreu, 1999; Pain et al., 2005].

En este trabajo se explora, por tanto, la posibilidad de desarrollar mallas irregulares para la representación de la zona de estudio, partiendo de la premisa de que la forma de los océanos no tiene por qué seguir una estructura geométrica determinada, planteando de esta manera facilidades en el desarrollo de técnicas de densificación de nodos en aquellas zonas cuya no linealidad sea más pronunciada, pero sin entrar en la complejidad computacional que supondría el desarrollo de mallas dinámicamente adaptativas, centrando el trabajo, por tanto, en la creación de mallas estáticas adaptadas al dominio.

Esta motivación se enmarca en la búsqueda de modelos más representativos y eficientes para la predicción de variables oceanográficas [Benedetti et al., 2023], donde, a través de una malla fija más eficiente, se pretende mejorar la capacidad de las GNN para capturar la influencia de la geografía submarina en los procesos oceanográficos, lo que a su vez puede contribuir a una mejora en la precisión y utilidad práctica de las predicciones generadas por estos modelos [Zhang et al., 2024a; Bolton and Zanna, 2019].

1.2. Objetivos

El presente proyecto tiene como finalidad contribuir al avance en la predicción de variables oceanográficas mediante el uso de técnicas de aprendizaje profundo aplicadas a representaciones espaciales más realistas del entorno marino. En particular, se plantea

el desarrollo y evaluación de un modelo basado en GNN, partiendo de la arquitectura propuesta por GraphCast, con el objetivo de explorar nuevas estrategias de modelización más eficientes y precisas. Los objetivos específicos del trabajo son los siguientes:

- Mejorar la predicción de variables oceanográficas mediante la implementación de un modelo basado en GNNs, derivado de la arquitectura GraphCast.
- Analizar el impacto que tiene el uso de distintos tipos de mallas sobre el rendimiento del modelo, evaluando cómo las variaciones en la estructura espacial subyacente afectan a la capacidad predictiva.
- Estudiar el efecto que tienen las mallas adaptativas —en comparación con mallas regulares— sobre la eficiencia computacional de los modelos basados en GNNs.

1.3. Organización del documento

Con el fin de facilitar la comprensión del desarrollo y los objetivos de este Trabajo de Fin de Grado, a continuación se presenta una descripción de la estructura del documento. Cada capítulo ha sido organizado de manera que responda de forma progresiva a las preguntas planteadas en la introducción, abordando desde los antecedentes teóricos hasta la implementación práctica y el análisis de resultados.

El capítulo 2 presenta un estudio del estado del arte en relación con la predicción de variables oceanográficas, poniendo especial énfasis en los distintos enfoques y modelos utilizados para la estimación de la SST, donde se expondrá de esta manera la evolución de los diferentes enfoques que ha tomado la predicción de dicha variable a lo largo de los años.

En el capítulo 3 se expone la planificación inicial del proyecto, así como los ajustes realizados durante su desarrollo. Estos cambios, aunque necesarios por razones prácticas y técnicas, no alteraron significativamente los objetivos fundamentales establecidos en un inicio. Por otra parte, también se detalla la metodología empleada para la realización de este trabajo, así como una relación de las tareas realizadas con el marco metodológico establecido y los recursos utilizados para la culminación de este trabajo.

El capítulo 4 describe los conjuntos de datos empleados, así como la fuente de los mismos, el proceso de descarga y preparación de los mismos, detallando la estructura de las muestras que serán presentadas al modelo, así como de las muestras para pruebas. Por otra parte, también se incluye el proceso de construcción de diversas máscaras auxiliares utilizadas para representar características espaciales del entorno marino y facilitar tanto la creación de las diferentes propuestas de mallas generadas para este trabajo como para el entrenamiento del modelo.

En el capítulo 5 se introducen los fundamentos teóricos de las redes neuronales de grafos, junto con una descripción detallada de los modelos específicos en los que se basa esta propuesta, donde se incluyen GraphCast, el modelo Hi-LAM de Oskarsson y SeaCast, donde se incluye no solo una descripción de su funcionamiento sino las aportaciones de cada uno.

El capítulo 6 describe los distintos modelos de mallas generados para representar el dominio oceánico, así como el proceso seguido para su construcción. Se incluyen tanto mallas con geometrías modificadas como mallas adaptadas específicamente a la zona de estudio mediante técnicas de densificación espacial. Además, se detallan las funciones de densidad utilizadas para guiar dicha densificación, con el objetivo de analizar el impacto que estas representaciones espaciales tienen en el rendimiento del modelo.

El capítulo 7 presenta la configuración de los experimentos realizados, así como los resultados obtenidos en cada uno de ellos. Se detalla cómo los resultados intermedios fueron guiando el diseño de experimentos posteriores, siguiendo un enfoque iterativo propio de la metodología CRISP-DM. Asimismo, se incluye la definición de los escenarios de prueba, los parámetros de entrenamiento empleados y el análisis de los resultados para cada configuración de malla evaluada.

El capítulo 8 recoge las conclusiones derivadas del análisis de los resultados obtenidos, junto con una reflexión crítica sobre las limitaciones del trabajo y una propuesta de posibles líneas de mejora y desarrollo futuro.

El anexo A enumera las competencias específicas del Grado en Ciencia e Ingeniería de Datos (GCID) que han sido desarrolladas durante el transcurso del proyecto, en consonancia con los objetivos formativos del título.

Por último, el anexo B identifica los Objetivos de Desarrollo Sostenible (ODS) relacionados con este trabajo, incluyendo una justificación de cómo se han abordado y aplicado a lo largo del proyecto.

Capítulo 2

Estado del arte

La predicción de variables oceanográficas representa uno de los desafíos más complejos en las ciencias marinas contemporáneas, requiriendo la integración de múltiples disciplinas que abarcan desde la física de fluidos y la dinámica atmosférica hasta las técnicas computacionales más avanzadas y los métodos de inteligencia artificial. El océano, como sistema dinámico, exhibe comportamientos que van desde escalas temporales de segundos hasta décadas, y desde escalas espaciales de metros hasta cuencas oceánicas completas, lo que demanda enfoques metodológicos diversos y complementarios para su comprensión y predicción [Robinson and Lermusiaux, 2002].

Tradicionalmente, la oceanografía predictiva se ha fundamentado en el modelado físico del océano, aproximación que emplea las ecuaciones primitivas de la mecánica de fluidos para describir la dinámica oceánica mediante modelos de circulación general que han demostrado su eficacia durante décadas en la predicción de variables como temperatura, salinidad y corrientes marinas. Sin embargo, el desarrollo exponencial de las técnicas de inteligencia artificial y la disponibilidad masiva de datos oceanográficos han propiciado la emergencia del modelado oceánico con aprendizaje profundo como una aproximación complementaria y, en algunos casos, alternativa a los métodos tradicionales [Reichstein et al., 2019].

Esta nueva metodología promete superar algunas limitaciones inherentes a los modelos físicos, como la parametrización de procesos subescala y los elevados costos computacionales, mediante algoritmos capaces de aprender representaciones complejas directamente de los datos observacionales. La convergencia entre estas dos aproximaciones está configurando un nuevo paradigma en la oceanografía predictiva que combina el rigor físico de los modelos tradicionales con la capacidad de reconocimiento de patrones no lineales de las redes neuronales profundas.

2.1. Modelado físico del océano

El modelado físico constituye la base tradicional y fundamental para la predicción de variables oceanográficas, al apoyarse en la formulación explícita de las leyes que gobiernan la dinámica del océano. Este enfoque se basa en la resolución numérica de las ecuaciones

primitivas que describen el comportamiento de los fluidos geofísicos en un entorno marino. En particular, los denominados Modelos de Circulación General Oceánica (OGCM, por sus siglas en inglés) resuelven las ecuaciones de Navier–Stokes bajo la aproximación hidrostática, junto con la ecuación de continuidad y las ecuaciones de conservación de calor y salinidad [Marshall et al., 1997b].

Entre los modelos más ampliamente utilizados en la comunidad científica internacional destaca el Nucleus for European Modelling of the Ocean (NEMO), un marco de modelado de vanguardia desarrollado por el consorcio europeo del mismo nombre. Este sistema integra múltiples componentes especializados, como NEMO-OPA para la dinámica oceánica, NEMO-LIM para el modelado del hielo marino y NEMO-TOP para procesos biogeoquímicos, todo ello recogido en su documentación técnica oficial [Madec et al., 2016].

Existen no obstante otros modelos igualmente relevantes que han sido adoptados en una amplia gama de aplicaciones oceanográficas. *Hybrid Coordinate Ocean Model* (HYCOM) combina coordenadas de profundidad, isopícnicas y sigma para representar adecuadamente las transiciones verticales, siendo especialmente eficaz en simulaciones globales y regionales de alta resolución [Bleck, 2002]. *Massachusetts Institute of Technology General Circulation Model* (MITgcm) permite la simulación de flujos hidrostáticos y no hidrostáticos, lo que lo hace adecuado para estudios desde escalas costeras hasta globales [Marshall et al., 1997a]. *Regional Ocean Modeling System* (ROMS) es ampliamente utilizado en regiones costeras por su capacidad de representar con precisión procesos dinámicos locales [Shchepetkin and McWilliams, 2005], mientras que *Coastal and Regional Ocean Community model* (CROCO), una evolución de ROMS, ha sido diseñado para mejorar la simulación de flujos de marea y procesos costeros en arquitecturas de alto rendimiento [Debreu et al., 2007; Team, 2023].

Estos modelos físicos requieren condiciones iniciales precisas y forzamientos externos, principalmente viento e intercambio de calor en superficie, para calcular la evolución temporal de las variables oceanográficas mediante esquemas numéricos que discretizan las ecuaciones diferenciales en grillas espaciales tridimensionales [Griffies et al., 2010].

La capacidad predictiva de estos modelos depende críticamente de la parametrización adecuada de procesos subescala, como la turbulencia y la mezcla vertical, así como de la resolución espacial y temporal empleada en la discretización numérica.

2.2. Modelado oceánico con aprendizaje profundo

En las últimas décadas, el uso de técnicas de aprendizaje profundo ha experimentado un crecimiento notable en el ámbito de las ciencias del mar, permitiendo avances significativos en la modelización de fenómenos oceánicos complejos. La capacidad de estas técnicas para aprender representaciones no lineales a partir de grandes volúmenes de datos ha impulsado su aplicación en tareas como la predicción de la temperatura superficial del mar (SST), la reconstrucción de campos incompletos, la identificación de estructuras dinámicas o la detección de eventos extremos, entre otras.

La literatura reciente ha recopilado distintas estrategias arquitectónicas para afrontar estos desafíos, destacando sus ventajas y limitaciones [Zhao et al., 2024].

2.2.1. Redes neuronales convolucionales

Las redes neuronales convolucionales (CNN) han demostrado una notable eficacia en el procesamiento de datos con estructura espacial [LeCun et al., 1989], gracias a su capacidad para aprender representaciones jerárquicas mediante el uso de filtros locales y la compartición de pesos.

En el ámbito oceanográfico, las redes neuronales convolucionales (CNN) han sido ampliamente utilizadas para aprovechar la naturaleza espacial de los datos marinos, especialmente aquellos obtenidos mediante sensores satelitales. Dado que estas redes fueron originalmente desarrolladas para tareas de visión por computador, su aplicación a mapas oceánicos —como imágenes multiespectrales o registros de temperatura superficial del mar (SST)— resulta particularmente adecuada. Las capas de convolución permiten extraer automáticamente patrones espacio-temporales relevantes, aprovechando la fuerte correlación espacial y temporal inherente al entorno marino.

Por ejemplo, en tareas de predicción de SST, es común la aplicación de convoluciones 2D o 3D sobre secuencias de mapas temporales, lo que facilita la detección de estructuras complejas como frentes térmicos y remolinos. Se ha observado que incluso modelos convolucionales relativamente simples pueden anticipar con precisión la evolución de la SST con varios meses de antelación en ciertas regiones. Por su parte, arquitecturas más profundas y sofisticadas incorporan técnicas de *pooling* y procesamiento multiescala, lo que permite capturar tanto variaciones locales como patrones de gran escala.

En el ámbito de las aproximaciones derivadas de redes convolucionales (CNN), se han propuesto múltiples modelos recientes que aplican estas técnicas al tratamiento de datos oceanográficos [Zhao et al., 2024].

Una Temporal Convolutional Network (TCN) ha sido utilizada para extraer características temporales en series de temperatura oceánica, mediante el apilamiento de capas convolucionales que capturan patrones a distintas escalas temporales [Sun et al., 2022]. El modelo alcanza un RMSE de $0,5^{\circ}\text{C}$ sobre un conjunto de datos reales.

Se ha propuesto una arquitectura de doble vía, denominada Dual Path CNN (DP-CNN), que integra información de temperatura superficial del mar (SST), salinidad superficial (SSS) y altura del nivel del mar (SSH), con el objetivo de paliar la escasez de datos subsuperficiales [Mao et al., 2023]. En el Mar de China Meridional, este enfoque alcanza un RMSE de $0,31^{\circ}\text{C}$.

Otro modelo combina la SST y el contenido de calor oceánico (heat content) como variables de entrada, lo que permite predecir la evolución de la SST con hasta seis meses de antelación [Feng et al., 2022], ampliando así el horizonte de predicción.

Una arquitectura multiescala bayesiana, conocida como Multi-Scale Bayesian CNN (MSB-CNN), ha sido desarrollada para capturar estructuras internas de la temperatura oceánica, superando a métodos convencionales gracias a su capacidad de modelar la incertidumbre y adaptarse a diferentes escalas espaciales [Mu et al., 2020].

También se ha diseñado una red híbrida, DGCnetwork, que combina CNN y GRU para modelar relaciones espaciales y dependencias temporales profundas [Yu et al., 2020]. Este modelo obtiene un RMSE de $0,67^{\circ}\text{C}$ en el Mar de China Oriental y $0,56^{\circ}\text{C}$ en el Mar

Amarillo.

Una arquitectura tridimensional basada en 3D U-Net ha sido entrenada con datos multi-fuente para predecir la SST en el Mar de China Meridional, logrando identificar patrones como olas de calor marinas y alcanzar un RMSE de $0,69^{\circ}\text{C}$ [Xie et al., 2024].

Asimismo, se ha desarrollado una Spatiotemporal Siamese CNN (SSCNN) que incorpora información posicional —latitud y longitud— en el modelado de variaciones regionales de la SST en el Océano Índico [Zhang et al., 2024b]. El modelo alcanza un RMSE de $0,47^{\circ}\text{C}$, mejorando su capacidad de generalización espacial.

El modelo Stereo Spatial and Temporal 4-D CNN (SST-4D-CNN) implementa convoluciones en cuatro dimensiones para representar la temperatura oceánica en sus componentes horizontales y verticales, hasta los 2000 m de profundidad, alcanzando una precisión superior al 98 % [Zuo et al., 2022].

Por último, se ha aplicado una red generativa adversarial de superresolución, Enhanced SRGAN (ESRGAN), con el objetivo de mejorar la resolución espacial de los mapas satelitales de temperatura oceánica [Izumi et al., 2022]. Esta estrategia permite representar con mayor detalle la distribución superficial de la temperatura y las corrientes marinas.

2.2.2. Redes recurrentes

Muchas variables oceánicas presentan una dinámica temporal con estacionalidad asociada al ciclo anual. En este contexto, las redes neuronales recurrentes (RNN) han ganado popularidad por su capacidad para modelar secuencias temporales complejas [Hopfield, 1982; Rumelhart et al., 1986]. Sus variantes más destacadas, Long Short-Term Memory (LSTM) y Gated Recurrent Units (GRU), se han utilizado en tareas de predicción tanto univariadas como multivariadas, capturando tendencias temporales en variables oceanográficas [Hochreiter and Schmidhuber, 1997; Cho et al., 2014].

Las redes LSTM se caracterizan por su capacidad para retener dependencias de largo alcance mediante un sistema de compuertas que regula el flujo de información. Esta propiedad resulta especialmente útil para representar fenómenos como los ciclos estacionales, ya que permite aprender patrones anuales o multianuales sin sufrir el desvanecimiento del gradiente en secuencias extensas.

Las GRU, por su parte, presentan una arquitectura más compacta y computacionalmente eficiente, lo que las hace especialmente adecuadas para grandes volúmenes de datos oceánicos. Estas estructuras han demostrado ser altamente eficaces en tareas de predicción espacio-temporal.

También es común combinar redes neuronales recurrentes con redes convolucionales, con el fin de aprovechar la capacidad de las CNN para extraer patrones espaciales y la de las RNN para modelar la evolución temporal. Un ejemplo representativo se encuentra en [Yang et al., 2018], donde una CNN procesa mapas de temperatura superficial del mar (SST) y una LSTM modela su evolución temporal, mejorando el rendimiento respecto a modelos puramente convolucionales. Otro enfoque relevante son las redes ConvLSTM [Shi et al., 2015], que integran operaciones de convolución en las transiciones entre estados de la

LSTM, resultando eficaces para tareas espacio-temporales como la predicción de SST o la concentración de clorofila. En [Zhang et al., 2020] se propone una arquitectura multicapa de este tipo para extraer cambios de temperatura desde la superficie hasta los 2000 metros de profundidad.

En conjunto, las redes neuronales recurrentes (RNN) y sus variantes han demostrado ser adecuadas para el pronóstico de variables oceánicas con dinámica temporal compleja. Su combinación con redes convolucionales (CNN) ha dado lugar a soluciones robustas capaces de explotar simultáneamente las dimensiones espacial y temporal. Entre los principales modelos híbridos recopilados en [Zhao et al., 2024], se encuentran diversas propuestas que integran mecanismos de memoria, atención y extracción multiescala de características aplicadas a la predicción de la SST en distintos entornos oceánicos.

Un ejemplo representativo es el modelo CFCC-LSTM, que integra capas convolucionales con LSTM totalmente conectadas (FC-LSTM) para capturar simultáneamente dependencias espaciales y temporales en datos satelitales del Mar de Bohai, alcanzando un RMSE de $1,07^{\circ}\text{C}$ [Yang et al., 2018]. Con un enfoque diferente, el modelo MC-LSTM descompone las secuencias de SST en componentes semánticos —como tendencia y ciclos—, facilitando la captura de variaciones específicas en el Mar Amarillo, con un RMSE de $0,36^{\circ}\text{C}$ [Lin and Zhong, 2021].

Otros trabajos han incorporado mejoras arquitectónicas para reforzar la eficiencia del modelado espaciotemporal. El modelo D2CL (Dense Dilated ConvLSTM) introduce conexiones densas y dilataciones dentro de una red ConvLSTM, integrando además información hidrológica y ambiental [Hou et al., 2021]. Este enfoque logra un RMSE de $0,66^{\circ}\text{C}$ en el Mar de China Oriental. De forma complementaria, el modelo EAM (Effective Attention Model) incorpora mecanismos de atención sobre las características extraídas por una ConvLSTM, reduciendo redundancias y alcanzando un rendimiento similar en términos de error [Pan et al., 2024].

La predicción multihorizonte también ha sido abordada mediante arquitecturas jerárquicas. El modelo MSH-LSTM (Multi-Head Hierarchical LSTM) implementa múltiples cabezas de atención y capas LSTM organizadas jerárquicamente, con el objetivo de mitigar la acumulación de error en predicciones a múltiples pasos [Liu et al., 2019]. Esta arquitectura alcanza un RMSE de $1,28^{\circ}\text{C}$.

Finalmente, el modelo CGMP combina redes convolucionales GRU (Conv-GRU) con un perceptrón multicapa (MLP). Esta arquitectura modular permite aprovechar las correlaciones espaciales entre celdas adyacentes y se ha aplicado con éxito a la predicción de SST en el Mar de Bohai, destacando por su capacidad de aprendizaje multiescala [Xie et al., 2021].

2.2.3. Modelos basados en redes neuronales de grafos

Las redes neuronales de grafos (GNN) resultan especialmente adecuadas en oceanografía para modelar relaciones espaciales no euclidianas, como las conexiones entre distintas localizaciones geográficas [Scarselli et al., 2009]. En este enfoque, cada nodo representa una posición concreta (por ejemplo, una celda georreferenciada o una boya), y las aristas reflejan conexiones físicas, oceanográficas o estadísticas entre nodos vecinos. Tanto los

atributos nodales (correspondientes a variables oceanográficas a predecir) como los de las aristas (distancia, correlación, etc.) pueden integrarse para capturar dependencias complejas en el espacio-tiempo.

Dada la alta versatilidad que ofrecen este tipo de modelos, existen diferentes estrategias para definir la estructura del grafo subyacente con el objetivo de mejorar la representatividad de los datos. Las dos categorías más comunes son:

Grafos estáticos: En esta modalidad, la topología del grafo permanece constante durante todo el proceso de predicción, capturando dependencias espaciales fijas entre ubicaciones oceánicas. Esta estructura ha demostrado ser eficaz para modelar correlaciones espaciales en áreas marinas con condiciones estables [Samsi et al., 2017]. Las GNN aplicadas a la predicción de SST con topologías estáticas han superado en precisión a los enfoques puramente convolucionales, al representar de manera efectiva la estructura geográfica del dominio y las relaciones espaciales entre posiciones oceánicas.

Grafos dinámicos: Cuando la topología del grafo varía con el tiempo, el modelo puede representar dinámicamente las relaciones en el entorno marino, reflejando así los cambios en las condiciones oceánicas. Estudios recientes han mostrado que la actualización de nodos y aristas en cada paso temporal permite capturar mejor fenómenos estacionales o eventos transitorios [Goyal et al., 2020; Manessi et al., 2020].

Modelos relevantes en oceanografía basados en GNN: Sobre la base de estas estrategias, se han desarrollado diversas arquitecturas específicas para tareas de predicción en oceanografía, particularmente centradas en la temperatura superficial del mar (SST). Uno de los modelos más destacados es GNN-OAM, que integra múltiples matrices de adyacencia y mecanismos de atención optimizados para capturar relaciones espacio-temporales en tres dimensiones, alcanzando un RMSE de $0,41^{\circ}\text{C}$ [Ou et al., 2024]. Otro ejemplo relevante es GMNN, que emplea un codificador gráfico para representar explícitamente la estructura espacial y una unidad de memoria tipo LSTM para modelar dependencias temporales prolongadas, logrando un RMSE de $0,23^{\circ}\text{C}$ en el Pacífico Norte [Liang et al., 2023].

Una aproximación distinta es la propuesta de DYGOV, una red dinámica capaz de ajustar la estructura del grafo a lo largo del tiempo, integrando múltiples variables oceánicas y meteorológicas para reflejar interacciones físicas cambiantes. Este modelo alcanza un RMSE de $0,96^{\circ}\text{C}$ [Wang et al., 2024a]. Por su parte, la arquitectura HiGRN aplica una estructura jerárquica basada en agrupación de nodos mediante *clustering* y atención multinivel, combinando información local y global para lograr un RMSE de $0,40^{\circ}\text{C}$ en la predicción global de la SST [Yang et al., 2023].

Importancia e impacto. Las GNN ofrecen una capacidad destacada para integrar relaciones espaciales irregulares entre ubicaciones oceánicas, lo que resulta especialmente útil en dominios con geometrías complejas. Su compatibilidad con grafos dinámicos permite reflejar cambios temporales en las interacciones marinas, mientras que la incorporación de mecanismos como la atención [Vaswani et al., 2017] y las estructuras jerárquicas posibilita

capturar información en distintos niveles de resolución. Esta flexibilidad ha demostrado mejorar significativamente la predicción de SST, en especial en regiones con gran variabilidad dinámica como zonas costeras, archipiélagos o áreas influenciadas por corrientes oceánicas variables [Cuervo-Londoño et al., 2025].

Los resultados reportados por los modelos analizados muestran errores medios cuadráticos (RMSE) entre 0,23 y 0,96 °C, lo que evidencia la capacidad de estas redes para representar con fidelidad el comportamiento espacio-temporal de variables oceanográficas. Su eficacia se debe tanto a la integración de atributos nodales y de aristas relevantes como a su adaptabilidad para incorporar mecanismos avanzados como atención multiescala, memoria recurrente o topologías dinámicas, facilitando una representación más rica y realista del entorno marino.

2.2.4. Aprendizaje profundo guiado por la física

El aprendizaje profundo guiado por la física ha emergido como una estrategia clave para mejorar la precisión y la interpretabilidad de los modelos de predicción de temperatura oceánica. Esta aproximación integra conocimiento físico en redes neuronales, ya sea mediante el uso de variables físicas como entradas, o incorporando directamente leyes físicas como restricciones durante el entrenamiento. Este enfoque puede llevarse a cabo de diferentes maneras, tal y como se expone a continuación.

Modelos con entradas de variables físicas: Diversos estudios han demostrado que la incorporación de parámetros físicos relevantes —como la velocidad del viento, la velocidad de las corrientes superficiales, la presión atmosférica, la humedad relativa y los flujos de calor— mejora significativamente la capacidad predictiva de los modelos de temperatura superficial del mar (SST). Estos parámetros representan procesos dinámicos y termodinámicos que influyen directamente sobre la evolución térmica del océano, por lo que su inclusión permite una representación más realista de las condiciones oceánicas.

Uno de los primeros enfoques en integrar variables físicas mediante GNN consiste en modelar la interacción entre la SST y la velocidad del viento a través de estructuras no euclidianas. En este contexto, se propone una red que representa relaciones espaciales complejas y refleja con mayor precisión las dinámicas regionales [Kim et al., 2023]. De forma similar, el estudio de [Chen et al., 2023] plantea un modelo multivariable que integra parámetros dinámicos y termodinámicos para predecir cambios de SST a largo plazo, alcanzando horizontes de hasta seis meses y subrayando la importancia de considerar procesos físicos interrelacionados.

Más recientemente, [Miao et al., 2023] presenta un modelo basado en redes convolucionales (CNN) que incorpora como entradas la velocidad del viento y de las corrientes superficiales. Esta combinación permite capturar relaciones físicas entre variables forzantes y la SST, guiando la predicción con mayor precisión. En la misma línea, [Zhang et al., 2024b] propone un enfoque que combina datos multihidrológicos con información sobre la velocidad del viento mediante una CNN, lo que permite extraer características prolongadas a partir de secuencias temporales largas, mejorando la detección de patrones persistentes en el comportamiento térmico del océano.

Redes neuronales informadas por física Las redes neuronales informadas por física (Physics-Informed Neural Networks, PINN) representan un enfoque avanzado que incorpora directamente el conocimiento físico en el diseño y entrenamiento de los modelos. Este paradigma busca mejorar la capacidad predictiva e interpretabilidad de las redes, integrando leyes físicas relevantes como restricciones durante la optimización. Esta integración permite reducir la dependencia exclusiva de los datos y mitigar comportamientos no realistas, lo cual resulta especialmente valioso en contextos como la oceanografía, donde las relaciones físicas son conocidas pero difíciles de capturar mediante redes puramente empíricas [Krishnapriyan et al., 2021; Cuomo et al., 2022].

Existen principalmente dos estrategias comunes para incorporar este conocimiento: por un lado, el diseño de arquitecturas explícitamente guiadas por leyes físicas (como ecuaciones de conservación o balances energéticos); por otro, la inclusión de términos de regularización en la función de pérdida, que penalizan desviaciones respecto a principios fundamentales.

En el contexto de la predicción de temperatura oceánica, varios estudios recientes han aplicado este enfoque con resultados prometedores [Zhao et al., 2024]. Por ejemplo, en [Yuan et al., 2023], se incorpora la ecuación del balance de calor en la capa mixta del océano para guiar la estimación de temperatura en profundidad, permitiendo una mejor captura de procesos termodinámicos verticales. El estudio de [Rice et al., 2020] combina redes convolucionales con la teoría de Koopman para modelar de forma más interpretable las dinámicas no lineales subyacentes en la evolución térmica marina. En [Wu et al., 2023], se introduce el fenómeno de estratificación térmica para modelar explícitamente la relación entre la temperatura superficial y las capas profundas del océano, reforzando la coherencia vertical del modelo. Finalmente, [Wang et al., 2024b] desarrolla un módulo de simulación de convección oceánica que se integra en el modelo de predicción de SST, logrando mejoras de entre un 1 % y un 5 % en precisión, y aumentando la interpretabilidad del sistema.

En conjunto, estos enfoques muestran que la incorporación explícita de principios físicos no solo mejora la exactitud de las predicciones, sino que también garantiza su coherencia con el conocimiento científico existente. Este aspecto resulta especialmente relevante en aplicaciones operacionales y de modelado climático, donde la validez física de las soluciones es tan importante como su ajuste a los datos.

Capítulo 3

Metodología y planificación del trabajo

3.1. Metodología

El desarrollo de este proyecto ha seguido la metodología Cross Industry Standard Process for Data Mining (CRISP-DM), un enfoque ampliamente reconocido en el ámbito de la ciencia e ingeniería de datos, así como en proyectos basados en aprendizaje profundo. Esta metodología se ha consolidado como un estándar de facto gracias a su carácter iterativo, flexible y adaptable a distintos dominios, y es empleada tanto en entornos académicos como industriales [Shearer, 2000]. En el contexto de este trabajo, CRISP-DM ha proporcionado un marco estructurado para abordar de forma sistemática las distintas fases del proceso, desde la definición del problema hasta la evaluación experimental del modelo propuesto, cuyo ciclo de vida se representa en la Figura 3.1.

El proceso se inició con la fase de comprensión del negocio, en la que se analizó el problema desde una perspectiva aplicada, definiendo como objetivo principal la adaptación del modelo SeaCast [Holmberg et al., 2024] a una región geográfica concreta: la costa africana próxima al archipiélago canario. Esta zona presenta un interés particular debido a la frecuente aparición de fenómenos de afloramiento, que provocan el ascenso de aguas profundas más frías y ricas en nutrientes hacia la superficie. Estos procesos tienen una repercusión significativa en el ámbito pesquero, al favorecer una mayor concentración de biomasa durante los periodos de afloramiento. Por tanto, mejorar la precisión de las predicciones en esta región se plantea como un objetivo de elevado valor tanto socioeconómico como medioambiental.

A continuación, se abordó la fase de comprensión de los datos disponibles en la plataforma Copernicus, un sistema abierto y público de distribución de datos planetarios. Se realizó una exploración preliminar centrada en identificar la estructura, calidad y características de las variables relevantes ofrecidas por este servicio en relación con la problemática planteada. Por otra parte, se verificó que el modelo original podía ser ejecutado localmente siguiendo las instrucciones del autor. Considerando la disponibilidad de los datos y las limitaciones computacionales, se seleccionaron como variables principales la temperatura



Figura 3.1: Flujo de trabajo de la metodología CRISP-DM. Fuente: IBM Corporation [2024]

superficial del mar y los componentes zonal y meridional del viento. Posteriormente, se delimitó el dominio espacial mediante coordenadas geográficas específicas y se evaluaron distintas máscaras oceánicas con el fin de representar adecuadamente la superficie marina durante el entrenamiento. Durante las pruebas del modelo, se analizaron los formatos de entrada requeridos, así como el contenido esperado en los archivos, lo que permitió adecuar los datos a las exigencias del sistema base.

La fase de preparación de los datos incluyó las tareas necesarias para construir un conjunto de datos adecuado, tales como la limpieza, transformación y adaptación de variables. Aunque muchos de los procesos de preprocesamiento ya estaban integrados en el repositorio original, fue imprescindible modificarlos para ajustarlos a los nuevos datos y al dominio geográfico seleccionado. Donde fue necesario interpolar los datos batimétricos al nuevo dominio y a adaptar su resolución a la de los datos de Copernicus. Además, se aplicaron factores de ponderación latitudinal para mitigar las distorsiones espaciales propias de las rejillas rectangulares utilizadas.

Durante la fase de modelado, se empleó el modelo SeaCast, una implementación basada en la arquitectura GraphCast, con el objetivo de predecir variables oceánicas. Se mantuvieron constantes el número de capas y los hiperparámetros para poder aislar el efecto producido exclusivamente por las modificaciones en la estructura de los grafos, es decir, por el uso de diferentes mallas espaciales. Esta decisión permitió realizar una comparación controlada entre distintas configuraciones de malla y su impacto sobre el rendimiento del modelo.

La evaluación del modelo se llevó a cabo utilizando la métrica Root Mean Square Error

(RMSE), ampliamente empleada en el ámbito de la predicción meteorológica y oceanográfica por su capacidad para cuantificar errores considerando la importancia relativa de cada celda geográfica [Ben Bouallègue et al., 2024]. Esta evaluación permitió analizar el impacto de las distintas configuraciones de malla sobre la calidad de las predicciones, verificando así el grado de cumplimiento de los objetivos establecidos en la fase inicial del trabajo.

Por último, aunque la metodología CRISP-DM contempla una fase de despliegue orientada a la implementación del modelo en entornos reales, este proyecto no aborda dicha etapa. En su lugar, se ha optado por centrar el trabajo en un análisis exploratorio y experimental, cuyo propósito principal ha sido generar conocimiento aplicable a futuras investigaciones y proponer mejoras metodológicas en el uso de grafos para predicción oceánica.

3.2. Planificación

3.2.1. Planificación inicial

En lo que respecta a la planificación inicial de este Trabajo de Fin de Grado, se elaboró un esquema temporal teniendo en cuenta tanto los recursos computacionales disponibles como los objetivos específicos planteados para el proyecto. Esta planificación, recogida de forma detallada en la Tabla 3.1, se articula en cinco fases principales que abarcan desde el estudio preliminar hasta la presentación final del trabajo.

La primera fase estuvo dedicada al estudio y revisión bibliográfica, centrada principalmente en la arquitectura del modelo GraphCast. La comprensión profunda de este modelo resultó esencial para orientar correctamente el desarrollo del proyecto, ya que permitió identificar sus elementos clave, establecer comparaciones con trabajos previos y formular hipótesis fundamentadas. Esta etapa también sirvió para diseñar una hoja de ruta clara que guiara el resto de las actividades.

Seguidamente, se abordó la selección y preparación de los datos, donde se definieron las fuentes a utilizar y se delimitó la región geográfica objeto de estudio. Esta fase incluyó un análisis exploratorio inicial que permitió evaluar la calidad, resolución y cobertura de los datos disponibles. En los casos en que se detectaron inconsistencias o carencias, se aplicaron técnicas de limpieza, transformación y normalización con el objetivo de garantizar su adecuación para ser procesados por el modelo.

El núcleo técnico del proyecto se desarrolló durante la fase de diseño, desarrollo e implementación, centrada en adaptar el modelo GraphCast al nuevo contexto. Para ello, se planteó el uso de técnicas de ajuste fino (*fine-tuning*) como estrategia principal, permitiendo reutilizar parte del conocimiento aprendido por el modelo original y ajustarlo de forma específica a las características del nuevo dominio.

A continuación, se procedió al entrenamiento, evaluación y optimización del modelo. Esta etapa requirió una gestión eficiente de los recursos de hardware disponibles, especialmente unidades de procesamiento gráfico (GPU), para reducir los tiempos de cómputo. Tras el entrenamiento, se evaluó el rendimiento del modelo mediante métricas reconocidas en el ámbito de la predicción oceánica, lo que permitió valorar su eficacia y, en caso necesario,

realizar ajustes en su configuración para mejorar los resultados.

Finalmente, se llevó a cabo la fase de documentación y presentación, en la cual se sistematizó todo el trabajo realizado. Esta fase incluyó la redacción de la memoria del TFG, con una descripción detallada de la metodología empleada, los resultados obtenidos y su análisis crítico. Asimismo, se prepararon los materiales complementarios necesarios para la defensa oral del proyecto, asegurando una presentación clara, estructurada y respaldada por evidencias.

3.2.2. Tareas realizadas

Durante el desarrollo del proyecto fue necesario reformular algunas de las tareas inicialmente planteadas, adaptándolas a las condiciones reales del trabajo y a los recursos disponibles. No obstante, la mayoría de ellas se mantuvieron conforme a la planificación inicial.

En primer lugar, se decidió comenzar el trabajo a partir del modelo SeaCast [Holmberg et al., 2024], basado en una metodología similar a la utilizada en GraphCast [Lam et al., 2022] y en la propuesta de Oskarsson [Oskarsson et al., 2023]. Esta elección permitió establecer un punto de partida sólido y operativo, sin desviar los objetivos principales del proyecto.

Por otra parte, se constató que no era viable aplicar técnicas de ajuste fino (*fine-tuning*) al modificar la malla de la red neuronal, lo que llevó a descartar dicha estrategia. En su lugar, se optó por entrenar los modelos desde cero, priorizando configuraciones que permitiesen obtener resultados en tiempos de ejecución razonables.

En cuanto al diseño experimental, y considerando las limitaciones computacionales existentes, el enfoque se centró no solo en mejorar la capacidad predictiva sobre las variables oceanográficas, sino también en optimizar el rendimiento del modelo base mediante una reducción de los tiempos de entrenamiento asociados a las distintas configuraciones de malla propuestas.

3.3. Recursos empleados

Durante el desarrollo de este Trabajo de Fin de Grado se emplearon diversos recursos, tanto a nivel de hardware como de software, con el objetivo de garantizar una ejecución eficiente de los experimentos y una correcta elaboración del documento. A continuación, se describen los principales componentes utilizados en cada uno de estos ámbitos.

En lo relativo al hardware, las pruebas y entrenamientos del modelo se llevaron a cabo en uno de los equipos disponibles en el laboratorio 2 del Centro de Tecnologías de la Imagen de la Universidad de Las Palmas de Gran Canaria. El equipo contaba con un procesador Intel Core i9-9900K a 3.6 GHz, 32 GB de memoria RAM y una tarjeta gráfica NVIDIA RTX 3060 con 12 GB de VRAM, lo que permitió ejecutar modelos de aprendizaje profundo de forma local y eficiente, aprovechando la capacidad de cómputo paralelo proporcionada por la GPU.

En cuanto al entorno software, se utilizó el sistema operativo Windows 10 Enterprise. El desarrollo del código se realizó íntegramente en Python (versión 3.10.16), utilizando la distribución Anaconda para la gestión del entorno, la instalación de dependencias y el control de versiones de librerías. Para la experimentación interactiva se emplearon entornos Jupyter Notebooks, mientras que el editor Visual Studio Code se utilizó como entorno principal de desarrollo.

La redacción de la memoria se llevó a cabo en la plataforma colaborativa Overleaf, basada en LaTeX, lo que facilitó la escritura estructurada del documento y la gestión automatizada de referencias bibliográficas. Adicionalmente, se utilizaron Microsoft PowerPoint e Inkscape para la creación y edición de figuras y gráficos incluidos en el trabajo.

En lo que respecta a librerías y herramientas de programación, se hizo uso intensivo de NumPy y Xarray para la manipulación de datos numéricos y multidimensionales, fundamentales en el tratamiento de variables meteorológicas y oceanográficas. La implementación del modelo se llevó a cabo con el framework PyTorch, complementado con PyTorch Geometric para la construcción y entrenamiento de redes neuronales sobre grafos. Para la visualización de resultados se utilizó Matplotlib, mientras que NetworkX fue empleada en la generación y análisis de las estructuras de malla que definen el dominio espacial del modelo. Asimismo, SciPy se incorporó en el desarrollo de algoritmos asociados a la creación de dichas mallas.

El entrenamiento en GPU se realizó gracias al uso del CUDA Toolkit, herramienta indispensable para aprovechar la aceleración por hardware en operaciones matriciales intensivas. Por último, el control de versiones del código fuente se gestionó mediante el sistema Git, utilizando la plataforma GitHub para el almacenamiento remoto, la colaboración y la trazabilidad del desarrollo.

Tabla 3.1: Fases y tareas desarrolladas en el proyecto

Fases	Duración Estimada (horas)	Tareas
Estudio/Revisión Bibliográfica	70 horas	Tarea 1.1: Revisión de la literatura sobre el modelo de GraphCast, analizando su arquitectura, características, y resultados, así como su aplicabilidad a la predicción de variables oceanográficas. Tarea 1.2: Estudio de las técnicas de “Transfer-Learning” y “Fine-Tuning”, enfocándose en los métodos propuestos por el artículo y su adaptación al contexto oceanográfico así como analizar implementaciones de ajustes de mallas hechas en el mediterráneo.
Selección y Preparación de los Datos	30 horas	Tarea 2.1: Selección y recopilación de datos oceanográficos desde “Copernicus Marine Environment Monitoring Service”. Tarea 2.2: Preprocesamiento de datos, incluyendo limpieza, interpolación y normalización.
Diseño/ Desarrollo/ Implementación	220 horas	Tarea 3.1: Uso del modelo preentrenado para realizar inferencias iniciales con nuestros datos y evaluación preliminar del rendimiento antes del ajuste completo. Tarea 3.2: Adaptación del modelo GraphCast mediante la implementación de técnicas de ajuste fino, ajuste de pesos y parámetros.
Entrenamiento, Evaluación y Optimización del Modelo	200 horas	Tarea 4.1: Entrenamiento del modelo GraphCast en local, utilizando GPUs con CUDA para acelerar el proceso de ajuste del modelo, realizando múltiples pruebas con distintos pesos y enfoques de ajuste fino. Tarea 4.2: Validación del rendimiento del modelo adaptado utilizando técnicas de validación cruzada y métricas de evaluación. Tarea 4.3: Comparación del rendimiento adaptado con diferentes enfoques de fine-tuning, utilizando diferentes parámetros y pesos.
Documentación/ Presentación	80 horas	Tarea 5.1: Redacción de la memoria, incluyendo la descripción detallada de la metodología, resultados y conclusiones. Tarea 5.2: Preparación de la defensa del TFG, resaltando los principales hallazgos y contribuciones del proyecto.

Capítulo 4

Conjuntos de datos

El desarrollo de un modelo predictivo eficaz en el ámbito oceanográfico requiere partir de una base de datos sólida y representativa, capaz de reflejar la complejidad y variabilidad del entorno marino. En este contexto, la selección y caracterización del conjunto de datos adquieren un papel fundamental, ya que condicionan en gran medida la capacidad del modelo para generalizar y capturar los patrones dinámicos presentes en el medio. Este capítulo se centra en describir los datos empleados a lo largo del trabajo, detallando su procedencia, las variables consideradas y los criterios seguidos para su preprocesamiento y utilización en los experimentos.

4.1. Fuentes de datos

Las fuentes de datos empleadas en este estudio provienen, en su mayoría, del Programa de Observación de la Tierra de la Unión Europea, Copernicus. Este programa se organiza en distintos servicios especializados, cada uno orientado al monitoreo de un componente específico del sistema terrestre. En el presente trabajo se han utilizado principalmente el Copernicus Marine Service y el Copernicus Climate Change Service, seleccionados por su relevancia en la modelización de procesos físicos y dinámicos del entorno oceánico.

4.1.1. Copernicus Marine Service

El Copernicus Marine Service proporciona datos relativos al medio marino estructurados en tres dominios fundamentales: el estado físico del océano (ocean blue), la cobertura de hielo marino (ocean white) y las variables biogeoquímicas (ocean green). Estos datos, disponibles a escala global y regional, se generan mediante una combinación de observaciones satelitales, mediciones in situ y modelos numéricos avanzados.

Para este trabajo se ha seleccionado la variable de temperatura superficial del mar (SST) proporcionada por este servicio [European Union-Copernicus Marine Service, 2019]. Esta elección se fundamenta en el alto nivel de procesamiento del producto, así como en su elevada resolución espacial ($0.05^\circ \times 0.05^\circ$) y temporal, características esenciales para un análisis riguroso del comportamiento térmico del océano. La SST representa un factor

clave en la identificación de fenómenos de afloramiento, de gran importancia para el estudio oceanográfico y la caracterización de los ecosistemas marinos.

4.1.2. Copernicus Climate Change Service

El Copernicus Climate Change Service tiene como finalidad principal proporcionar datos, indicadores y proyecciones que respalden las políticas de adaptación y mitigación frente al cambio climático en Europa. Su catálogo incluye productos derivados de modelos climáticos globales y observaciones sistemáticas de variables atmosféricas y superficiales.

En este estudio se han empleado datos del viento a 10 metros de altura, en particular los componentes zonal (este-oeste) y meridional (norte-sur) [Hersbach et al., 2023]. El componente zonal representa la velocidad del aire en dirección este, mientras que el componente meridional refleja la velocidad hacia el norte, ambos expresados en metros por segundo. A partir de estos datos, es posible reconstruir el vector de viento horizontal, permitiendo analizar tanto su intensidad como su dirección, aspectos fundamentales para comprender la dinámica del transporte oceánico, donde estos componentes del viento serán de utilidad para mejorar la predicción de la SST.

La inclusión de estas variables responde a su utilidad en la detección del fenómeno de afloramiento, particularmente relevante en la región de la costa africana. Este proceso, caracterizado por el ascenso de aguas profundas frías y ricas en nutrientes hacia la superficie, está estrechamente asociado a un aumento en la productividad marina y la concentración de bancos de peces en dichas zonas. Por tanto, su estudio resulta clave tanto desde el punto de vista ecológico como para el seguimiento y gestión de los recursos pesqueros.

4.2. Proceso de descarga de los datos

La obtención de los datos se llevó a cabo mediante las interfaces de programación de aplicaciones (API) oficiales proporcionadas por ambos servicios. En el caso del Copernicus Climate Change Service, los archivos fueron descargados en formato NetCDF a través del acceso programático a la base de datos ERA5. Por su parte, el Copernicus Marine Service permitió la descarga directa de datos estructurados como objetos `xarray`, lo que facilitó su integración en el flujo de procesamiento.

Con el fin de optimizar tanto el almacenamiento como el tiempo de cómputo, se procedió a descargar únicamente las variables necesarias para el desarrollo del modelo. Esta selección dirigida permitió reducir significativamente la carga de datos sin comprometer la calidad ni la representatividad de la información.

El periodo temporal considerado abarca desde el 1 de enero de 2018 hasta el 31 de diciembre de 2023. La elección de este intervalo se justifica por tratarse del rango más reciente disponible en los servicios de Copernicus, compatible con las restricciones computacionales del entorno de desarrollo. Para cada día dentro de ese intervalo, se descargaron los valores correspondientes a todas las variables seleccionadas, garantizando una resolución temporal diaria adecuada para el análisis de fenómenos oceanográficos.

4.3. Procesado de los datos

El procesado de los datos descargados se estructura en tres etapas fundamentales: la generación de variables estáticas, la construcción del conjunto de datos diario y la definición del conjunto de entrenamiento en formato PyTorch. Todo el flujo ha sido implementado en Python mediante bibliotecas especializadas como `xarray`, `numpy`, `cdsapi`, `copernicusmarine` y `torch`.

Primeramente, se construyó el conjunto de datos diario. Para cada día comprendido entre 2018 y 2023, se descargaron las variables escogidas para este Trabajo de Fin de Grado, donde cada archivo diario se organizó como una matriz tridimensional de forma (n_{grid}, f) , donde n_{grid} representa el número de celdas oceánicas y f el número de variables. Antes de su almacenamiento, los datos fueron interpolados espacialmente a la rejilla definida por una máscara tierra-agua, se descartaron las celdas terrestres, se reemplazaron los valores ausentes por ceros y se homogenizaron las dimensiones para asegurar compatibilidad con las estructuras esperadas por los modelos de aprendizaje automático.

Para preparar correctamente las muestras de entrenamiento y evaluación, se diseñó un conjunto de scripts encargados de concatenar secuencias temporales de estados diarios, generar datos de forzamiento atmosférico y combinar análisis y predicciones con máscaras de contorno. En primer lugar, las secuencias oceánicas fueron organizadas mediante la concatenación de un número fijo de días consecutivos, conformando así una muestra coherente en el tiempo. En paralelo, se procesaron los datos de forzamiento atmosférico mediante el mismo proceso que los datos oceánicos de tal manera que haya una correspondencia temporal entre ellos, donde el número de días consecutivos se estableció en seis en ambos casos, donde tendríamos dos días de inicialización para darle contexto al modelo y cuatro días de evolución de las variables en el tiempo. A cada muestra se le asignó una marca temporal a partir del nombre del archivo, y todos los datos fueron almacenados en formato `.npy` para su posterior uso por el cargador de datos. Estos procedimientos garantizan la coherencia temporal y espacial de las muestras, así como la correcta sincronización entre los estados del océano y las variables atmosféricas que actúan como forzamiento externo del sistema.

En el caso del conjunto de test, se aplicó un procedimiento ligeramente distinto con el objetivo de evaluar la capacidad del modelo para realizar predicciones a largo plazo. A diferencia de las muestras de entrenamiento y validación, que contienen secuencias de seis días, las muestras de test comprenden un total de diecisiete días consecutivos. Este incremento permite simular un escenario más exigente de predicción continuada sin realimentación. La construcción de estas secuencias se realiza mediante la concatenación de los dos primeros días —utilizados como contexto para inicializar el modelo— con quince días adicionales cuya evolución debe ser predicha. Para las variables de forzamiento, se incluyeron datos atmosféricos desde dos días antes del inicio hasta el final del periodo de predicción, asegurando así la disponibilidad de la información necesaria en cada paso de la secuencia. Esta configuración permite evaluar no solo la precisión del modelo en horizontes temporales prolongados, sino también su estabilidad y coherencia dinámica a lo largo de una predicción extendida.

Tras la preparación de las muestras de datos en conjuntos de días, se da paso a la definición del conjunto de entrenamiento. Para ello, se empleó la clase denominada `WeatherDataset`,

basada en la interfaz `torch.utils.data.Dataset`. Esta clase permite construir de forma dinámica las muestras de entrada y salida a partir de los archivos diarios, gestionando operaciones como la selección temporal de ventanas, el submuestreo a diferentes escalas, la estandarización de las variables a partir de estadísticas precomputadas, la generación de variables de forzamiento —que incluyen tanto datos atmosféricos como codificaciones temporales basadas en funciones seno y coseno—, y la combinación de dichas variables en ventanas deslizantes. Las codificaciones temporales se obtienen mediante la transformación de la fecha de cada muestra en una representación angular normalizada del año, lo que permite capturar la estacionalidad implícita en los fenómenos oceánicos. Al aplicar funciones seno y coseno sobre este ángulo, se generan dos variables periódicas que reflejan de manera continua y diferenciable la posición relativa dentro del ciclo anual, evitando discontinuidades asociadas a los cambios de año. Cada muestra está compuesta por dos días iniciales utilizados como contexto, seguidos de una secuencia de predicción, acompañados por las variables de forzamiento correspondientes a cada paso temporal.

De manera complementaria, se llevó a cabo el cálculo de estadísticas globales sobre el conjunto de entrenamiento. Este procedimiento incluyó la estimación de la media y desviación típica de cada variable oceanográfica y de los forzamientos atmosféricos, así como de las diferencias entre pasos temporales consecutivos. Estas estadísticas fueron calculadas únicamente sobre las celdas oceánicas definidas por la máscara tierra-agua y almacenadas en archivos binarios para su uso posterior durante el entrenamiento. La estandarización resultante permite estabilizar el proceso de optimización del modelo y facilita la interpretación cuantitativa de los errores de predicción.

4.4. Representación de la zona de estudio

La zona de estudio considerada en este trabajo se encuentra delimitada por las coordenadas indicadas en la Tabla 4.1, las cuales se sitúan dentro de la región oceánica de Iberia–Vizcaya–Irlanda (IBI), representada en la Figura 4.1. En dicha figura se ha resaltado gráficamente el área específica seleccionada para el desarrollo del presente trabajo. Esta región abarca parte de la costa africana, incluyendo enclaves destacados como los cabos Blanco, Jubi y Ghir, así como los archipiélagos de Canarias y Madeira, tal como se muestra en la Figura 4.2. Por último, en la Figura 4.3 se representan los valores correspondientes a la temperatura superficial del mar (SST) para el primer día del conjunto de datos utilizado.

Coordenada	Mínima (°)	Máxima (°)
Latitud	19.550	34.525
Longitud	-20.970	-5.975

Tabla 4.1: Rango de coordenadas geográficas

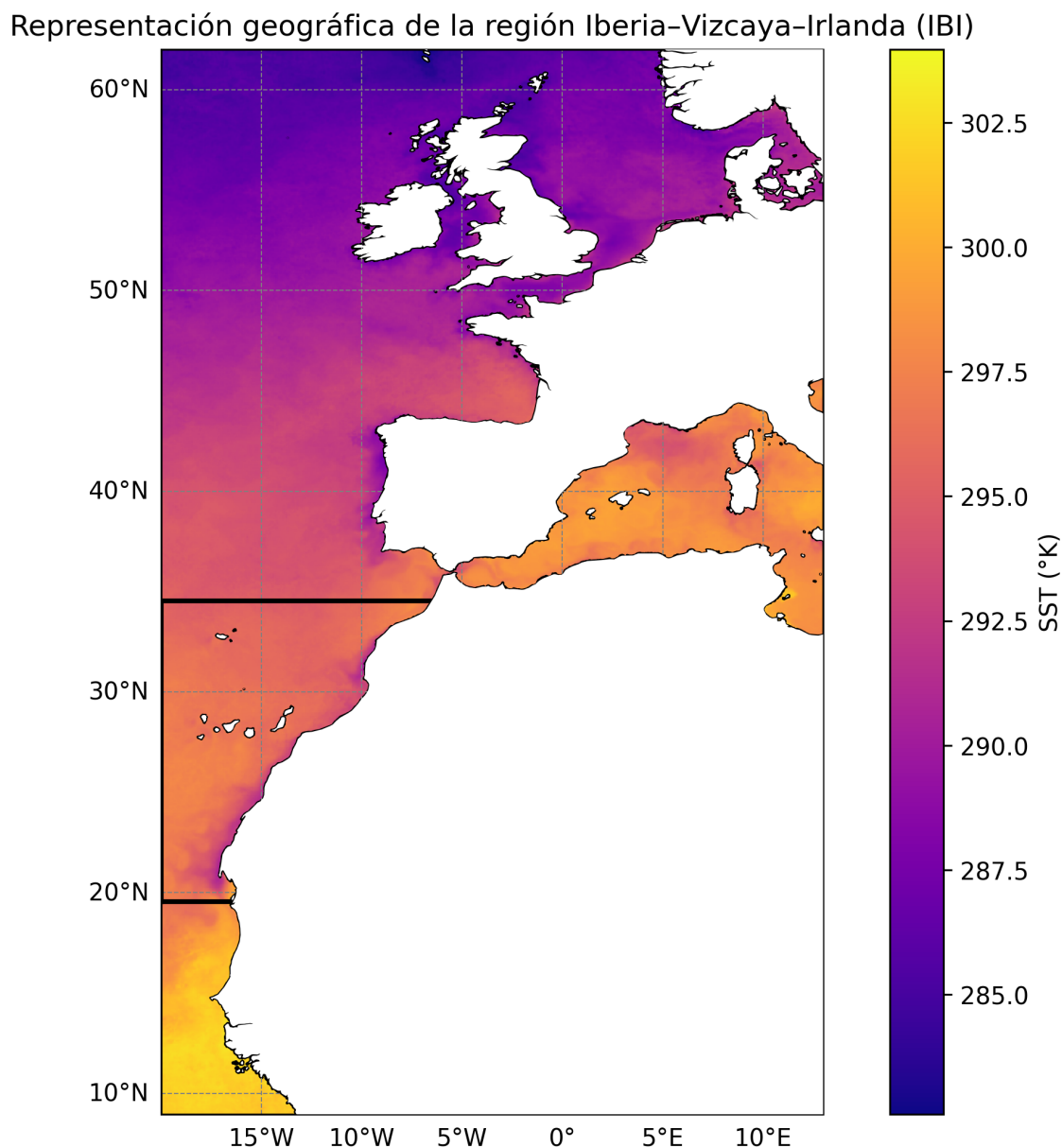


Figura 4.1: Representación de la zona de estudio dentro de la zona IBI

4.5. Creación de las máscaras de la zona de estudio

La generación de máscaras espaciales constituye una etapa fundamental en la preparación de datos para modelos de predicción oceanográfica, ya que permite incorporar conocimiento geográfico, físico y geométrico del dominio de estudio. En el desarrollo original del modelo SeaCast, aplicado al mar Mediterráneo, se integraron múltiples máscaras con distintos propósitos: desde la delimitación de zonas válidas para el modelado hasta el ajuste de pesos en las métricas o la incorporación de condiciones de contorno. En el presente trabajo, se ha llevado a cabo un análisis detallado del impacto de estas máscaras en la calidad de las predicciones, así como una adaptación y recreación de las mismas en función de los datos disponibles para la región del océano Atlántico seleccionada.

La primera máscara implementada fue la máscara tierra-agua. Esta permite diferenciar

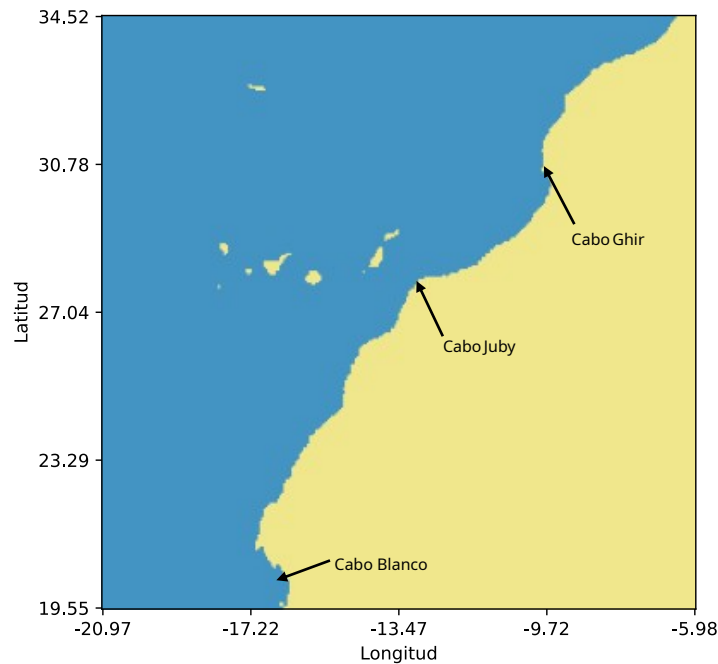


Figura 4.2: Representación de la zona de estudio

las celdas oceánicas de las continentales, siendo imprescindible para descartar regiones no marinas tanto en el entrenamiento como en la inferencia del modelo. Se construyó a partir de un día representativo del conjunto de datos de Copernicus Marine, aprovechando que los valores correspondientes a zonas terrestres, como el continente africano, las Islas Canarias o Madeira, ya venían definidos como cero. A partir de estos valores se generó una máscara binaria que distingue de forma clara el dominio oceánico, como puede apreciarse en la Figura 4.4. Adicionalmente, existe una versión tridimensional de esta máscara que extiende dicha diferenciación a todos los niveles verticales del océano. Sin embargo, dado que este trabajo se enfoca exclusivamente en la superficie oceánica, ambas versiones son funcionalmente equivalentes.

En segundo lugar, se diseñó una máscara de batimetría para caracterizar la profundidad oceánica en la zona de estudio. Esta se elaboró a partir de los datos del modelo ETOPO [NOAA National Centers for Environmental Information, 2022], ajustando las coordenadas a los límites definidos en la Tabla 4.1. El objetivo de incorporar la batimetría es proporcionar al modelo un contexto físico adicional que podría mejorar la precisión de las predicciones, especialmente en zonas costeras o con fuerte gradiente batimétrico. No obstante, fue necesario adaptar la resolución espacial de los datos batimétricos a la de los datos oceánicos mediante técnicas de *downscaling*. Asimismo, se procedió a transformar el formato de los datos para alinearlos con el utilizado en SeaCast: mientras que ETOPO emplea una codificación donde los valores positivos representan tierra y los negativos profundidad, SeaCast codifica la profundidad como valores positivos y asigna ceros a las zonas terrestres. Para lograr esta homogeneización, se utilizó la máscara tierra-agua para identificar las zonas oceánicas, se invirtió el signo de los valores batimétricos y se truncaron a cero aquellos que permanecieran negativos. El resultado final se muestra en la

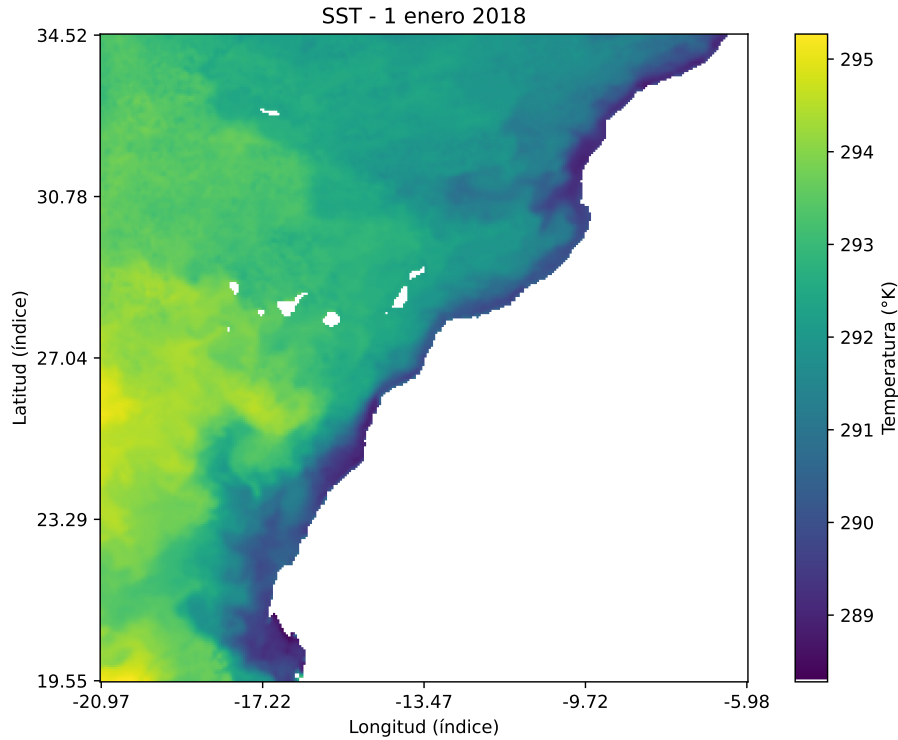


Figura 4.3: Temperatura superficial del mar (K)

Figura 4.5.

Otro elemento clave fue la construcción de una máscara de pesos para la grilla. Dado que la superficie terrestre no es una esfera perfecta sino un geoide, la densidad de puntos sobre una grilla regular en latitud-longitud no representa de forma equitativa las distintas áreas geográficas. Este sesgo se ve reflejado en los datos satelitales, donde las observaciones son más densas en las regiones ecuatoriales que en las polares. Para corregir este efecto y ponderar adecuadamente las métricas durante el entrenamiento y evaluación del modelo, se adoptó la fórmula propuesta por Rasp et al. [2020] en el proyecto WeatherBench, la cual ajusta los pesos de cada latitud en función del coseno de su valor angular:

$$L(j) = \frac{\cos(\text{lat}(j))}{\frac{1}{N_{\text{lat}}} \sum_j^{N_{\text{lat}}} \cos(\text{lat}(j))}$$

Esta normalización garantiza que la suma total de los pesos corresponda al número total de píxeles oceánicos, permitiendo una evaluación justa del rendimiento del modelo a lo largo del dominio. La distribución resultante de estos pesos se ilustra en la Figura 4.6.

Adicionalmente, se evaluó la incorporación de una máscara basada en la topografía dinámica media (MDT, por sus siglas en inglés), que representa la altura media del nivel del mar respecto al geoide. Esta máscara fue utilizada en la versión original de SeaCast como una forma de codificar la circulación media del océano. Sin embargo, dado que los datos MDT no se encuentran disponibles en el producto utilizado para esta implementación y considerando tanto su complejidad de obtención como su impacto marginal en los resultados finales, se optó por prescindir de esta componente.

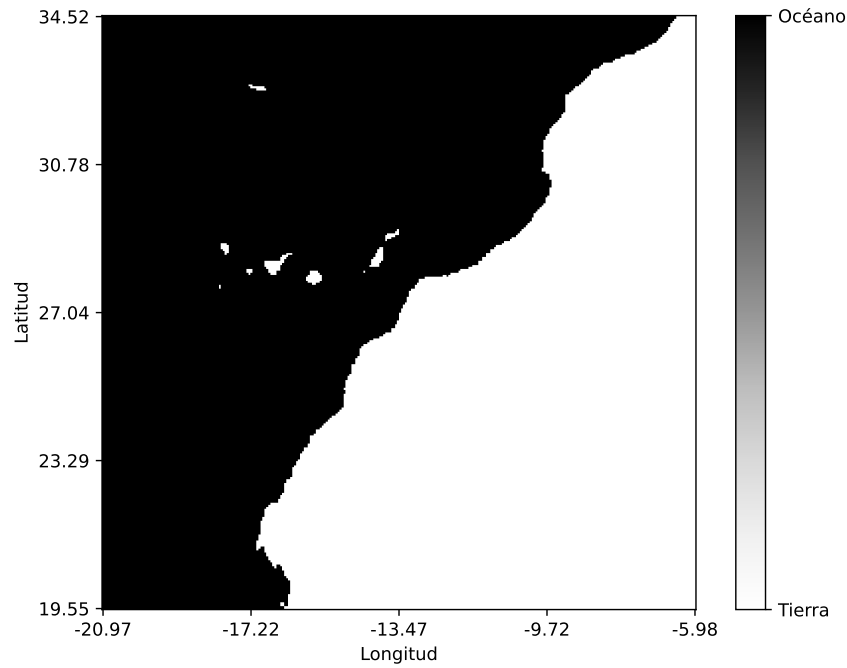


Figura 4.4: Máscara tierra-agua

Finalmente, se descartó también la máscara de forzamiento de contorno utilizada en el modelo original para el Mediterráneo, donde se sustituían las predicciones en la región del estrecho de Gibraltar por valores reales del conjunto de test. Dado que en el caso del Atlántico la dinámica es más abierta y la no linealidad de la temperatura superficial del mar es menos crítica en los bordes del dominio considerado, se decidió no aplicar dicho forzamiento, fijando esta máscara completamente a cero.

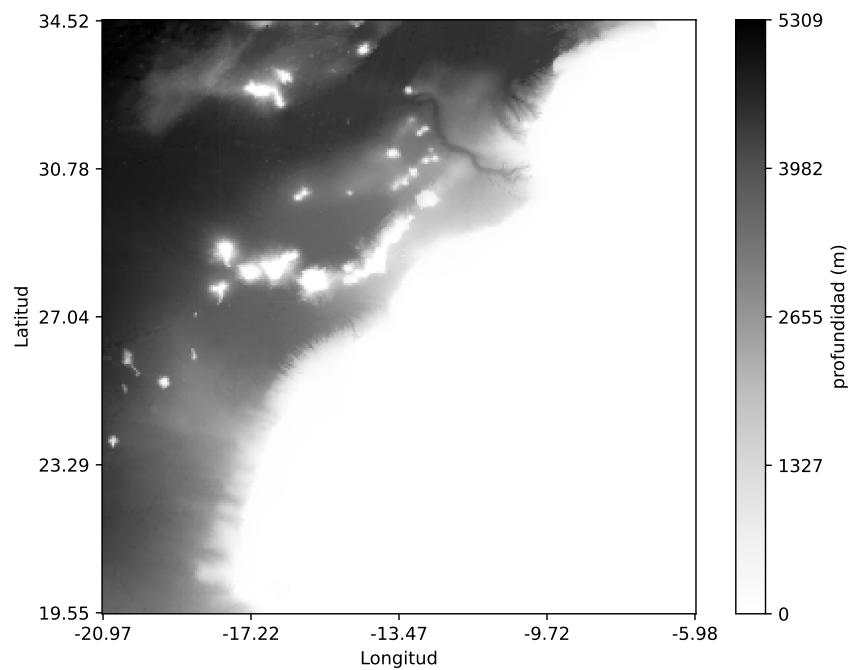


Figura 4.5: Máscara de batimetría

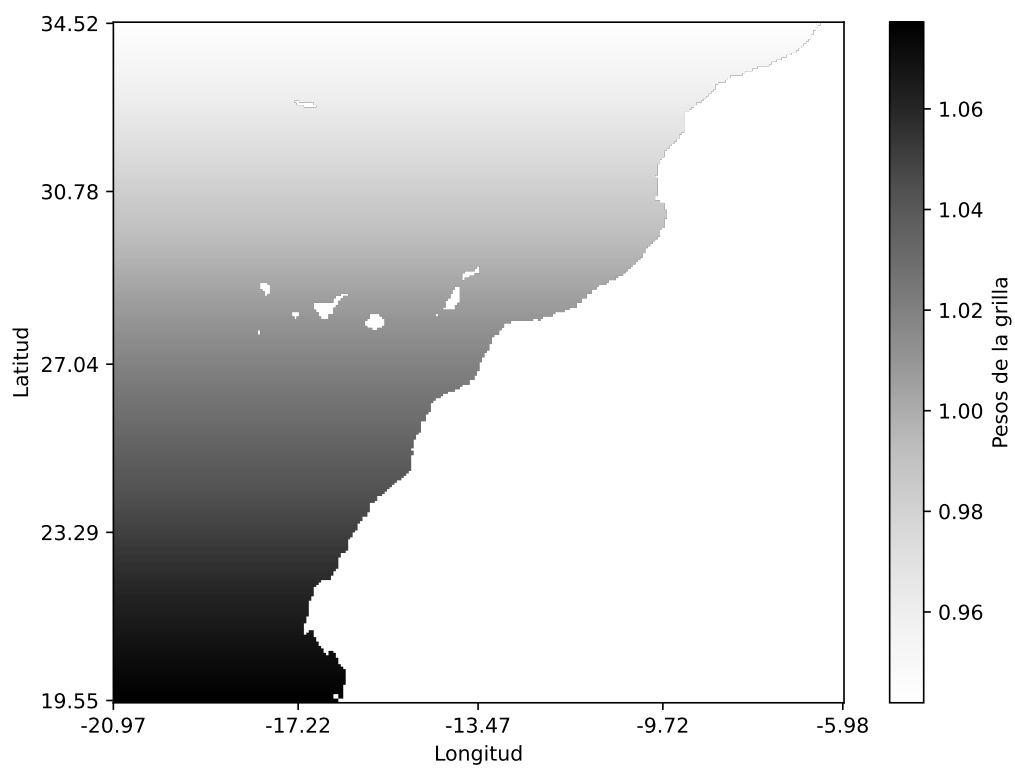


Figura 4.6: Pesos de la grilla

Capítulo 5

Redes neuronales empleadas

Tal como se expone en la sección de estado del arte, las investigaciones más recientes en predicción de variables meteorológicas y oceanográficas destacan la importancia de incorporar variables físicas interrelacionadas, especialmente aquellas que inciden directamente en la temperatura superficial del mar. Factores como el viento o la presión atmosférica condicionan de forma significativa esta variable, lo que justifica el uso de modelos capaces de capturar dependencias espaciales y temporales complejas.

En esta línea, destacan modelos avanzados de predicción meteorológica como Aurora, desarrollado por Microsoft [Bodnar et al., 2024], y GraphCast, de Google [Lam et al., 2022]. Este último ha sido adoptado como base metodológica, al emplear GNNs para representar de manera eficiente las relaciones espaciales entre celdas geográficas. Además, tal como se demuestra en trabajos como [Cuervo-Londoño et al., 2025], esta arquitectura ha mostrado una alta capacidad de adaptación al ámbito oceanográfico, incluyendo tareas de predicción de temperatura superficial del mar.

La elección de GraphCast se sustenta en su arquitectura flexible y su capacidad para modelar dinámicas físicas complejas en múltiples escalas. Estas características han dado lugar a desarrollos posteriores como el modelo Hi-LAM [Oskarsson et al., 2023], orientado a regiones geográficas limitadas, y SeaCast [Holmberg et al., 2024], adaptado al estudio de variables oceánicas en el mar Mediterráneo.

Antes de presentar los detalles específicos del enfoque propuesto en este trabajo, se introducen a continuación los fundamentos teóricos de las redes neuronales de grafos, así como una justificación de su idoneidad frente a otras alternativas metodológicas para la tarea abordada.

5.1. Las redes neuronales de grafos

En esta sección se presenta la estructura general y el funcionamiento de las redes neuronales de grafos, tomando como punto de partida el trabajo seminal de Scarselli et al. [Scarselli et al., 2009]. En dicho trabajo se consolidó por primera vez el término red neuronal de grafos (GNN), a partir de desarrollos previos centrados en el procesamiento de datos estructurados en forma de grafos, donde los elementos (nodos) y sus relaciones

(aristas) representan información compleja y no euclidiana.

Scarselli propuso un modelo de red neuronal capaz de operar directamente sobre grafos dirigidos, no dirigidos, cíclicos y acíclicos, extendiendo así las capacidades de las redes neuronales tradicionales y estableciendo las bases teóricas de este tipo de arquitecturas.

El modelo planteado se fundamenta en la idea de que cada nodo puede actualizar iterativamente su estado interno (*embedding*) en función de la información procedente de sus vecinos y de las características de las aristas que los conectan. Este proceso se repite hasta alcanzar una representación estable para cada nodo en el grafo.

Formalmente, un grafo se representa como:

$$\mathcal{G} = (\mathcal{N}, \mathcal{E}),$$

donde $\mathcal{N}$ es el conjunto de nodos, cada uno asociado a una etiqueta o vector de características l_n , y $\mathcal{E}$ es el conjunto de aristas, donde cada arista (n, u) conecta dos nodos y puede estar etiquetada con $l_{(n,u)}$.

Las ecuaciones fundamentales que definen el comportamiento de una GNN son la función de transición local y la función de salida local, expresadas como:

$$x_n^{(t+1)} = f_w \left(l_n, l_{\text{co}[n]}, x_{\text{ne}[n]}^{(t)}, l_{\text{ne}[n]} \right), \quad (5.1)$$

$$o_n^{(t)} = g_w \left(x_n^{(t)}, l_n \right), \quad (5.2)$$

donde $x_n^{(t)}$ representa el estado interno del nodo n en la iteración t , y $o_n^{(t)}$ la salida generada por dicho nodo en esa misma iteración. La función f_w , parametrizada por w , actualiza el estado del nodo utilizando su información propia l_n , las características de sus conexiones $l_{\text{co}[n]}$, los estados de sus vecinos $x_{\text{ne}[n]}^{(t)}$, y sus respectivas características $l_{\text{ne}[n]}$. Por su parte, g_w es la función de salida local, también parametrizada por w , y produce la salida final del nodo a partir de su estado interno y sus atributos.

Este modelo ofrece una gran flexibilidad para representar y aprender información en dominios estructurados, permitiendo trabajar con grafos de naturaleza diversa. Además, favorece una integración efectiva de la información tanto a nivel global, considerando la estructura completa del grafo, como a nivel local, mediante las interacciones entre nodos vecinos.

A partir de esta formulación han surgido numerosas variantes y extensiones que han mejorado significativamente la forma de procesar información estructurada en grafos. Estas variantes se suelen agrupar en tres enfoques principales, siendo estos el convolucional, basado en atención y basado en paso de mensajes, tal como se expone en [Bronstein et al., 2021].

Enfoque convolucional. Este enfoque propone actualizar la representación de un nodo u agregando las características de sus vecinos mediante pesos fijos. La actualización se formaliza como:

$$h_u = \varphi \left(x_u, \bigoplus_{v \in \mathcal{N}_u} c_{uv} \cdot \psi(x_v) \right), \quad (5.3)$$

donde h_u es la nueva representación del nodo u , x_u su vector de características, y $\mathcal{N}_u$ su vecindario. La función ψ transforma las características de cada vecino v , c_{uv} es un coeficiente fijo, $\bigoplus$ representa una operación de agregación invariante al orden (por ejemplo, suma o media), y φ es una función de combinación no lineal, como una red neuronal o una activación ReLU.

Enfoque atencional. Introduce un mecanismo para ponderar de forma diferenciada las contribuciones de los vecinos según su relevancia. La representación del nodo u se actualiza como:

$$\mathbf{h}_u = \phi \left(\mathbf{x}_u, \bigoplus_{v \in \mathcal{N}_u} a(\mathbf{x}_u, \mathbf{x}_v) \cdot \psi(\mathbf{x}_v) \right), \quad (5.4)$$

donde $a(\mathbf{x}_u, \mathbf{x}_v)$ es una función de atención que asigna un peso α_{uv} a cada vecino v , normalmente normalizado con **softmax**. El resto de los componentes mantiene la misma interpretación que en el modelo anterior.

Enfoque de paso de mensajes. Generaliza los anteriores permitiendo que cada nodo actualice su representación a partir de vectores arbitrarios, denominados mensajes, recibidos de sus vecinos. La formulación general es:

$$\mathbf{h}_u = \phi \left(\mathbf{x}_u, \bigoplus_{v \in \mathcal{N}_u} \psi(\mathbf{x}_u, \mathbf{x}_v) \right), \quad (5.5)$$

donde ψ genera el mensaje que v envía a u , y ϕ combina los mensajes con la representación de u . Este enfoque incluye como casos particulares a los dos anteriores, utilizando expresiones específicas para ψ y a .

Estas estrategias de agregación no solo representan pilares teóricos del aprendizaje profundo en dominios no euclidianos, sino que también han permitido desarrollar arquitecturas eficaces para problemas complejos del mundo real. En particular, destacan aplicaciones como la predicción de variables ambientales, donde las relaciones espaciales y temporales son claves. Un ejemplo representativo es GraphCast [Lam et al., 2022], una arquitectura basada en el enfoque de paso de mensajes que ha logrado resultados sobresalientes en predicción meteorológica. Esta arquitectura servirá como base para el desarrollo propuesto en este trabajo.

Una vez explicado el funcionamiento general de las redes neuronales de grafos, a continuación se presentan los modelos concretos en los que se fundamenta este proyecto, destacando aquellos aspectos técnicos que resultaron especialmente relevantes en su implementación.

5.2. El modelo GraphCast

GraphCast es un modelo de aprendizaje automático diseñado para la predicción meteorológica global a rango medio. A diferencia de los sistemas de predicción numérica del tiempo (NWP), que resuelven las ecuaciones físicas en servidores, GraphCast se entrena directamente con datos históricos de reanálisis [Hersbach et al., 2020], aprovechando grandes volúmenes de información para optimizar el modelo subyacente, donde la mayoría de los resultados expuestos en la investigación se obtuvieron con un modelo entrenado con 39 años (1979-2017) y evaluado con 3 años (2018-2021).

Este cambio de enfoque ofrece ventajas significativas en eficiencia y rapidez, donde GraphCast genera pronósticos globales con resolución de $0,25^\circ$ (aprox. 28 km) para un horizonte de más de 10 días en menos de un minuto, utilizando un único dispositivo Google Cloud TPU v4. Además, resulta especialmente útil para aplicaciones como la predicción de trayectorias de ciclones tropicales, ríos atmosféricos y temperaturas extremas.

El modelo de Graphcast funciona con datos con un paso temporal de seis horas: para cada predicción toma los dos estados más recientes —el actual y el de seis horas atrás— y estima el estado seis horas en el futuro. Gracias a su carácter autorregresivo, similar al de los sistemas numéricos de predicción meteorológica, puede “desplegarse” de forma iterativa, alimentando sus propias predicciones como nueva entrada para generar trayectorias de estados meteorológicos de longitud arbitraria.

La estructura del modelo destaca por la malla multinivel propuesta, especialmente relevante para nuestro proyecto, donde GraphCast emplea una GNN con arquitectura *encode-process-decode*, que opera directamente sobre esta malla. En la fase de codificación, los datos de entrada—dispuestos en una rejilla uniforme de latitud/longitud a $0,25^\circ$ (aprox. 28×28 km en el ecuador)—se proyectan hacia los nodos de la malla; más adelante profundizaremos en su estructura multinivel durante el procesamiento.

A continuación, se describen las tres etapas principales de la arquitectura de GraphCast:

5.2.1. Codificación

La etapa de codificación constituye el primer componente fundamental de la arquitectura de GraphCast, que adopta una estructura basada en codificador, procesador y decodificador empleando GNNs. Su objetivo principal es mapear la información del estado meteorológico de entrada, representada sobre una rejilla latitud/longitud de resolución 0.25° (con 721×1440 puntos), a una representación interna definida sobre una malla multinivel más homogénea espacialmente.

El uso de una malla interna resulta ventajoso, ya que la rejilla latitud/longitud presenta una distribución no uniforme de puntos, con mayor densidad cerca de los polos, lo que genera una carga computacional desproporcionada. La malla utilizada por GraphCast se construye a partir de un icosaedro refinado iterativamente seis veces ($\mathcal{M}_6$), como se muestra en la Figura 5.1, apartado g). Esta malla contiene 40,962 nodos en su nivel de resolución más alto (ver Tabla 5.1) y permite interacciones tanto locales como de largo alcance a través de aristas de diferentes longitudes.

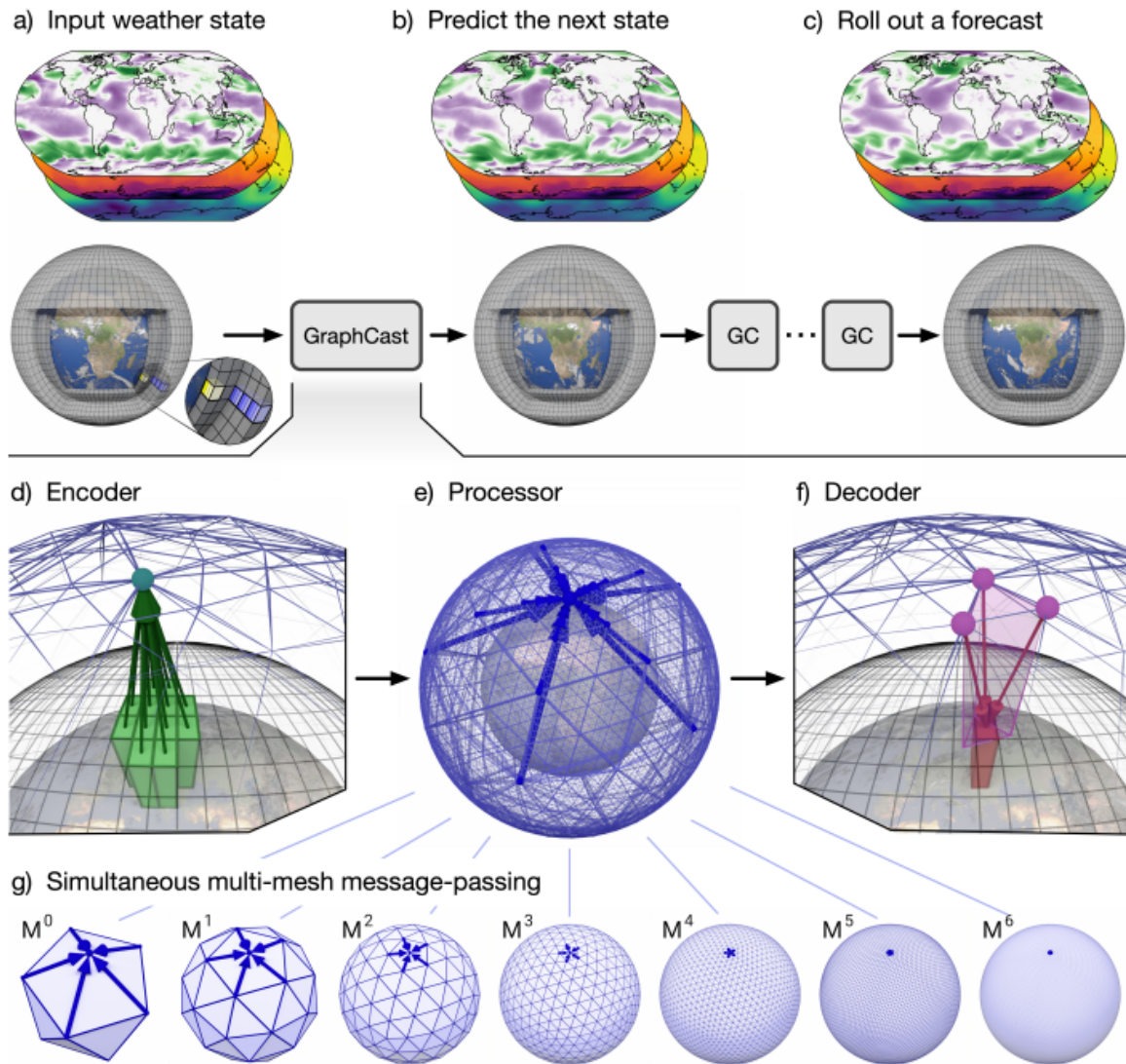


Figura 5.1: Estructura de la malla de GraphCast con detalles sobre como los distintos procesos dentro de la arquitectura *encode-process-decode* se llevan a cabo dentro de la malla. Fuente: [Lam et al., 2022]

El proceso de codificación se compone de dos etapas principales. En primer lugar, se realiza la inclusión de características (*embedding*) mediante perceptrones multicapa (MLP) que proyectan las características de los nodos de la rejilla, los nodos de la malla y las aristas—incluyendo tanto las de la malla como las que conectan la rejilla con la malla— a un espacio latente de dimensión fija.

Las características asociadas a cada nodo de la rejilla v_i^G en el instante t incluyen el estado meteorológico en los tiempos t y $t - 1$, considerando variables de superficie y variables atmosféricas en 37 niveles de presión (lo que da un total de $5 + 6 \times 37 = 227$ variables por punto). Además, se añaden términos de forzado dependientes del tiempo (como radiación solar, hora local del día y progreso del año), así como características estáticas (máscara tierra-mar, geopotencial de superficie, latitud y longitud). Por su parte, los nodos de la malla v_i^M codifican el coseno de la latitud y el seno/coseno de la longitud. Las características de las aristas se derivan de las posiciones tridimensionales de los nodos que conectan.

Las proyecciones latentes de los nodos y aristas se obtienen mediante las siguientes expresiones:

$$v_i^G = \text{MLP}_{\text{embedder VG}}(v_{\text{features},i}^G), \quad (5.6)$$

$$v_i^M = \text{MLP}_{\text{embedder VM}}(v_{\text{features},i}^M), \quad (5.7)$$

$$e_{v_s^M \rightarrow v_r^M}^M = \text{MLP}_{\text{embedder EM}}(e_{\text{features},v_s^M \rightarrow v_r^M}^M), \quad (5.8)$$

$$e_{v_s^G \rightarrow v_r^M}^{G2M} = \text{MLP}_{\text{embedder EG2M}}(e_{\text{features},v_s^G \rightarrow v_r^M}^{G2M}), \quad (5.9)$$

$$e_{v_s^M \rightarrow v_r^G}^{M2G} = \text{MLP}_{\text{embedder EM2G}}(e_{\text{features},v_s^M \rightarrow v_r^G}^{M2G}). \quad (5.10)$$

En estas expresiones, v_i^G y v_i^M representan los vectores latentes de los nodos de la rejilla y de la malla, respectivamente, mientras que e^M , e^{G2M} y e^{M2G} representan las aristas de la malla, de la rejilla a la malla y de la malla a la rejilla. Los subíndices de cada MLP indican redes entrenadas de forma específica para cada tipo de nodo o arista. Los superíndices G , M , $G2M$ y $M2G$ hacen referencia al origen y destino de los elementos. Además, s y r denotan, respectivamente, los nodos remitente y receptor de una arista.

La segunda etapa consiste en aplicar una GNN bipartita unidireccional sobre un subgrafo que conecta los nodos de la rejilla (v_s^G) con los nodos de la malla (v_r^M). Se define una arista $e_{v_s^G \rightarrow v_r^M}^{G2M}$ si la distancia entre los nodos involucrados es menor o igual a 0.6 veces la longitud de la arista más larga de $\mathcal{M}_6$, según se ilustra en la Figura 5.1, apartado d). Esta condición garantiza que cada nodo de la rejilla se conecte al menos con un nodo de la malla.

La actualización en esta GNN sigue un enfoque de paso de mensajes similar a una red de interacción [Battaglia et al., 2016, 2018], adaptado para múltiples tipos de nodos. Primero, cada arista del tipo Grid2Mesh se actualiza concatenando su representación original con las representaciones latentes de sus nodos adyacentes:

$$e_{v_s^G \rightarrow v_r^M}^{G2M'} = \text{MLP}_{\text{Grid2Mesh EG2M}}([e_{v_s^G \rightarrow v_r^M}^{G2M}, v_s^G, v_r^M]). \quad (5.11)$$

Posteriormente, cada nodo de la malla se actualiza agregando todas las aristas entrantes que provienen de la rejilla, mediante una operación de suma:

$$v_i^{M'} = \text{MLP}_{\text{Grid2Mesh VM}} \left(\left[v_i^M, \sum_{e_{v_s^G \rightarrow v_r^M}^{G2M} : v_r^M = v_i^M} e_{v_s^G \rightarrow v_r^M}^{G2M'} \right] \right). \quad (5.12)$$

En este paso, la MLP correspondiente procesa la concatenación del estado actual del nodo y la suma de los mensajes entrantes. Por su parte, los nodos de la rejilla también se actualizan, aunque de manera independiente, ya que en esta fase actúan únicamente como emisores:

$$v_i^{G'} = \text{MLP}_{\text{Grid2Mesh VG}}(v_i^G). \quad (5.13)$$

Finalmente, se aplican conexiones residuales para integrar las actualizaciones con las representaciones originales, lo cual se formaliza de la siguiente manera:

$$v_i^G \leftarrow v_i^G + v_i^{G'}, \quad (5.14)$$

$$v_i^M \leftarrow v_i^M + v_i^{M'}, \quad (5.15)$$

$$e_{v_s^G \rightarrow v_r^M}^{G2M} \leftarrow e_{v_s^G \rightarrow v_r^M}^{G2M} + e_{v_s^G \rightarrow v_r^M}^{G2M'}. \quad (5.16)$$

Con ello, la información meteorológica original queda transformada en una representación latente distribuida sobre los nodos de la malla, que servirá como entrada para la siguiente etapa de procesamiento en la arquitectura de GraphCast.

5.2.2. Procesamiento

. La etapa de procesamiento constituye un componente clave de la arquitectura del modelo GraphCast, y se encarga de operar sobre la representación interna generada previamente por el codificador. A diferencia de la rejilla latitud/longitud de entrada, cuya distribución de puntos no es uniforme, la malla interna utilizada durante esta etapa, siendo esta una multimalla jerárquica, es espacialmente homogénea y proporciona una alta resolución global.

La estructura de esta multimalla, ilustrada en la Figura 5.1, se construye a partir de un icosaedro regular inicial (M_0), compuesto por 12 nodos y 20 caras. A este icosaedro se le aplican seis refinamientos iterativos; en cada uno de ellos, las caras triangulares del nivel anterior se subdividen en cuatro caras más pequeñas, lo que multiplica por cuatro el número de caras y aristas en cada nivel sucesivo.

Una vez completado el refinamiento hasta M_6 , se toman los 40,962 nodos de esta malla de máxima resolución y se superpone sobre ellos la unión de todas las aristas generadas desde M_0 hasta M_6 . Esto es posible porque los nodos de las mallas de menor resolución son subconjuntos de los nodos en M_6 , lo que permite definir una jerarquía plana de aristas

con longitudes variables. Las aristas provenientes de niveles gruesos (M_0 a M_5) conectan nodos lejanos en la esfera, mientras que las aristas de M_6 representan conexiones locales entre nodos cercanos. Esta organización facilita la propagación eficiente de información tanto a corta como a larga distancia mediante pocos pasos de intercambio de mensajes. La Tabla 5.1 resume las características estructurales de cada nivel de la malla.

Refinamiento	0	1	2	3	4	5	6
Número de nodos	12	42	162	642	2 562	10 242	40 962
Número de caras	20	80	320	1 280	5 120	20 480	81 920
Número de aristas	60	240	960	3 840	15 360	61 440	245 760
Número de aristas multinivel	60	300	1 260	5 100	20 460	81 900	327 660

Tabla 5.1: Detalles de refinamiento de la malla multinivel de GraphCast.

Durante la etapa de procesamiento, la interacción entre los nodos de la malla se modela a través de una GNN profunda, compuesta por 16 capas. Cada una de estas capas corresponde a una red de interacción independiente [Battaglia et al., 2016, 2018], entrenada con un conjunto de pesos distintos. Esta independencia permite a la red capturar patrones complejos y específicos a distintas escalas espaciales en cada capa del procesamiento.

En cada una de las 16 capas, se lleva a cabo un paso de mensajes dividido en dos fases: primero se actualizan las aristas y luego los nodos. La actualización de cada arista $e_{v_s^M \rightarrow v_r^M}^M$, que conecta el nodo emisor v_s^M con el nodo receptor v_r^M , se realiza concatenando sus características con las de los nodos que conecta y procesándolas mediante un perceptrón multicapa específico para esa capa:

$$e_{v_s^M \rightarrow v_r^M}^{M'} = \text{MLP}_{\text{Mesh EM}}([e_{v_s^M \rightarrow v_r^M}^M, v_s^M, v_r^M]). \quad (5.17)$$

Una vez actualizadas las aristas, se procede a actualizar los nodos. Para cada nodo v_i^M , se agregan los mensajes provenientes de todas las aristas entrantes $e^{M'}$ y se combinan con su representación actual, utilizando un perceptrón multicapa distinto:

$$v_i^{M'} = \text{MLP}_{\text{Mesh VM}} \left(\left[v_i^M, \sum_{\substack{e_{v_s^M \rightarrow v_r^M}^M : v_r^M = v_i^M}} e_{v_s^M \rightarrow v_r^M}^{M'} \right] \right). \quad (5.18)$$

En esta expresión, la operación de suma actúa como agregador de los mensajes entrantes. La MLP resultante toma como entrada la concatenación de la representación actual del nodo con la información agregada para producir una nueva representación actualizada.

Finalizada la actualización de nodos y aristas, se aplican conexiones residuales para preservar la información original y mejorar la estabilidad del entrenamiento. Esta técnica consiste en sumar las actualizaciones obtenidas a las representaciones previas, de acuerdo con las siguientes asignaciones:

$$v_i^M \leftarrow v_i^M + v_i^{M'}, \quad (5.19)$$

$$e_{v_s^M \rightarrow v_r^M}^M \leftarrow e_{v_s^M \rightarrow v_r^M}^M + e_{v_s^M \rightarrow v_r^M}^{M'}. \quad (5.20)$$

Este ciclo de actualización de aristas y nodos, seguido de conexiones residuales, se repite en cada una de las 16 capas del procesador. A través de esta estructura multinivel y el refinamiento progresivo de la información, GraphCast logra capturar eficazmente las dinámicas meteorológicas a múltiples escalas. Las representaciones latentes resultantes de los nodos de la malla al final de esta etapa se utilizarán como entrada para la fase final de decodificación del modelo.

5.2.3. Decodificación

La fase final de la arquitectura de GraphCast corresponde al proceso de decodificación, cuyo objetivo es mapear las representaciones latentes aprendidas por el procesador de vuelta a la rejilla original en coordenadas latitud/longitud. Este paso permite reconstruir el estado meteorológico predicho sobre el dominio de salida esperable. Además, como se ha descrito anteriormente, GraphCast es un modelo autoregresivo: utiliza sus propias predicciones como entrada para producir pronósticos futuros. En concreto, el decodificador estima una actualización residual sobre el estado de entrada más reciente para obtener el siguiente estado del sistema.

El decodificador está compuesto por una única capa de red neuronal de grafos que opera sobre un subgrafo bipartito denominado Mesh2Grid (M2G), el cual conecta nodos de la malla como emisores con nodos de la rejilla como receptores. Para cada punto de la rejilla, se establecen tres conexiones con los nodos de malla que forman el triángulo de la malla $\mathcal{M}_6$ en el que se encuentra contenido dicho punto. Las aristas Mesh2Grid son unidireccionales y contienen cuatro características, calculadas de forma análoga a las de las aristas internas de la malla. Esta estructura puede observarse en el apartado f) de la Figura 5.1.

El mecanismo de paso de mensajes en el decodificador sigue una lógica similar a la utilizada en la fase de codificación (Grid2Mesh), pero en dirección inversa. Primero, se actualizan las características de cada arista $e_{v_s^M \rightarrow v_r^G}^{M2G}$, combinando su valor actual con las representaciones de los nodos extremos mediante un perceptrón multicapa:

$$e_{v_s^M \rightarrow v_r^G}^{M2G'} = \text{MLP}_{\text{Mesh2Grid EM2G}} \left(\left[e_{v_s^M \rightarrow v_r^G}^{M2G}, v_s^M, v_r^G \right] \right). \quad (5.21)$$

En esta ecuación, $e_{v_s^M \rightarrow v_r^G}^{M2G}$ representa las características originales de la arista, v_s^M las características latentes del nodo de malla emisor (obtenidas del procesador) y v_r^G las características del nodo de rejilla receptor (calculadas previamente en el codificador). La concatenación de estos tres vectores se procesa mediante un MLP específico para aristas Mesh2Grid.

Una vez actualizadas las aristas, se procede a actualizar los nodos de la rejilla. Para cada

nodo v_i^G , se agregan los mensajes entrantes de las aristas que lo conectan con la malla, y el resultado se combina con su representación anterior mediante otro MLP:

$$v_i^{G'} = \text{MLP}_{\text{Mesh2Grid VG}} \left(\left[v_i^G, \sum_{e_{v_s^M \rightarrow v_r^G}^{M2G}: v_r^G = v_i^G} e_{v_s^M \rightarrow v_r^G}^{M2G'} \right] \right). \quad (5.22)$$

Aquí, el nodo v_i^G incorpora las características actualizadas de todas las aristas que tienen como destino ese nodo, mediante una operación de suma. El MLP correspondiente se encarga de calcular una nueva representación latente que integra esta información.

Una vez completada la actualización, se aplica una conexión residual para preservar la información del nodo original y estabilizar el entrenamiento:

$$v_i^G \leftarrow v_i^G + v_i^{G'}. \quad (5.23)$$

Finalmente, se obtiene la predicción $\hat{y}_i$ para cada nodo de la rejilla a partir de su representación latente final, usando un MLP adicional:

$$\hat{y}_i = \text{MLP}_{\text{Output VG}} (v_i^G). \quad (5.24)$$

Esta predicción representa las 227 variables meteorológicas estimadas en el punto correspondiente de la rejilla. Una vez obtenidas las salidas para todos los nodos, se agrupan para formar la actualización del estado global del sistema, denotada como Y^t . Esta actualización se suma al estado más reciente de entrada X^t para obtener el siguiente estado del sistema meteorológico:

$$X^{t+1} = \text{GraphCast}(X^t, X^{t-1}) = X^t + Y^t. \quad (5.25)$$

Cabe señalar que GraphCast toma como entrada los dos estados meteorológicos más recientes (X^t y X^{t-1}) y predice de forma autorregresiva el siguiente estado X^{t+1} . Además, la salida del modelo, Y^t , se reescala multiplicándola por la desviación estándar de las diferencias temporales (por nivel de presión y por variable), lo que garantiza una correcta interpretación física de las magnitudes predichas.

5.3. El modelo Hi-LAM

El trabajo de Joel Oskarsson et al., titulado *Graph-based Neural Weather Prediction for Limited Area Modeling* [Oskarsson et al., 2023], propone una adaptación de los esquemas de Predicción Meteorológica Neuronal basados en grafos (NeurWP) al contexto de la modelización de áreas limitadas (LAM). Este enfoque complementa y especializa el modelo global GraphCast, centrándose en los retos propios de la predicción meteorológica regional de alta resolución, inspirándose en los principios arquitectónicos desarrollados por Google.

A diferencia de GraphCast, orientado a pronósticos globales a medio plazo, los modelos LAM están diseñados para generar previsiones detalladas en dominios geográficos acotados. En particular, Oskarsson valida su propuesta en la región nórdica utilizando datos del sistema MEPS (MetCoOp Ensemble Prediction System) [Müller et al., 2017], demostrando así su aplicabilidad práctica y su capacidad para mejorar el rendimiento en escenarios regionales.

La principal aportación de este trabajo, en relación con el presente trabajo, radica en el aprovechamiento de la flexibilidad de las GNNs para construir estructuras espaciales adaptadas a regiones específicas. De este modo, se superan las limitaciones de las rejillas en latitud-longitud, cuya representación desigual en zonas polares conduce a una sobre-densidad de nodos y a un incremento notable del coste computacional. Sobre la base de la arquitectura modular de GraphCast, se proponen a continuación dos variantes del modelo aplicadas a áreas limitadas:

GC-LAM (GraphCast–Limited Area Model) GC-LAM constituye una adaptación directa del modelo GraphCast a dominios geográficos acotados, basada en la combinación de grafos icosaédricos de distintas resoluciones. No obstante, esta configuración inicial presentó limitaciones relevantes: se observaron artefactos circulares en las predicciones, como se muestra en la Figura 5.5, atribuibles a la presencia de nodos con un alto grado de conexión. Estos nodos resultaban sobrecargados al tener que procesar simultáneamente información local y remota.

Con el propósito de mitigar dichos artefactos —claramente identificables en la Figura 5.2— se propuso una segunda variante arquitectónica. Esta alternativa mantiene el esquema modular de GraphCast basado en *encode–process–decode*, e incorpora mecanismos de paso de mensajes inspirados en redes de interacción [Battaglia et al., 2016]. El modelo resultante, denominado Hi-LAM, se describe con mayor detalle en el siguiente apartado.

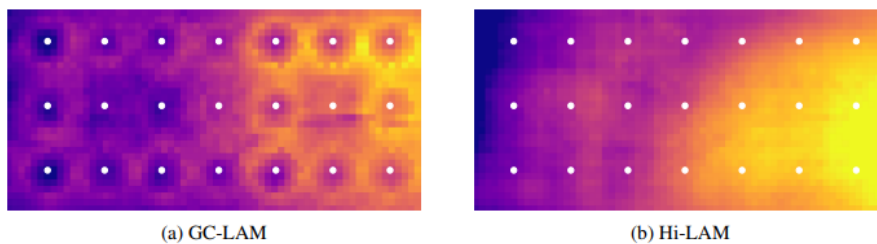


Figura 5.2: Representación del problema de los artefactos encontrados por Oskarsson, donde los puntos blancos se corresponden con nodos de la malla. Fuente: Oskarsson et al. [2023].

Hi-LAM (*Hierarchical Limited Area Model*) Para superar las limitaciones observadas en GC-LAM, particularmente la aparición de artefactos visuales derivados de nodos con alta conectividad, Oskarsson y su equipo propusieron una arquitectura alternativa denominada Hi-LAM. Este nuevo modelo introduce una estructura jerárquica de grafos con múltiples niveles de resolución espacial, manteniendo separados los distintos niveles de malla en lugar de fusionarlos como se hacía en GraphCast. De este modo, cada nivel representa una escala diferente y está conectado mediante aristas adicionales que

permiten el flujo de información entre niveles consecutivos. Esta organización jerárquica habilita una parametrización más específica, con perceptrones multicapa (MLPs) independientes para cada escala, lo que permite capturar dinámicas espaciales heterogéneas y reduce significativamente tanto la aparición de artefactos como el error cuadrático medio (RMSE) observado en las predicciones.

El modelo Hi-LAM conserva la filosofía de diseño de GraphCast, basada en un esquema de codificación–procesamiento–decodificación con GNNs, apoyándose en redes de interacción para las actualizaciones de estados de nodos y aristas [Battaglia et al., 2016, 2018]. Sin embargo, redefine y amplía cada una de estas fases para adaptarlas a su estructura jerárquica. A diferencia de la jerarquía plana empleada en GraphCast —que fusiona mallas icosaédricas de distintas resoluciones superponiéndolas sobre un conjunto común de nodos de alta resolución—, Hi-LAM mantiene una separación explícita entre los distintos niveles. Así, el nivel 1 se conecta directamente con la cuadrícula de entrada y contiene una mayor densidad de nodos, mientras que el nivel L representa la resolución más baja, caracterizada por una menor densidad y mayor separación espacial entre nodos. Aunque esta jerarquía puede visualizarse de forma vertical (como en la Figura 5.5), todos los grafos están definidos sobre la superficie terrestre, sin incorporar una componente vertical en la posición de los nodos.

Una diferencia adicional reside en la naturaleza de las mallas utilizadas. En lugar de emplear mallas icosaédricas, Hi-LAM adopta mallas cuadriláteras regulares, en las que los nodos se disponen en una rejilla de filas y columnas y se conectan bidireccionalmente con sus ocho vecinos adyacentes —es decir, horizontal, vertical y diagonalmente— como se observa en la Figura 5.3. Para representar múltiples escalas espaciales, este patrón de conectividad se replica en cuatro niveles, aumentando la separación entre nodos por un factor de tres en cada nivel sucesivo. Este diseño permite alinear estructuralmente los distintos niveles jerárquicos: cada nodo en un nivel superior se sitúa en el centro de un bloque 3×3 de nodos del nivel inferior, como se muestra en la Figura 5.4, lo que garantiza una correspondencia espacial consistente entre resoluciones.

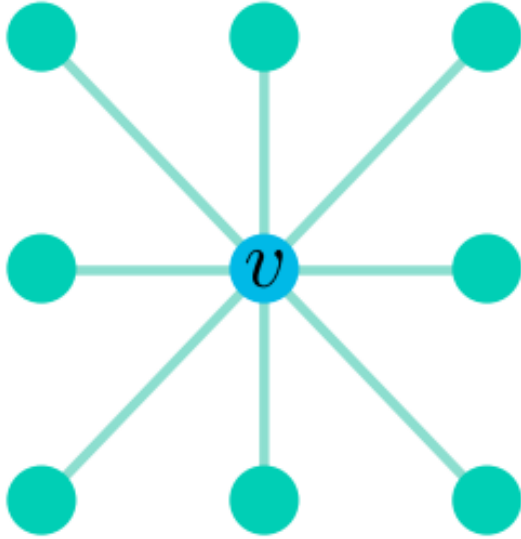


Figura 5.3: Conectividad de un nodo en su propio nivel. Fuente: Oskarsson et al. [2023].

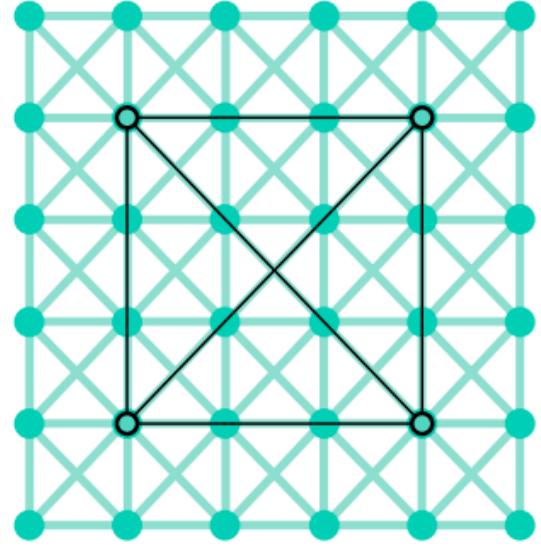


Figura 5.4: Posicionamiento de nodos entre niveles. Fuente: Oskarsson et al. [2023].

Hi-LAM emplea un conjunto más diverso de tipos de aristas en comparación con GraphCast, con el fin de manejar de forma efectiva su estructura jerárquica. En primer lugar, las aristas $\mathcal{E}_{G2M}$ y $\mathcal{E}_{M2G}$ conectan los nodos de la cuadrícula con los nodos de la malla, de manera análoga a como se hace en GraphCast. Sin embargo, en Hi-LAM estas conexiones están restringidas al nivel más bajo de la jerarquía, es decir, $\ell = 1$. Este diseño evita que las capas superiores reciban información directa de la cuadrícula de entrada, forzando una agregación progresiva desde resoluciones más finas a más gruesas.

Por otro lado, se incluyen aristas internas dentro de cada nivel de malla, representadas por $\mathcal{E}^{[\ell \leftrightarrow \ell]}$, las cuales son bidireccionales y permiten la propagación de información local entre nodos vecinos dentro de una misma escala espacial. A esto se suman las conexiones jerárquicas entre niveles consecutivos. Las aristas $\mathcal{E}^{[\ell \nearrow \ell+1]}$ permiten transmitir información ascendente desde el nivel ℓ hacia el nivel inmediatamente superior, mientras que las aristas $\mathcal{E}^{[\ell \searrow \ell-1]}$ cumplen la función inversa, facilitando la retropropagación descendente desde niveles más gruesos hacia niveles más detallados.

Una característica fundamental del diseño de Hi-LAM es que los perceptrones multicapa (MLPs) empleados para actualizar tanto los nodos como las aristas son independientes para cada tipo de conexión y para cada nivel de la jerarquía. Esta decisión de diseño permite una parametrización diferenciada por escala, adaptando el comportamiento del modelo a las características espaciales propias de cada nivel. Este planteamiento contrasta con el de GraphCast, donde los MLPs se comparten entre los distintos niveles de la multi-malla, limitando la especialización por resolución.

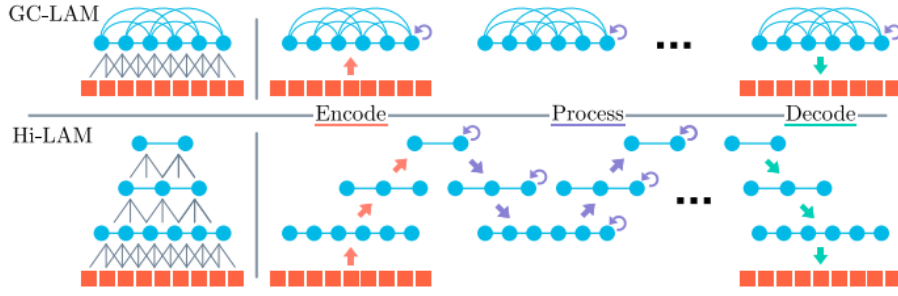


Figura 5.5: Comparativa de las estructuras de mallas de GC-LAM y Hi-LAM. Fuente: Oskarsson et al. [2023].

Codificación

En el modelo GraphCast, la fase de codificación consiste en un único paso de GNN que proyecta las características de la cuadrícula de entrada sobre los nodos de la multi-malla. Esta proyección se lleva a cabo mediante redes de interacción [Battaglia et al., 2016], que actualizan tanto las aristas como los nodos implicados en la proyección. Primero, las aristas que conectan nodos de la cuadrícula con nodos de la malla (Grid2Mesh) se actualizan mediante:

$$e'_{G2M_{v_s \rightarrow v_r}} = \text{MLP}_{\text{Grid2Mesh}_{EG2M}} \left([e_{G2M_{v_s \rightarrow v_r}}, v_{G_s}, v_{M_r}] \right), \quad (5.26)$$

Posteriormente, se actualizan los nodos de la malla agregando la información de las aristas entrantes actualizadas:

$$v'_{M_i} = \text{MLP}_{\text{Grid2Mesh}_{VM}} \left(\left[v_{M_i}, \sum_{\substack{e_{G2M_{v_s \rightarrow v_r}} \\ v_r = v_{M_i}}} e'_{G2M_{v_s \rightarrow v_r}} \right] \right), \quad (5.27)$$

A su vez, los nodos de la cuadrícula también son actualizados, aunque en este caso sin agregación:

$$v'_{G_i} = \text{MLP}_{\text{Grid2Mesh}_{VG}}(v_{G_i}). \quad (5.28)$$

Estas actualizaciones se integran finalmente mediante conexiones residuales que preservan parte de la representación original:

$$v_{G_i} \leftarrow v_{G_i} + v'_{G_i}, \quad v_{M_i} \leftarrow v_{M_i} + v'_{M_i}, \quad e_{G2M_{v_s \rightarrow v_r}} \leftarrow e_{G2M_{v_s \rightarrow v_r}} + e'_{G2M_{v_s \rightarrow v_r}}. \quad (5.29)$$

En contraste con este esquema único de codificación, Hi-LAM propone una estrategia jerárquica que propaga la información desde la cuadrícula hacia la parte superior de la malla. Inicialmente, se transfiere información desde la cuadrícula a la malla de nivel

inferior ($\ell = 1$) mediante:

$$\tilde{e}_{G2M_{s \rightarrow r}} \leftarrow \text{MLP}_{G2ME} \left(\left[e_{G2M_{s \rightarrow r}}, v_{G_s}, v_{M_r}^{[1]} \right] \right), \quad (5.30)$$

$$e_{G2M_{s \rightarrow r}} \leftarrow e_{G2M_{s \rightarrow r}} + \tilde{e}_{G2M_{s \rightarrow r}}, \quad (5.31)$$

$$v_{G_s} \leftarrow v_{G_s} + \text{MLP}_{G2MG}(v_{G_s}), \quad (5.32)$$

$$v_{M_r}^{[1]} \leftarrow v_{M_r}^{[1]} + \text{MLP}_{M_{\text{Enc}}}^{[1]} \left(\left[v_{M_r}^{[1]}, \sum_{s:(s,r) \in \mathcal{E}_{G2M}} \tilde{e}_{G2M_{s \rightarrow r}} \right] \right). \quad (5.33)$$

Una vez completado este primer paso, la información se transmite progresivamente a los niveles superiores de la jerarquía. Para cada nivel $\ell = 2, \dots, L$, la propagación se formaliza mediante:

$$\tilde{e}_{s \rightarrow r}^{[\ell-1] \nearrow \ell} \leftarrow \text{MLP}_{\text{Enc}}^{[\ell-1] \nearrow \ell} \left(\left[e_{s \rightarrow r}^{[\ell-1] \nearrow \ell}, v_{M_s}^{[\ell-1]}, v_{M_r}^{[\ell]} \right] \right), \quad (5.34)$$

$$e_{s \rightarrow r}^{[\ell-1] \nearrow \ell} \leftarrow e_{s \rightarrow r}^{[\ell-1] \nearrow \ell} + \tilde{e}_{s \rightarrow r}^{[\ell-1] \nearrow \ell}, \quad (5.35)$$

$$v_{M_r}^{[\ell]} \leftarrow v_{M_r}^{[\ell]} + \text{MLP}_{\text{Enc}}^{[\ell]} \left(\left[v_{M_r}^{[\ell]}, \sum_{s:(s,r) \in \mathcal{E}^{[\ell-1] \nearrow \ell}} \tilde{e}_{s \rightarrow r}^{[\ell-1] \nearrow \ell} \right] \right). \quad (5.36)$$

Esta estrategia marca una diferencia fundamental respecto a GraphCast: mientras este último realiza una única transformación sobre una malla plana, Hi-LAM construye progresivamente una representación jerárquica multiescala mediante la activación de MLPs específicos para cada etapa de propagación inter-nivel.

5.3.1. Procesamiento

En el modelo GraphCast, el procesamiento se realiza mediante 16 capas secuenciales de GNN, cada una operando sobre la multi-malla con parámetros no compartidos entre capas. En una capa típica, primero se actualizan las aristas que conectan pares de nodos en la malla (Mesh2Mesh) mediante:

$$e'_{M_{v_s \rightarrow v_r}} = \text{MLP}_{\text{MeshEM}} \left([e_{M_{v_s \rightarrow v_r}}, v_{M_s}, v_{M_r}] \right). \quad (5.37)$$

A continuación, los nodos de la malla se actualizan agregando la información de las aristas entrantes ya actualizadas:

$$v'_{M_i} = \text{MLP}_{\text{MeshVM}} \left(\left[v_{M_i}, \sum_{e_{M_{v_s \rightarrow v_r}}: v_r = v_{M_i}} e'_{M_{v_s \rightarrow v_r}} \right] \right). \quad (5.38)$$

Finalmente, las actualizaciones se integran mediante conexiones residuales que preservan información previa:

$$v_{M_i} \leftarrow v_{M_i} + v'_{M_i}, \quad e_{M_{v_s \rightarrow v_r}} \leftarrow e_{M_{v_s \rightarrow v_r}} + e'_{M_{v_s \rightarrow v_r}}. \quad (5.39)$$

En contraposición, Hi-LAM redefine el procesamiento como un barrido jerárquico a través de los niveles de resolución. Cada capa realiza dos pasos consecutivos: primero un barrido descendente, desde el nivel superior L hasta el más bajo, y luego un barrido ascendente, desde el nivel inferior hasta el superior. En ambos casos, se utilizan MLPs independientes para cada tipo de arista y cada nivel, permitiendo una parametrización adaptada a la escala.

Durante el barrido descendente, para cada nivel $\ell = L, \dots, 1$, se actualizan las aristas y nodos internos al nivel, seguidos de las conexiones hacia el nivel inmediatamente inferior. Las operaciones se detallan a continuación:

$$\tilde{e}_{s \rightarrow r}^{[\ell \leftrightarrow \ell]} \leftarrow \text{MLP}_{k \downarrow}^{[\ell \leftrightarrow \ell]} \left([e_{s \rightarrow r}^{[\ell \leftrightarrow \ell]}, v_{M_s}^{[\ell]}, v_{M_r}^{[\ell]}] \right), \quad (5.40)$$

$$e_{s \rightarrow r}^{[\ell \leftrightarrow \ell]} \leftarrow e_{s \rightarrow r}^{[\ell \leftrightarrow \ell]} + \tilde{e}_{s \rightarrow r}^{[\ell \leftrightarrow \ell]}, \quad (5.41)$$

$$v_{M_r}^{[\ell]} \leftarrow v_{M_r}^{[\ell]} + \text{MLP}_{k \downarrow}^{[\ell]} \left(\left[v_{M_r}^{[\ell]}, \sum_{s:(s,r) \in \mathcal{E}^{[\ell \leftrightarrow \ell]}} \tilde{e}_{s \rightarrow r}^{[\ell \leftrightarrow \ell]} \right] \right), \quad (5.42)$$

$$\tilde{e}_{s \rightarrow r}^{[\ell \searrow \ell-1]} \leftarrow \text{MLP}_k^{[\ell \searrow \ell-1]} \left([e_{s \rightarrow r}^{[\ell \searrow \ell-1]}, v_{M_s}^{[\ell]}, v_{M_r}^{[\ell-1]}] \right), \quad (5.43)$$

$$e_{s \rightarrow r}^{[\ell \searrow \ell-1]} \leftarrow e_{s \rightarrow r}^{[\ell \searrow \ell-1]} + \tilde{e}_{s \rightarrow r}^{[\ell \searrow \ell-1]}, \quad (5.44)$$

$$v_{M_r}^{[\ell-1]} \leftarrow v_{M_r}^{[\ell-1]} + \text{MLP}_{k \searrow}^{[\ell-1]} \left(\left[v_{M_r}^{[\ell-1]}, \sum_{s:(s,r) \in \mathcal{E}^{[\ell \searrow \ell-1]}} \tilde{e}_{s \rightarrow r}^{[\ell \searrow \ell-1]} \right] \right). \quad (5.45)$$

Posteriormente, el barrido ascendente recorre los niveles en orden creciente $\ell = 1, \dots, L$, actualizando las aristas internas y propagando información hacia los niveles superiores:

$$\tilde{e}_{s \rightarrow r}^{[\ell \leftrightarrow \ell]} \leftarrow \text{MLP}_{k \uparrow}^{[\ell \leftrightarrow \ell]} \left([e_{s \rightarrow r}^{[\ell \leftrightarrow \ell]}, v_{M_s}^{[\ell]}, v_{M_r}^{[\ell]}] \right), \quad (5.46)$$

$$e_{s \rightarrow r}^{[\ell \leftrightarrow \ell]} \leftarrow e_{s \rightarrow r}^{[\ell \leftrightarrow \ell]} + \tilde{e}_{s \rightarrow r}^{[\ell \leftrightarrow \ell]}, \quad (5.47)$$

$$v_{M_r}^{[\ell]} \leftarrow v_{M_r}^{[\ell]} + \text{MLP}_{k \uparrow}^{[\ell]} \left(\left[v_{M_r}^{[\ell]}, \sum_{s:(s,r) \in \mathcal{E}^{[\ell \leftrightarrow \ell]}} \tilde{e}_{s \rightarrow r}^{[\ell \leftrightarrow \ell]} \right] \right), \quad (5.48)$$

$$\tilde{e}_{s \rightarrow r}^{[\ell \nearrow \ell+1]} \leftarrow \text{MLP}_k^{[\ell \nearrow \ell+1]} \left([e_{s \rightarrow r}^{[\ell \nearrow \ell+1]}, v_{M_s}^{[\ell]}, v_{M_r}^{[\ell+1]}] \right), \quad (5.49)$$

$$e_{s \rightarrow r}^{[\ell \nearrow \ell+1]} \leftarrow e_{s \rightarrow r}^{[\ell \nearrow \ell+1]} + \tilde{e}_{s \rightarrow r}^{[\ell \nearrow \ell+1]}, \quad (5.50)$$

$$v_{M_r}^{[\ell+1]} \leftarrow v_{M_r}^{[\ell+1]} + \text{MLP}_{k \nearrow}^{[\ell+1]} \left(\left[v_{M_r}^{[\ell+1]}, \sum_{s:(s,r) \in \mathcal{E}^{[\ell \nearrow \ell+1]}} \tilde{e}_{s \rightarrow r}^{[\ell \nearrow \ell+1]} \right] \right). \quad (5.51)$$

Este diseño jerárquico y el uso de MLPs diferenciados por nivel permiten a Hi-LAM abordar una de las principales limitaciones de GraphCast: la sobrecarga de los nodos que, en la multi-malla fusionada, debían manejar simultáneamente relaciones locales y de largo alcance. Al distribuir la responsabilidad entre escalas y especializar el procesamiento por nivel, Hi-LAM logra una modelización espacial más precisa y reduce notablemente los artefactos visuales observados en versiones anteriores.

5.3.2. Decodificación

En el modelo GraphCast, la fase de decodificación consiste en un único paso de GNN que transforma las representaciones obtenidas en la multi-malla de vuelta a la cuadrícula de latitud-longitud. Este proceso se lleva a cabo mediante redes de interacción que actualizan tanto las aristas como los nodos implicados en la operación de retorno. En primer lugar, se actualizan las aristas *Mesh2Grid* conectando los nodos de la malla con los nodos de la cuadrícula, mediante la siguiente expresión:

$$e'_{\text{M2G}_{v_s \rightarrow v_r}} = \text{MLP}_{\text{Mesh2Grid}_{\text{EM2G}}} \left([e_{\text{M2G}_{v_s \rightarrow v_r}}, v_{M_s}, v_{G_r}] \right). \quad (5.52)$$

Posteriormente, los nodos de la cuadrícula se actualizan agregando la información proveniente de las aristas entrantes recién actualizadas:

$$v'_{G_i} = \text{MLP}_{\text{Mesh2Grid}_{\text{VG}}} \left(\left[v_{G_i}, \sum_{\substack{e_{\text{M2G}_{v_s \rightarrow v_r}} \\ v_r = v_{G_i}}} e'_{\text{M2G}_{v_s \rightarrow v_r}} \right] \right). \quad (5.53)$$

Finalmente, estas actualizaciones se integran mediante una conexión residual, de modo que la representación final del nodo en la cuadrícula incorpora tanto su estado anterior como la nueva información obtenida:

$$v_{G_i} \leftarrow v_{G_i} + v'_{G_i}. \quad (5.54)$$

El modelo Hi-LAM, por su parte, redefine el mecanismo de decodificación como una operación jerárquica en dos fases. En una primera etapa, se propaga información desde los niveles superiores hacia los inferiores de la jerarquía. Para cada nivel $\ell = L - 1, \dots, 1$, se actualizan las aristas que conectan niveles adyacentes mediante:

$$\tilde{e}_{s \rightarrow r}^{[\ell+1 \searrow \ell]} \leftarrow \text{MLP}_{\text{Dec.}}^{[\ell+1 \searrow \ell]} \left([e_{s \rightarrow r}^{[\ell+1 \searrow \ell]}, v_{M_s}^{[\ell+1]}, v_{M_r}^{[\ell]}] \right), \quad (5.55)$$

y a continuación, los nodos del nivel inferior se actualizan incorporando la información agregada desde el nivel superior:

$$v_{M_r}^{[\ell]} \leftarrow v_{M_r}^{[\ell]} + \text{MLP}_{\text{Dec}}^{[\ell]} \left(\left[v_{M_r}^{[\ell]}, \sum_{s:(s,r) \in \mathcal{E}^{[\ell+1] \setminus \setminus \ell}} \tilde{c}_{s \rightarrow r}^{[\ell+1] \setminus \setminus \ell} \right] \right). \quad (5.56)$$

Una vez completada esta propagación descendente jerárquica, se realiza la decodificación final desde el nivel más bajo de la malla hacia la cuadrícula. Primero, se actualizan las aristas que conectan la malla con la cuadrícula:

$$\tilde{e}_{M2G_{s \rightarrow r}} \leftarrow \text{MLP}_{M2G_E} \left([e_{M2G_{s \rightarrow r}}, v_{M_s}^{[1]}, v_{G_r}] \right), \quad (5.57)$$

y finalmente, los nodos de la cuadrícula se actualizan con la agregación correspondiente:

$$v_{G_r} \leftarrow v_{G_r} + \text{MLP}_{M2G_V} \left(\left[v_{G_r}, \sum_{s:(s,r) \in \mathcal{E}_{M2G}} \tilde{e}_{M2G_{s \rightarrow r}} \right] \right). \quad (5.58)$$

Este procedimiento jerárquico y secuencial permite que la información obtenida en cada nivel de resolución se integre de forma estructurada y coherente en la predicción final, mejorando la capacidad del modelo para capturar patrones multiescala sin introducir artefactos espaciales.

5.3.3. Predicción final

Tanto la implementación basada en GraphCast como el modelo Hi-LAM siguen una filosofía común de predicción residual para estimar el estado meteorológico futuro. Tras completar el ciclo de codificación, procesamiento y decodificación (*encode-process-decode*), el decodificador produce una predicción interna $\hat{X}_t$ que representa la estimación del siguiente estado temporal. Esta predicción se obtiene mediante una capa MLP aplicada sobre las representaciones finales de los nodos de la cuadrícula, y su cálculo se expresa como:

$$\hat{X}_t = \text{Model}(X_{t-1}, X_{t-2}) = X_{t-1} + \text{MLP}_{\text{pred}}(v_G), \quad (5.59)$$

donde X_{t-1} y X_{t-2} son los dos estados anteriores del sistema utilizados como entrada, lo que mejora la estabilidad y la precisión del pronóstico. El vector v_G representa las características finales de los nodos de la cuadrícula generadas por la red neuronal gráfica, mientras que $\text{MLP}_{\text{pred}}(v_G)$ estima la diferencia residual entre el estado anterior y la predicción actual.

Una vez obtenida esta predicción, se aplica un mecanismo de forzamiento de contorno (*boundary forcing*) para incorporar información externa en la región limítrofe del dominio. Esta operación asegura la coherencia con las condiciones del modelo anfitrión global en las zonas fronterizas. La integración de este forzamiento se formaliza mediante la siguiente ecuación:

$$\hat{X}_t^v \leftarrow (1 - \mathbb{I}_{\{v \in B\}}) \hat{X}_t^v + \mathbb{I}_{\{v \in B\}} X_{t,\text{host}}^v, \quad (5.60)$$

donde $\mathbb{I}_{\{v \in B\}}$ es una función indicadora que toma el valor 1 si el nodo v pertenece a la franja de borde B , y 0 en caso contrario. De este modo, las predicciones realizadas sobre los nodos internos del dominio se conservan tal como han sido generadas por la GNN, mientras que aquellas correspondientes a nodos en el borde son reemplazadas por los valores provenientes del modelo global anfitrión $X_{t,\text{host}}^v$.

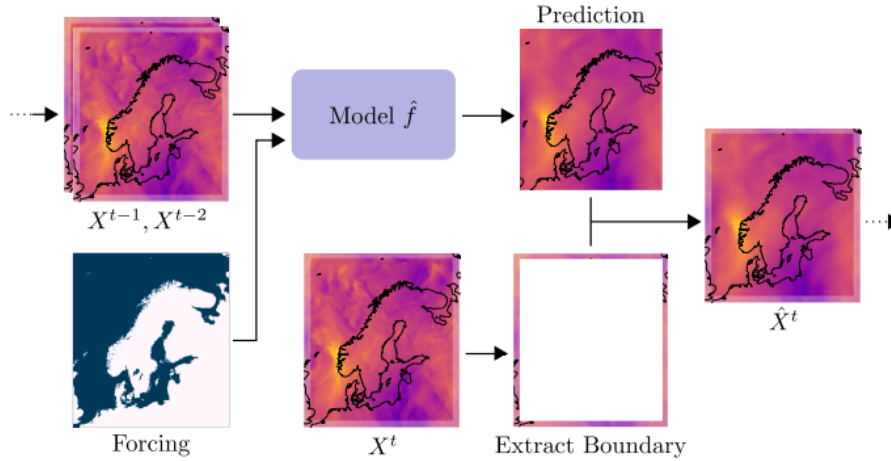


Figura 5.6: Representación del proceso de predicción con aplicación del forzamiento de contorno. Fuente: [Oskarsson et al., 2023]

Este protocolo, basado en una predicción residual seguida de un forzamiento de contorno, garantiza tanto la consistencia interna del modelo como la correcta inyección de información externa. Así, el sistema logra mantener el contexto global necesario para mejorar la predicción en el interior del dominio en cada paso autorregresivo.

5.3.4. Resultados y beneficios

Los experimentos realizados con Hi-LAM han mostrado una reducción significativa de los artefactos visuales, como los observados en GC-LAM, así como una mejora cuantitativa en términos de error cuadrático medio global (RMSE). Estos resultados destacan la relevancia de la estructura jerárquica y la parametrización diferenciada por escala, que permiten al modelo aprender fenómenos atmosféricos de forma más eficiente y adaptada a distintas resoluciones espaciales.

Pese a que Hi-LAM es un modelo más grande y con un proceso de inferencia secuencial, el coste computacional añadido es moderado: el tiempo de entrenamiento se incrementa únicamente en un 12 % con respecto a GC-LAM. Esta relación favorable entre complejidad y rendimiento convierte a Hi-LAM en una alternativa viable y poderosa para la predicción meteorológica regional.

Finalmente, los autores subrayan una limitación común a los modelos de predicción meteorológica neuronal (NeurWP): la tendencia a generar salidas excesivamente suaves. Este

aspecto pone de manifiesto la necesidad de avanzar hacia enfoques probabilísticos o generativos que introduzcan mayor variabilidad y realismo en las predicciones. En conjunto, el modelo Hi-LAM propuesto por Oskarsson et al. representa un paso importante hacia sistemas de predicción regional más precisos y detallados, lo cual reviste especial interés para los objetivos de este proyecto, en el contexto del análisis de datos geofísicos y su aplicación a la mitigación y adaptación al cambio climático.

5.4. El modelo SeaCast

El modelo SeaCast comparte su estructura general con el modelo Hi-LAM propuesto por Oskarsson, en tanto que también está diseñado para operar en un dominio de área limitada. Sin embargo, mientras Hi-LAM se orienta a la predicción meteorológica, SeaCast se focaliza en variables oceanográficas, centrándose específicamente en la región del mar Mediterráneo. El entrenamiento del modelo se llevó a cabo utilizando datos de reanálisis desde enero de 1987 hasta diciembre de 2021 [Escudier et al., 2021], así como datos de análisis desde enero de 2022 hasta abril de 2024 [Clementi et al., 2023]. Para la validación se utilizaron datos de análisis correspondientes a los meses de mayo a julio de 2024, y para la evaluación final (conjunto de test) se consideraron datos de análisis y simulaciones de los meses de julio y agosto de 2024.

SeaCast adopta una arquitectura jerárquica de mallas, similar a la empleada por Hi-LAM, que combina estructuras cuadrículas como las ilustradas en la Figura 5.5 con una organización multinivel, mostrada en la Figura 5.4. Para su adaptación al entorno oceánico, la malla se modifica eliminando todos los nodos situados sobre tierra firme, así como las aristas que los atraviesan.

El modelo realiza pronósticos sobre un conjunto de variables oceánicas que incluyen la velocidad del agua de mar hacia el este (**uo**), hacia el norte (**vo**), el espesor de la capa mixta oceánica (**m1otst**), la salinidad (**so**), la altura de la superficie del mar respecto al geoide (**zos**), la temperatura potencial del agua de mar (**thetao**) y la temperatura potencial en el fondo oceánico (**bottomT**). Tres de estas variables corresponden a un único nivel (ya sea superficial o de fondo), mientras que las otras cuatro se modelan a múltiples niveles de profundidad. En total, se manejan 75 variables distribuidas a lo largo de 18 niveles verticales, que abarcan hasta 200 metros de profundidad. Además, el conjunto de datos incorpora cuatro características estáticas: latitud, longitud, profundidad del lecho marino y topografía dinámica media.

Considerando la complejidad del sistema, la función de pérdida del modelo se define como la suma del error cuadrático medio (MSE) para cada variable oceanográfica i , nivel de profundidad l y nodo v de la malla. La función objetivo $\mathcal{L}$ queda formalmente expresada como:

$$\mathcal{L} = \frac{1}{T_{\text{rollout}}} \sum_{t=1}^{T_{\text{rollout}}} \sum_{i=1}^C \sum_{l=1}^{L_i} \frac{1}{|G_l|} \sum_{v \in G_l} a_v \lambda_i \left(\hat{X}_{v,i}^t - X_{v,i}^t \right)^2, \quad (5.61)$$

donde T_{rollout} representa el horizonte temporal de predicción, C es el número total de



Figura 5.7: Imagen representativa del dominio del modelo SeaCast. El color marrón indica el forzante de contorno, el verde la capa superficial del mar y el azul representa las zonas de mayor profundidad. Fuente: [Holmberg et al., 2024]

variables climáticas u oceanográficas consideradas, L_i el número de niveles para la variable i , G_l el conjunto de nodos en el nivel l , a_v un peso asociado al nodo v (por ejemplo, su área relativa), λ_i el peso específico asignado a la variable i en la función de pérdida, y $\hat{X}_{v,i}^t$ y $X_{v,i}^t$ los valores predichos y observados, respectivamente.

Para lograr una simulación realista del entorno marino, SeaCast incorpora forzamientos externos que proceden tanto de la atmósfera como de las fronteras del dominio oceánico. En lo que respecta al forzamiento atmosférico, se incluyen cuatro variables clave: las componentes zonal y meridional del viento a 10 metros (**u10**, **v10**), la temperatura del aire a 2 metros (**t2m**) y la presión media a nivel del mar (**msl**). Estas variables se obtienen a partir de los datos de reanálisis *ERA5* [Hersbach et al., 2023], con una resolución temporal de seis horas, y posteriormente se promedian de forma diaria para su uso en el modelo.

La calidad de estas variables se evalúa mediante su comparación con predicciones atmosféricas procedentes de dos sistemas operacionales del Centro Europeo de Predicción Meteorológica a Medio Plazo (ECMWF): el Ensemble Prediction System (ENS), que proporciona un conjunto de pronósticos probabilísticos generados a partir de múltiples simulaciones con condiciones iniciales ligeramente distintas; y el Artificial Intelligence Forecasting System (AIFS), un modelo experimental basado en redes neuronales profundas desarrollado para generar predicciones deterministas a partir de datos históricos.

Antes de su integración, todas las variables atmosféricas se interpolan mediante un esquema bilineal desde su resolución original de $\frac{1}{4}^\circ$ hasta la resolución del dominio oceánico, fijada en $\frac{1}{24}^\circ$. Además, se incorporan como variables auxiliares el seno y el coseno del día del año, con el fin de introducir información estacional de forma continua.

Respecto al forzamiento de frontera, el modelo impone condiciones que permiten una transición coherente entre el dominio limitado y el océano global. Este mecanismo garantiza que las masas de agua puedan entrar y salir del dominio simulado sin generar discontinuidades. En los experimentos realizados para el Mediterráneo, la frontera se define sobre los nodos de la cuadrícula situados al oeste de la longitud $-5,2^\circ$, lo cual corresponde a la región del Estrecho de Gibraltar. En la Figura 5.7 se representa gráficamente el dominio considerado, incluyendo la ubicación del forzamiento de contorno y la estructura oceánica interna.

Capítulo 6

Creación de las mallas

En este capítulo se describen los distintos tipos de mallas propuestas en el presente trabajo, los cuales serán empleados posteriormente durante la configuración de los experimentos en el Capítulo 7.

Nos centraremos, en primer lugar, en el proceso de construcción general de las mallas y en las variaciones que presentan entre sí. Asimismo, analizaremos los objetivos específicos que se persiguen con cada tipo de malla. Para la implementación, se ha utilizado principalmente la biblioteca `networkx`, debido a la facilidad que proporciona para la manipulación de grafos y a su compatibilidad con `PyTorch Geometric`, siendo esta la librería propia para el entrenamiento de redes neuronales de grafos con `PyTorch`.

6.1. Mecanismo de creación de las mallas

El procedimiento general seguido para la generación de las mallas que constituyen el grafo de la red neuronal se compone de una secuencia de etapas que se aplican de forma sistemática en cada nivel de resolución.

En primer lugar, se genera un conjunto de nodos cuyo número viene determinado por el tipo de malla especificado por el usuario. A continuación, se eliminan todos aquellos nodos que se encuentren ubicados sobre tierra firme, con el fin de restringir la malla exclusivamente al dominio oceánico.

Una vez filtrados los nodos, se procede a crear las aristas entre ellos, de acuerdo con la política de conexión correspondiente a cada tipo de malla. Cabe destacar que estas conexiones se definen como bidireccionales, permitiendo así el flujo de información en ambas direcciones.

Posteriormente, se eliminan las aristas que atraviesan zonas terrestres, con el objetivo de preservar la consistencia geográfica de la topología. Finalmente, se asignan atributos a las aristas restantes, en concreto, la distancia euclidiana y la distancia vectorial entre los nodos que conecta cada enlace.

Dado que este trabajo se inspira en el enfoque de mallas jerárquicas del modelo Hi-LAM,

propuesto por Oskarsson, y en la arquitectura *encoder-process-decoder* de GraphCast, se incorporan tres tipos adicionales de aristas, generadas de forma coherente para todos los tipos de malla considerados. Las fórmulas empleadas siguen las definiciones introducidas en los modelos Hi-LAM y SeaCast.

Las primeras, denominadas aristas **G2M**, se generan de manera unidireccional desde los nodos de la grilla hacia los nodos de la malla más fina. Para garantizar que cada nodo de la grilla esté conectado, se establece un enlace hacia el nodo más cercano en la malla jerárquica.

En segundo lugar, las aristas **M2M** conectan, también de forma unidireccional, cada nodo de la malla más fina con sus k vecinos más cercanos en la malla inmediatamente superior. El valor de k se define como un hiperparámetro del modelo. A partir de estas conexiones descendentes se obtienen las correspondientes conexiones ascendentes, invirtiendo la dirección de las aristas originales.

Por último, las aristas **M2G** enlazan unidireccionalmente los nodos de la malla más fina con la grilla. Estas conexiones se derivan directamente invirtiendo las direcciones de las aristas **G2M**.

Cabe destacar que se han introducido modificaciones respecto a los trabajos previos. En los modelos Hi-LAM y SeaCast, las aristas **G2M** y **M2G** se generan siguiendo la formulación de GraphCast. No obstante, se observó que dicha formulación no garantizaba una conexión mínima para todos los nodos de la grilla en la zona geográfica utilizada en este estudio. Por este motivo, se ha adoptado un enfoque alternativo que asegura al menos una conexión por nodo.

Asimismo, se ha redefinido el criterio de conexión de las aristas **M2G**, estableciéndolo de forma idéntica al de las aristas **G2M**. En SeaCast, cada nodo de la grilla se conectaba con sus cuatro vecinos más cercanos, mientras que en GraphCast se establecían tres conexiones por nodo, correspondientes a una cara del icosaedro regular. En nuestro caso, se define el número de conexiones como un hiperparámetro k , lo que permite una mayor flexibilidad y robustez ante distintas configuraciones de malla.

6.2. Mallas uniformes

Esta primera familia de mallas se caracteriza por no aplicar ningún mecanismo de densificación espacial durante la generación de nodos. El objetivo es analizar qué esquemas de posicionamiento permiten cubrir con mayor fidelidad la superficie oceánica dentro de la zona seleccionada.

Estas configuraciones se emplean como referencia inicial para evaluar tanto la necesidad como el impacto de introducir mecanismos de densificación que adapten la estructura de la malla a las particularidades geográficas y dinámicas de la región de estudio.

6.2.1. Mallas uniformes estructuradas

Dentro de las mallas uniformes, se han empleado las mallas estructuradas por defecto propuestas en SeaCast. Estas se basan en una cuadrícula regular en la que cada nodo se conecta con sus vecinos inmediatos, incluyendo diagonales siempre que exista un nodo en dicha dirección. Así, cada nodo (excepto en los bordes del dominio) cuenta con hasta ocho conexiones, donde tenemos una malla a modo de ilustración de la estructura generada en la Figura 6.1.

La malla se genera con un desplazamiento uniforme respecto a los bordes, dependiendo del número total de nodos, lo que permite centrarla en el dominio geográfico. En nuestro trabajo se ha mantenido esta característica para respetar el diseño original de SeaCast. Sin embargo, consideramos que una alternativa más adecuada para ciertas aplicaciones sería situar nodos en los bordes, lo cual permitiría una mejor cobertura del dominio.

Otro aspecto que se ha modificado respecto al diseño original es la parametrización de la resolución de las mallas. Mientras que en la versión original se utilizaba un valor heurístico basado en múltiplos de 3 —donde el número de nodos por lado se aproximaba a la potencia de 3 más cercana a la resolución de la grilla (por ejemplo, 81, 27 y 9 a partir de 243)—, en nuestro enfoque se permite especificar directamente la resolución deseada como un parámetro configurable. Esta modificación permite mayor control al usuario y elimina las restricciones impuestas por el diseño original.

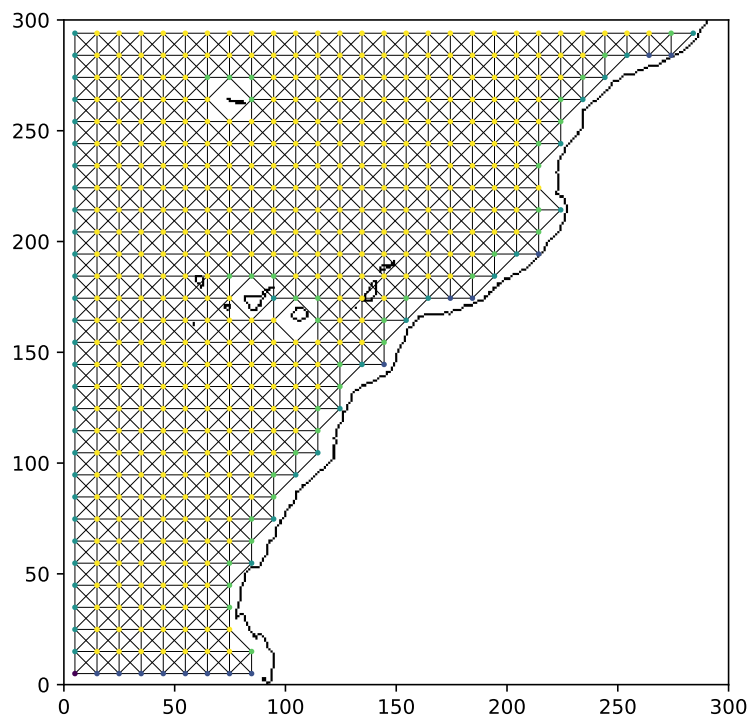


Figura 6.1: Malla uniforme estructurada cuadrangular con aristas diagonales cruzadas de 30x30 de resolución

Malla uniforme cuadrangular sin aristas cruzadas

Una modificación destacable introducida en este trabajo afecta al modo en que se definen las conexiones diagonales entre nodos en ciertas mallas. Considerando la naturaleza de las redes neuronales empleadas y las características de la zona de estudio, se ha determinado que el cruce de aristas diagonales debería evitarse, ya que podría resultar contraproducente. En lugar de establecer estas conexiones en función exclusiva de la presencia o ausencia de nodos en la diagonal, se ha optado por conectar los nodos mediante una triangulación de Delaunay, empleando para ello la implementación disponible en la librería `scipy`.

Dicho algoritmo se basa en el trabajo de [Barber et al., 1996], donde se propone un enfoque práctico y eficiente para la construcción de envolventes convexas en espacios de cualquier dimensión. Además, mediante un procedimiento geométrico adicional, también se facilita la obtención de triangulaciones de Delaunay a partir del mismo principio.

El funcionamiento de este algoritmo consiste en transformar los puntos en un espacio de mayor dimensión usando una transformación parabólica tal que:

$$(x, y) \mapsto (x, y, x^2 + y^2). \quad (6.1)$$

Posteriormente, se calcula la envolvente convexa de los puntos transformados, siendo la proyección inversa de la parte inferior de esa envolvente la triangulación de Delaunay, dando como resultado conexiones como las que podemos ver en la Figura 6.2.

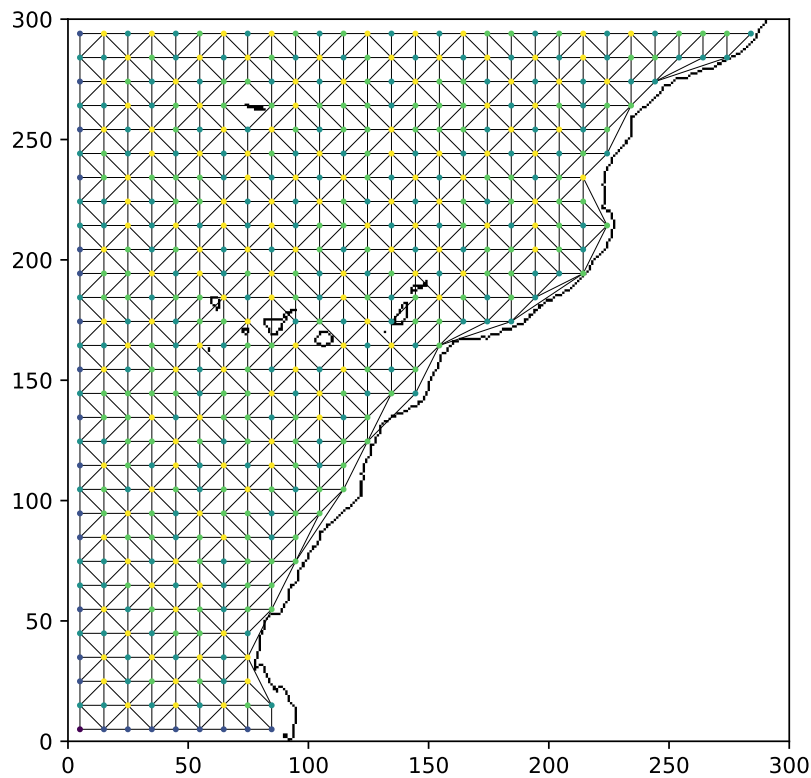


Figura 6.2: Malla uniforme estructurada cuadrangular con arista diagonal por triangulación de Delaunay de 30x30 de resolución

6.2.2. Mallas uniformes no estructuradas

En cuanto a las mallas uniformes no estructuradas, surgen de la necesidad de crear una nueva estructura de posicionamiento de nodos con el objetivo de crear una estrategia que sea capaz de cubrir la zona de estudio más eficientemente, ya que, tal y como se puede ver en las Figuras 6.1 y 6.2, al eliminar aquellos nodos que se encuentran en tierra, se acaban deformando los cuadrados en dicha zona, además de que en dicha estrategia, los nodos no se acercan demasiado a la costa, haciendo que esta zona se vea menos representada, ya no solo por falta de aristas respecto a nodos que se encuentran en la zona central de estudio, sino por dicha separación de los nodos de la costa y estructuras incompletas resultantes de la eliminación de nodos.

Malla aleatoria

Con el fin de analizar la influencia de la distribución espacial de los nodos sobre el rendimiento del modelo, se ha generado una malla en la que los nodos se colocan de forma aleatoria. Este enfoque permite investigar dos aspectos relevantes. Por un lado, se evalúa el impacto que puede tener la colocación de los nodos en la calidad de las predicciones, con el objetivo de extraer conclusiones generales sobre la sensibilidad del modelo a la geometría de entrada. Por otro, se estudia la efectividad del posicionamiento regular adoptado en SeaCast para la región objetivo, mediante una comparación directa de los resultados obtenidos con ambas configuraciones.

La malla aleatoria se utiliza, por tanto, como referencia de comparación para estimar la calidad relativa de las distintas mallas consideradas. Su construcción se realiza a partir de un parámetro n , que define el número de nodos a generar en el nivel correspondiente. En caso de que este parámetro no se especifique, se determina automáticamente a partir de la proporción tierra/agua del dominio, siguiendo el algoritmo de cálculo de resoluciones empleado en SeaCast.

Para asegurar que los nodos se sitúan exclusivamente sobre el océano, se emplea una máscara binaria tierra-agua que restringe el área disponible para la generación. Sobre cada píxel válido, se asigna una probabilidad de generación utilizando una distribución uniforme continua en el intervalo $[0, 1]$. Estas probabilidades se normalizan posteriormente de forma que su suma total sea igual a uno, obteniéndose así una distribución de probabilidad bien definida sobre el conjunto de ubicaciones posibles.

Con el fin de garantizar la reproducibilidad del proceso, se establece una semilla común para todas las operaciones que impliquen generación aleatoria. Esta misma semilla se reutiliza en cualquier otro procedimiento que dependa de mecanismos probabilísticos, lo que permite comparar de forma justa los resultados entre distintos experimentos. La expresión utilizada para normalizar los valores asignados se detalla a continuación en el Algoritmo 1.

Tras haber generado dichos pesos, procedemos a escoger aleatoriamente un par de coordenadas para crear los nodos que compondrán la malla, donde estas coordenadas son escogidas en base a los pesos generados con anterioridad, donde, además, se asegura que al menos exista un nodo en cada esquina oceánica de la región, donde también cabe

Algorithm 1 Generación de una distribución de probabilidad uniforme

Require: Máscara binaria del océano M de tamaño $H \times W$

Semilla aleatoria $s \in \mathbb{N}$

Ensure: Matriz P de tamaño $H \times W$ con una distribución de probabilidad normalizada

- 1: Establecer la semilla del generador de números aleatorios a s
 - 2: Calcular la matriz D de distancias desde cada celda oceánica hasta la tierra
 - 3: Definir el margen costero $m \leftarrow \sqrt{0,6}$
 - 4: Crear una máscara de validez V donde $V_{i,j} = 1$ si $M_{i,j} = 1$ y $D_{i,j} \geq m$, y 0 en caso contrario
 - 5: Inicializar una matriz P de ceros del mismo tamaño que M
 - 6: Contar el número de celdas válidas: $k \leftarrow \sum V$
 - 7: Generar una muestra $\{x_1, x_2, \dots, x_k\}$ de k valores aleatorios tal que $x_i \sim \mathcal{U}(0, 1)$
 - 8: Asignar estos valores x_i a las posiciones (i, j) donde $V_{i,j} = 1$
 - 9: Calcular la suma total de los valores de P : $Z \leftarrow \sum_{i,j} P_{i,j}$
 - 10: **for all** celdas (i, j) del dominio **do**
 - 11: $P_{i,j} \leftarrow P_{i,j}/Z$
 - 12: **end for**
 - 13: **return** P
-

destacar que se busca añadir al menos un nodo en las esquinas para evitar perder esta información.

Posterior al colocado de los nodos en la malla, creamos las aristas mediante la triangulación de Delaunay, donde también eliminamos aquellas aristas que crucen la tierra y se añaden las características, siendo estas tanto la distancia euclidiana como la distancia vectorial como características de las aristas.

El resultado de dicho algoritmo muestra como resultado una malla tal como la que se puede apreciar en la Figura 6.3, donde se ha utilizado la misma cantidad de nodos que en los ejemplos anteriores para conseguir una comparación más fidedigna.

Malla generada mediante muestreo del punto más lejano

Dados los resultados obtenidos en la generación de la malla aleatoria, hemos visto que este tipo de posicionamiento de nodos sin estructura nos permite adaptar la malla con mayor facilidad a la región de estudio, por lo que consideramos que un camino sin una geometría preestablecida de antemano puede obtener resultados interesantes; sin embargo, sería conveniente encontrar un algoritmo que aporte un mayor orden en la creación de la malla, ya que, tal y como se puede apreciar en la Figura 6.3, la distancia entre nodos es muy dispar.

Para solventar este problema, hemos optado por tomar los nodos siguiendo el algoritmo de muestreo del punto más lejano (comúnmente conocido como *farthest point sampling* en inglés) [Eldar et al., 1997] donde buscaremos cubrir la zona de estudio manteniendo la versatilidad que ofrece una malla no estructurada.

Con el Algoritmo 2, se busca elegir el punto más alejado del conjunto ya seleccionado, midiendo la distancia más corta posible a cualquiera de los puntos ya elegidos, evitando

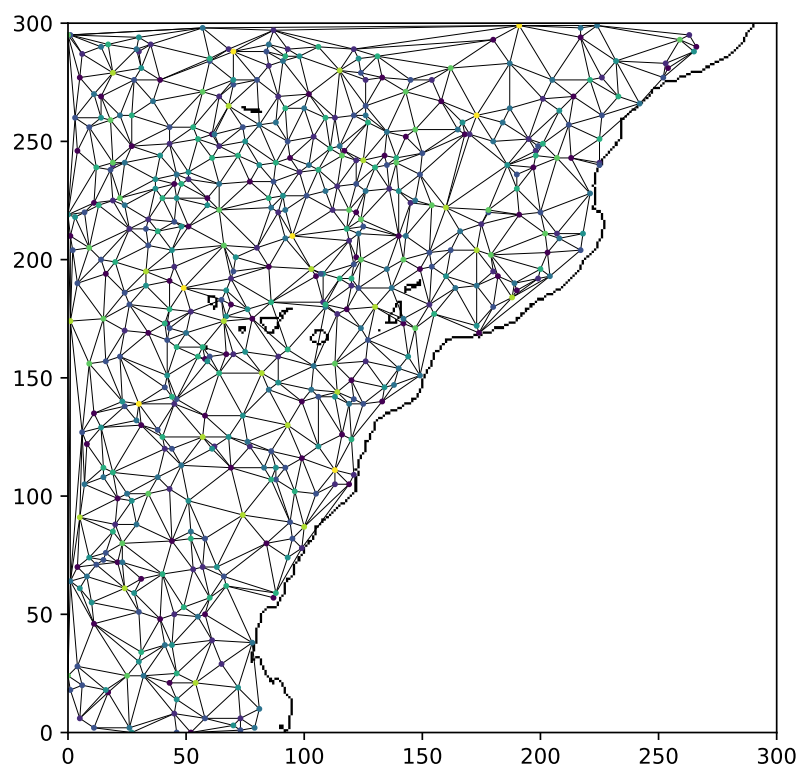


Figura 6.3: Malla generada mediante muestreo de nodos aleatorios

de esta manera que los nuevos puntos se agrupen demasiado cerca de los anteriores, tal y como ocurría en la malla aleatoria presentada anteriormente, donde también se unen los nodos de la malla mediante la triangulación de Delaunay, consiguiendo como resultado una malla como la que se puede apreciar en la Figura 6.4, donde, tal y como está reflejado en el ejemplo, con esta malla somos capaces de representar mejor las zonas costeras y cubrir completamente nuestra zona de estudio manteniendo una distancia entre nodos más uniforme.

Algorithm 2 Muestreo por el punto más lejano (Farthest Point Sampling)**Require:** Conjunto de puntos candidatos $P = \{p_1, p_2, \dots, p_N\} \subset \mathbb{R}^2$ Número de puntos a seleccionar $k \in \mathbb{N}$ **Ensure:** Subconjunto $S \subseteq P$ de hasta k puntos, seleccionados por criterio de lejanía máxima

```

1: if  $k \leq 0$  o  $N = 0$  then
2:   return conjunto vacío
3: end if
4: Inicializar el conjunto  $S$  con un punto aleatorio  $p_i \in P$ 
5: Para cada punto  $p_j \in P$ , calcular la distancia euclídea  $d_j$  hasta  $S$ , y almacenar
    $d_j = \|p_j - p_i\|$ 
6: for  $t = 2$  hasta  $\min(k, N)$  do
7:   Seleccionar el punto  $p^* \in P$  cuya distancia  $d_{p^*}$  sea máxima:
        $p^* \leftarrow \arg \max_{p_j \in P} d_j$ 
8:   Añadir  $p^*$  al conjunto  $S$ 
9:   Para cada  $p_j \in P$ , actualizar su distancia:
        $d_j \leftarrow \min(d_j, \|p_j - p^*\|)$ 
10: end for
11: return conjunto  $S$ 

```

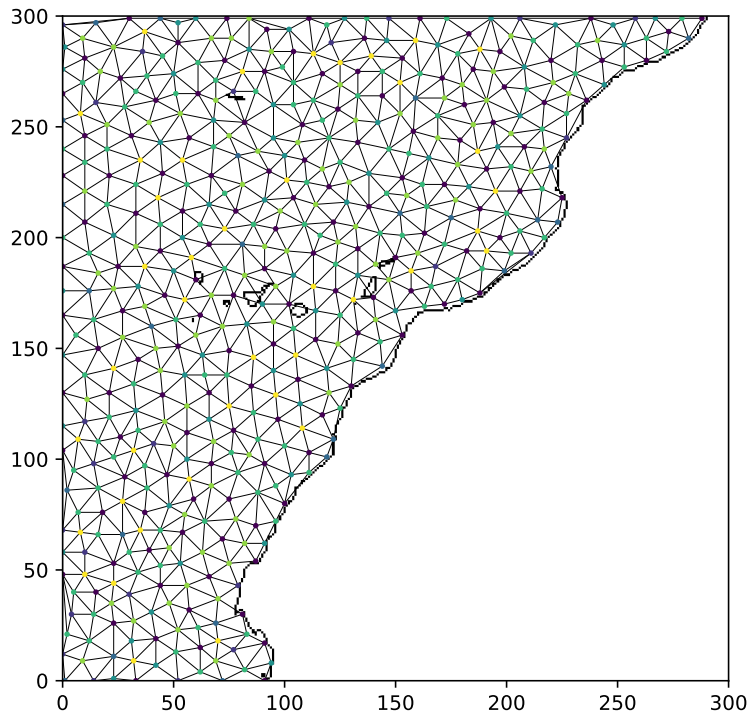


Figura 6.4: Malla generada mediante muestreo del punto más lejano

6.3. Mallas adaptativas

Con el objetivo de adaptar la malla al dominio de estudio, se ha optado por utilizar mallas estáticas en lugar de mallas dinámicas. Aunque estas últimas pueden resultar eficaces para modelar la evolución temporal de la temperatura superficial del mar, presentan importantes limitaciones en términos de coste computacional, complejidad de implementación y escalabilidad [Feng et al., 2024]. Por este motivo, se ha priorizado el diseño de mallas estáticas, explorando mecanismos de densificación espacial basados en la batimetría, con el fin de concentrar nodos en zonas costeras y mejorar la capacidad de representación en regiones geográficamente más complejas.

En este contexto, la batimetría de la zona de estudio se ha empleado como variable de referencia para los procesos de densificación propuestos, ya que, al igual que la malla, constituye una característica estática a lo largo del tiempo. Esta propiedad la convierte en un punto de partida adecuado para guiar la generación de mallas adaptadas al entorno físico en el contexto de este trabajo para la mejora de la predicción en las zonas de afloramiento, estando estas en regiones de la costa africana.

6.3.1. Mallas adaptativas estructuradas

En lo que respecta al uso de mallas adaptativas estructuradas, se ha tomado como referencia la estrategia presentada en [Ge et al., 2024], donde también se aborda la predicción de la temperatura superficial del mar. En dicho enfoque, la densificación de nodos se aplica en aquellas regiones del dominio donde el gradiente de la variable objetivo alcanza valores elevados. Para ello, se establecen dos umbrales de sensibilidad, fijados en 0.1 y 0.2, que determinan el nivel de refinamiento aplicado a la malla.

En las zonas donde el gradiente se encuentra por debajo de 0.1, se conserva únicamente el nodo original, al tratarse de áreas con cambios suaves. Cuando el gradiente se sitúa entre 0.1 y 0.2, se incorporan ocho puntos adicionales en torno al nodo original, con una resolución intermedia de $0,15^\circ$, lo que permite capturar transiciones moderadas. Finalmente, en aquellas regiones donde el gradiente supera el umbral de 0.2, se añaden ocho puntos más, esta vez con una resolución más fina de $0,05^\circ$, incrementando así la densidad de la malla en zonas caracterizadas por alta variabilidad espacial.

De esta malla se destaca su estructura cuadricular, similar a la utilizada en el modelo SeaCast. Sin embargo, al analizar la malla resultante, se ha considerado oportuno explorar alternativas de densificación basadas en estructuras menos rígidas, que permitan una representación espacial más flexible y adaptativa a las características físicas de la región de estudio.

6.3.2. Mallas adaptativas no estructuradas

En cuanto a la generación de mallas adaptativas no estructuradas, se han generado a partir de la máscara de batimetría de la Figura 4.5 dos funciones de densidad, con el objetivo de generar pesos que favorezcan la generación de nodos en las zonas costeras,

donde la predicción de la SST es más compleja.

Para llevar a cabo las funciones de densificación, se normalizaron previamente los valores de batimetría al rango $[0, 1]$ mediante la técnica de normalización min-max. Este proceso permite escalar los datos conservando su distribución relativa, facilitando así una integración más homogénea en los modelos posteriores y mejorando la estabilidad numérica de las operaciones involucradas.

$$\tilde{B}(x, y) = \frac{B(x, y) - B_{\min}}{B_{\max} - B_{\min}}. \quad (6.2)$$

En la primera función de densificación, se ha optado por construir una distribución de probabilidad básica que favorece las zonas oceánicas poco profundas. La idea principal consiste en asignar una densidad inversamente proporcional a la raíz cuadrada de la batimetría, de modo que las celdas con valores bajos (es decir, zonas cercanas a la costa o más someras) obtengan una mayor importancia relativa. Por otra parte, se ha añadido un pequeño valor de corrección $\varepsilon > 0$ con el objetivo de evitar valores excesivamente bajos en la división. El proceso completo se detalla en las siguientes expresiones:

$$\varepsilon = 10^{-3}, \quad (6.3)$$

$$D_{i,j} = \begin{cases} \frac{1}{\sqrt{B_{i,j}} + \varepsilon} & \text{si } B_{i,j} > 0, \\ 0 & \text{si } B_{i,j} = 0, \end{cases} \quad (6.4)$$

$$P_{i,j} = \frac{D_{i,j}}{\sum_{i,j} D_{i,j}}. \quad (6.5)$$

En cuanto a la otra distribución de probabilidad desarrollada, se ha buscado una distribución tal que no tenga un cambio excesivamente brusco en la asignación de probabilidades, intentando evitar posibles excesos de nodos en las zonas costeras sin representar adecuadamente el resto del océano. Por ello, aprovechando que el rango de valores de la batimetría se estaba usando en el rango 0-1, se ha optado por modificar ligeramente la fórmula del sigmoide con este propósito.

Se ha optado por usar la función sigmoide debido a que su fórmula es relativamente sencilla, permitiéndonos inferir ligeramente en ella, haciendo que la curvatura propia de esta función resultase atractiva para nuestro objetivo.

La formulación de esta distribución es la siguiente:

Dada una matriz de batimetría $B(x, y)$, se define la distribución de probabilidad $P(x, y)$ sobre las regiones oceánicas $B(x, y) > 0$ como:

$$P(x, y) = \begin{cases} \frac{1}{1 + \exp\left(\frac{B(x, y) - B_p}{\text{IQR}}\right)} & \text{si } B(x, y) > 0. \\ 0 & \text{si } B(x, y) = 0. \end{cases} \quad (6.6)$$

Donde B_p representa el valor correspondiente al percentil p de la batimetría en aquellas zonas oceánicas donde se cumple que $B(x, y) > 0$, y $\text{IQR} = Q_3 - Q_1$ denota el rango intercuartílico, es decir, la diferencia entre el tercer y el primer cuartil de la distribución batimétrica.

Finalmente, se normaliza la matriz resultante para que:

$$\sum_{x,y} P(x, y) = 1.$$

Dado que esto no fue suficiente para cumplir con el requisito suavizante previamente establecido, se optó por construir una distribución mixta entre dos funciones sigmoideas utilizando la fórmula anterior, variando el percentil de inflexión en cada caso.

La primera componente de esta mezcla, denotada como $P_1(x, y)$, tiene su punto de inflexión en el percentil $p_1 = 25$, lo que permite dar mayor peso a las zonas costeras. Por otro lado, la segunda componente, $P_2(x, y)$, se define con un punto de inflexión en el percentil $p_2 = 50$, lo que proporciona una distribución más neutra y equilibrada sobre el dominio oceánico.

se construye una distribución mixta $P_{\text{mix}}(x, y)$ como combinación convexa:

$$P_{\text{mix}}(x, y) = \alpha \cdot P_1(x, y) + (1 - \alpha) \cdot P_2(x, y), \quad (6.7)$$

donde $\alpha \in [0, 1]$ controla el peso asignado a cada distribución, donde, en nuestro caso, hemos optado por un valor de $\alpha = 0,7$. La mezcla final se normaliza para que:

$$\sum_{x,y} P_{\text{mix}}(x, y) = 1.$$

Con esta mezcla de distribuciones, buscamos que la transición entre zonas costeras y zonas de mar abierto no sea tan brusca, donde, como podemos ver en la Figura 6.5, con esta fórmula hemos podido generar un descenso en las probabilidades más controlado, permitiendo que exista una probabilidad mayor de generación de nodos en toda la zona de estudio.

Estas distribuciones, posteriormente, se pueden utilizar de dos maneras diferentes, teniendo la posibilidad de usar directamente estas fórmulas para generar las coordenadas de los nodos, donde se busca llevar al límite estas estrategias de densificación para la predicción de la variable objetivo al no existir restricción alguna en la generación de dichos nodos, salvo las comunes en nuestro proceso general, siendo estas representadas en las Figuras 6.6a, 6.6b. Por otro lado, estas también se pueden utilizar para establecer una ligera variación

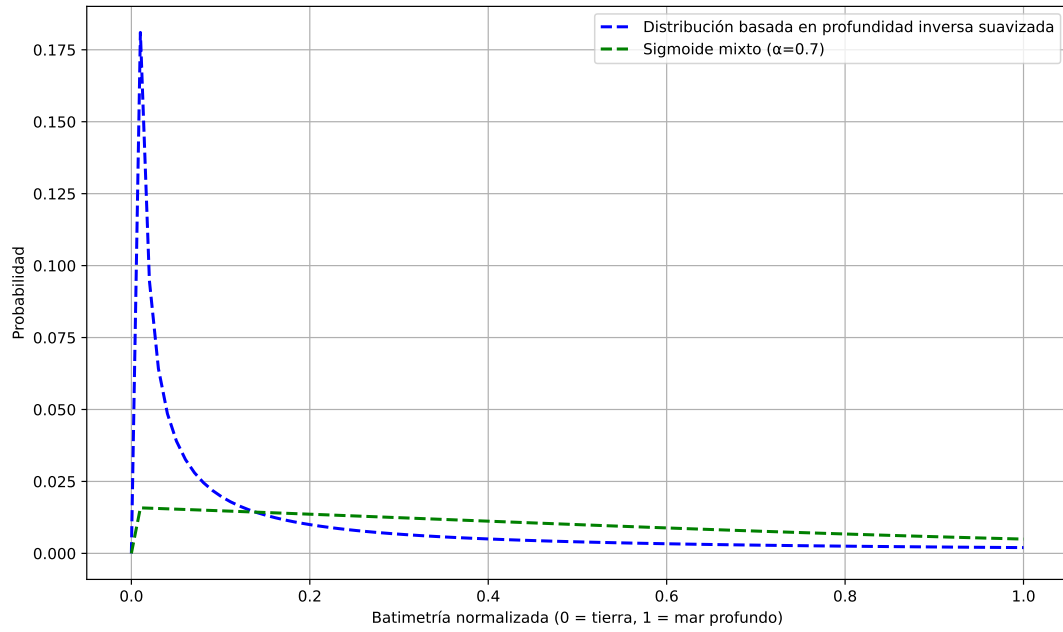


Figura 6.5: Comparación de la evolución de las probabilidades asignadas según valor batimétrico normalizado de las funciones de densidad propuestas.

en el algoritmo de muestreo del punto más lejano, siendo esta modificación la que se puede encontrar en el Algoritmo 3, donde se multiplica el factor determinante de selección de un nodo(en el caso del muestreo por el punto más lejano, la distancia euclídea) por los pesos generados en las funciones de probabilidad anteriores, donde los resultados obtenidos se pueden ver en las Figuras 6.6c, 6.6d.

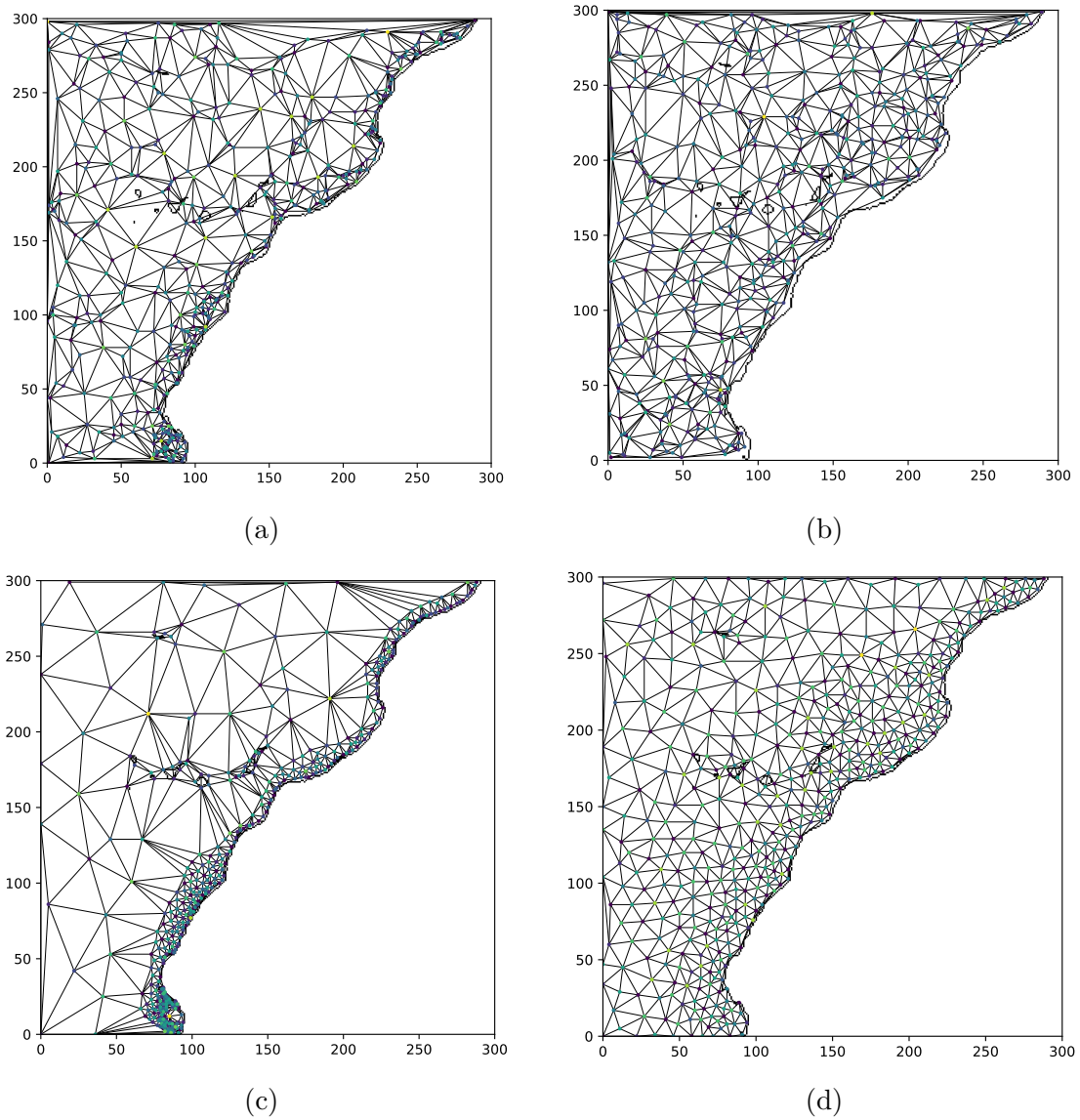


Figura 6.6: Mallas adaptativas generadas donde: (a) malla generada según las probabilidades obtenidas de la distribución basada en profundidad inversa suavizada, (b) malla generada según las probabilidades obtenidas de la distribución sigmoide mixto ($\alpha = 0,7$), (c) malla generada según muestreo del punto más lejano ponderado por la distribución basada en profundidad inversa suavizada, (d) malla generada según muestreo del punto más lejano ponderado por la distribución sigmoide mixto ($\alpha = 0,7$)

Algorithm 3 Muestreo por el punto más lejano ponderado (Weighted Farthest Point Sampling)

Require: Conjunto de puntos candidatos $P = \{p_1, p_2, \dots, p_N\} \subset \mathbb{R}^2$

Número de puntos a seleccionar $k \in \mathbb{N}$

Vector de pesos $w = (w_1, w_2, \dots, w_N)$, con $w_i \geq 0$ y $\sum_i w_i = 1$

Ensure: Subconjunto $S \subseteq P$ de hasta k puntos seleccionados por distancia ponderada

```

1: if  $k \leq 0$  o  $N = 0$  then
2:   return conjunto vacío
3: end if
4: Seleccionar un punto inicial  $p_i \in P$  con probabilidad proporcional a  $w_i$ 
5: Inicializar el conjunto  $S \leftarrow \{p_i\}$ 
6: Para cada  $p_j \in P$ , calcular la distancia euclídea  $d_j \leftarrow \|p_j - p_i\|$ 
7: for  $t = 2$  hasta  $\min(k, N)$  do
8:   Calcular las puntuaciones ponderadas:  $s_j \leftarrow d_j \cdot w_j$ 
9:   Seleccionar el punto  $p^* \leftarrow \arg \max_{p_j \in P} s_j$ 
10:  Añadir  $p^*$  al conjunto  $S$ 
11:  Para cada  $p_j \in P$ , actualizar su distancia:
       $d_j \leftarrow \min(d_j, \|p_j - p^*\|)$ 
12: end for
13: return conjunto  $S$ 

```

Capítulo 7

Configuración de experimentos y resultados

Con el objetivo de evaluar la influencia de distintas mallas espaciales en el rendimiento de modelos de aprendizaje profundo para la predicción de variables oceanográficas, se ha diseñado un conjunto de experimentos controlados que permitirán comparar de forma rigurosa las distintas representaciones topológicas empleadas. Esta sección describe detalladamente la configuración experimental utilizada, incluyendo la fijación de los hiperparámetros, la elección de la función de pérdida y el optimizador, así como los criterios que aseguran la validez de la comparación entre mallas. Finalmente, se presentan los resultados obtenidos y se discuten sus implicaciones en el contexto del modelado oceánico mediante redes neuronales.

Una vez descritas las mallas generadas en este trabajo, se procede a evaluar su rendimiento mediante diversos entrenamientos. Para ello, se mantendrán constantes tanto los parámetros de entrenamiento como el número de nodos, con el objetivo de obtener resultados que permitan determinar la influencia de la morfología de la malla en redes neuronales para la predicción de variables oceanográficas.

7.1. Hiperparámetros y estrategia de entrenamiento

Antes de abordar el análisis detallado de los experimentos realizados, es fundamental establecer un marco común que asegure la comparabilidad de los resultados. Esta sección describe los elementos que se han mantenido constantes a lo largo de todas las pruebas, incluyendo los hiperparámetros seleccionados, la función de pérdida empleada y el algoritmo de optimización utilizado. Al fijar estas configuraciones, se garantiza que las diferencias observadas en el rendimiento de los modelos puedan atribuirse exclusivamente a las características estructurales de las mallas sobre las que operan, y no a variaciones en la metodología de entrenamiento. A continuación, se detallan los valores adoptados y las justificaciones detrás de cada elección.

En cuanto a los hiperparámetros utilizados, se ha fijado un número de 100 épocas de entrenamiento, omitiendo los pasos autorregresivos debido al elevado coste computacional que

implican. La tasa de aprendizaje inicial se ha establecido en 0,001, incorporando un esquema de decaimiento cosenoidal para ajustarla dinámicamente a lo largo del entrenamiento. Esta técnica, conocida como *cosine annealing*, permite disminuir progresivamente la tasa de aprendizaje para favorecer la convergencia del modelo, emulando el comportamiento de una función coseno decreciente. Su formulación se presenta en la Ecuación 7.1:

$$lr(t) = lr_{\min} + \frac{1}{2}(lr_{\max} - lr_{\min}) \left(1 + \cos \left(\frac{t}{T} \pi \right) \right). \quad (7.1)$$

7.1.1. Función de pérdida

Para cuantificar el error de las predicciones, se ha empleado el error cuadrático medio ponderado (WMSE, por sus siglas en inglés), una extensión del MSE tradicional que incorpora dos factores de corrección: una máscara espacial m_i que define las regiones válidas del dominio, y una ponderación w_i/σ_i^2 que considera la relevancia e incertidumbre asociadas a cada punto de la malla. En este trabajo, la máscara de pesos utilizada corresponde a la grilla generada en la Sección 4. La expresión general del WMSE se muestra a continuación:

$$\text{WMSE} = \frac{\sum_{i=1}^N m_i w_i \frac{(\hat{y}_i - y_i)^2}{\sigma_i^2}}{\sum_{i=1}^N m_i w_i}. \quad (7.2)$$

7.1.2. Optimizador

El algoritmo de optimización empleado ha sido AdamW, una variante del optimizador Adam que introduce un término explícito de regularización mediante decaimiento de pesos (*weight decay*). Esta modificación mejora la generalización del modelo al penalizar directamente la magnitud de los parámetros. En este caso, se ha utilizado un valor de decaimiento $\lambda = 0,1$, junto con los coeficientes $\beta_1 = 0,9$ y $\beta_2 = 0,95$ para el cálculo de los momentos del gradiente. Las ecuaciones de actualización de AdamW se detallan en la Ecuación 7.3:

$$\begin{aligned} g_t &= \nabla_w \mathcal{L}(w_t), & (\text{gradiente del parámetro}) \\ m_t &= \beta_1 m_{t-1} + (1 - \beta_1) g_t, & (\text{promedio móvil del primer momento}) \\ v_t &= \beta_2 v_{t-1} + (1 - \beta_2) g_t^2, & (\text{promedio móvil del segundo momento}) \\ \hat{m}_t &= \frac{m_t}{1 - \beta_1^t}, & (\text{corrección de sesgo del primer momento}) \\ \hat{v}_t &= \frac{v_t}{1 - \beta_2^t}, & (\text{corrección de sesgo del segundo momento}) \\ w_{t+1} &= w_t - \eta \left(\frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon} + \lambda w_t \right). & (\text{actualización con decaimiento de pesos}) \end{aligned} \quad (7.3)$$

Una vez establecidos los parámetros de entrenamiento, se presentan a continuación los distintos experimentos diseñados para analizar la influencia de las mallas en el rendimiento de las redes neuronales. Cada experimento se ha llevado a cabo manteniendo inalteradas las condiciones de entrenamiento, de forma que cualquier diferencia en los resultados puede atribuirse directamente a la estructura geométrica y topológica de la malla utilizada. En las siguientes secciones se analizan los resultados obtenidos, incluyendo métricas cuantitativas y visualizaciones cualitativas del comportamiento del modelo bajo cada configuración.

7.2. Evaluación de la influencia estructural de la malla

El objetivo de este experimento es analizar el impacto que tiene la estructura de la malla sobre el rendimiento de un modelo de predicción de variables oceanográficas. Para ello, se comparan tres configuraciones distintas: una malla regular uniforme, utilizada por defecto en SeaCast, frente a dos mallas menos estructuradas diseñadas en este trabajo —una con disposición aleatoria y otra basada en una distribución proporcional a la profundidad inversa suavizada, detalladas en el Capítulo 6.

Con el fin de garantizar una comparación justa entre las distintas configuraciones, se ha mantenido constante el número total de nodos y niveles verticales en todas las mallas. Así, las diferencias observadas en el rendimiento pueden atribuirse directamente a las propiedades topológicas y geométricas de cada malla, como la distribución espacial de los nodos y el número de aristas, que varían de forma inherente a cada diseño.

Además, se ha fijado el número de vecinos para la interconexión entre niveles en 16, tal y como establece la configuración por defecto de SeaCast. No obstante, se ha modificado el esquema de conexión entre la malla y la grilla global, de modo que cada nodo de la grilla se conecta con los tres nodos más cercanos de la malla de mayor resolución. Esta elección se ha aplicado de manera simétrica tanto en la fase de codificación como en la de decodificación.

Las características estructurales de las mallas empleadas se resumen en la Tabla 7.1, donde se hace referencia a cada configuración mediante las siguientes denominaciones: **U-malla** (uniforme), **R-malla** (aleatoria) y **B-malla** (basada en profundidad inversa suavizada). Estas mallas son las correspondientes a las de la Figura 7.1. A su vez, se han reflejado los tiempos de entrenamiento resultantes de cada malla, siendo estos los encontrados en la Tabla 7.2.

Tabla 7.1: Resumen estructural de las mallas utilizadas por tipo y nivel jerárquico. Se indica el número total de nodos y aristas en cada uno de los tres niveles para los tres tipos de malla considerados: **U-malla** (uniforme), **R-malla** (aleatoria) y **B-malla** (basada en profundidad).

Tamaño de malla según modelo	Nivel 1			Nivel 2			Nivel 3		
	U-malla	R-malla	B-malla	U-malla	R-malla	B-malla	U-malla	R-malla	B-malla
Nodos	3568	3568	3568	394	394	394	45	45	45
Aristas	27578	21216	21192	2848	2298	2248	268	242	208

Tabla 7.2: Comparativa del tiempo de entrenamiento (en horas y minutos) requerido por cada configuración de malla, así como la mejora porcentual obtenida respecto a la U-malla. Estos resultados reflejan la eficiencia computacional alcanzada por las mallas alternativas sin comprometer la calidad predictiva.

Modelo	Tiempo de entrenamiento	Mejora respecto a U-malla (%)
U-malla	41 h 54 min	0.00 %
R-malla	40 h 25 min	3.57 %
B-malla	39 h 54 min	4.80 %

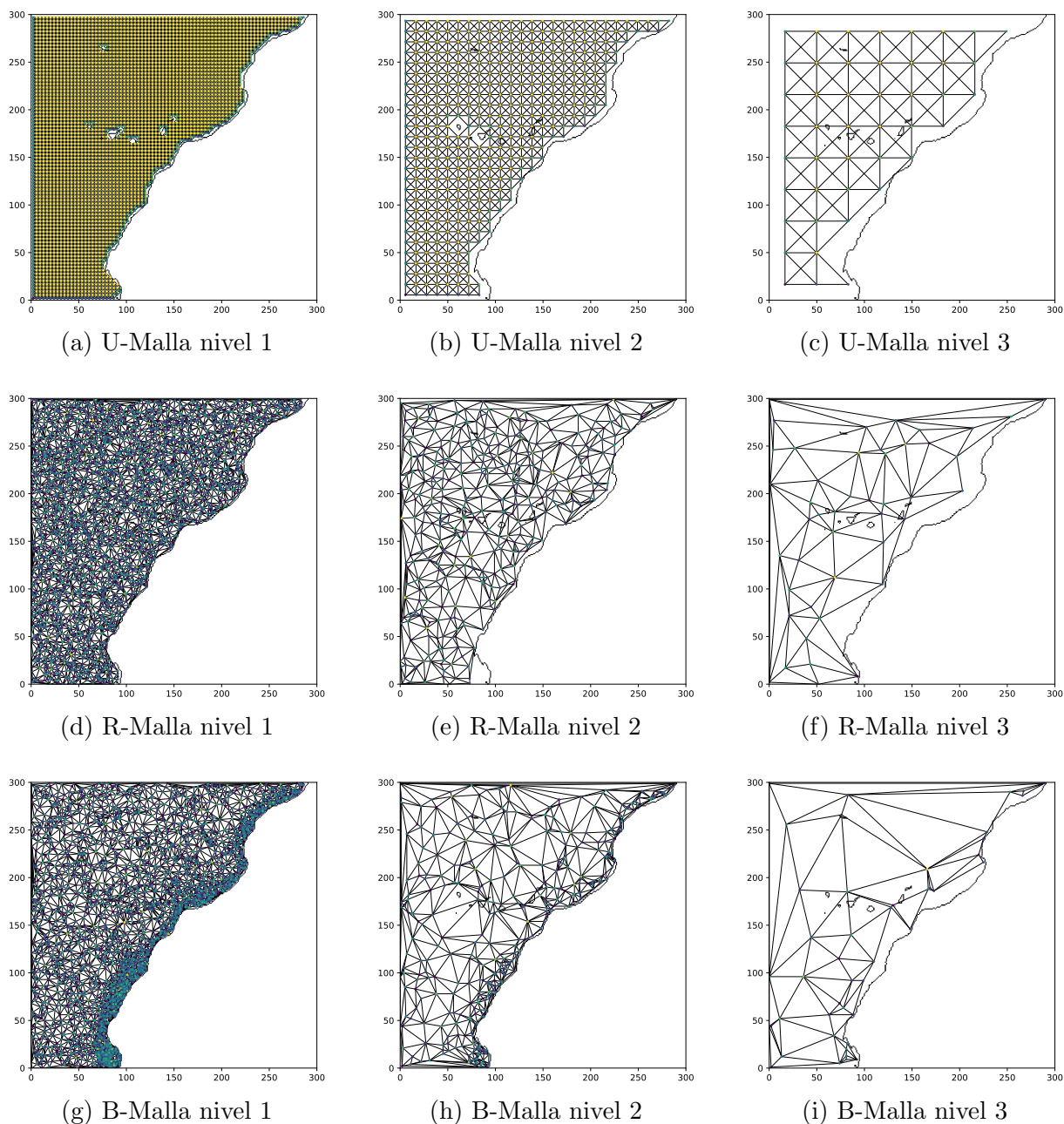
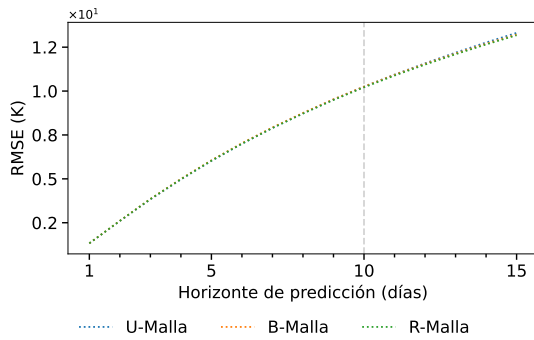


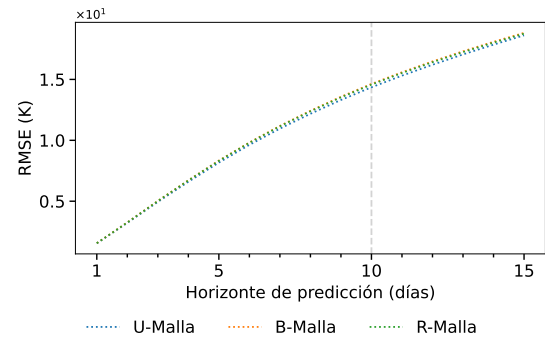
Figura 7.1: Visualización comparativa de las tres mallas jerárquicas generadas (U-malla, R-malla y B-malla) en sus respectivos tres niveles de resolución. Se muestran las estructuras nodales y conexiones para cada configuración evaluada en el experimento sobre la influencia estructural de la malla.

Resultados obtenidos del experimento sobre la evaluación de la influencia estructural de la malla

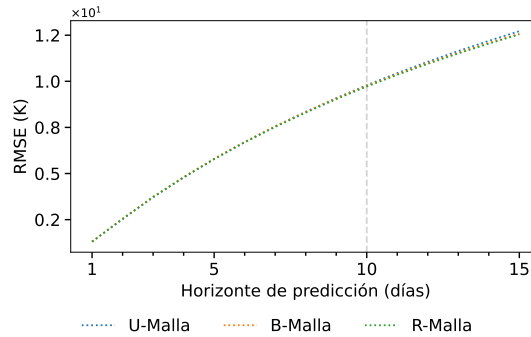
En relación con los resultados obtenidos en las predicciones, la Figura 7.2 muestra la evolución de la raíz cuadrada del error cuadrático medio (RMSE) a lo largo de los 15 días de pronóstico definidos. Los valores representados corresponden a la media de los errores cometidos durante el año seleccionado para la evaluación de los modelos. Tal como se observa, las diferencias entre las mallas son prácticamente inexistentes a lo largo del horizonte temporal. Este resultado resulta particularmente llamativo, ya que se esperaba un rendimiento sensiblemente inferior por parte de la R-malla debido a su construcción aleatoria y poco estructurada. Esta aparente igualdad en los errores globales plantea la necesidad de analizar si también se mantiene en el espacio o si, por el contrario, existen diferencias locales relevantes en determinadas zonas del dominio de estudio.



(a) RMSE en toda la región de estudio.



(b) RMSE en la costa africana.



(c) RMSE en el mar abierto.

Figura 7.2: Evolución del RMSE medio a lo largo del horizonte de predicción de 15 días para cada una de las configuraciones de malla consideradas (U-malla, R-malla y B-malla). Se presentan los errores promedio en tres dominios distintos: (a) toda la región de estudio, (b) zona costera africana y (c) mar abierto. Estos resultados permiten comparar el rendimiento global y local de cada tipo de malla.

Con el fin de explorar esta posibilidad, la Figura 7.3 presenta la distribución espacial del RMSE en toda la región de análisis. Nuevamente, las diferencias entre las mallas continúan siendo mínimas en términos generales. No obstante, al observar en detalle el último día del pronóstico, se aprecia que la U-malla presenta ciertos artefactos y picos localizados de error en zonas costeras, especialmente a lo largo del litoral africano. Este comportamiento sugiere que una malla estructurada y rígida podría no adaptarse de forma óptima a regiones con geometrías irregulares, como es el caso de los litorales.

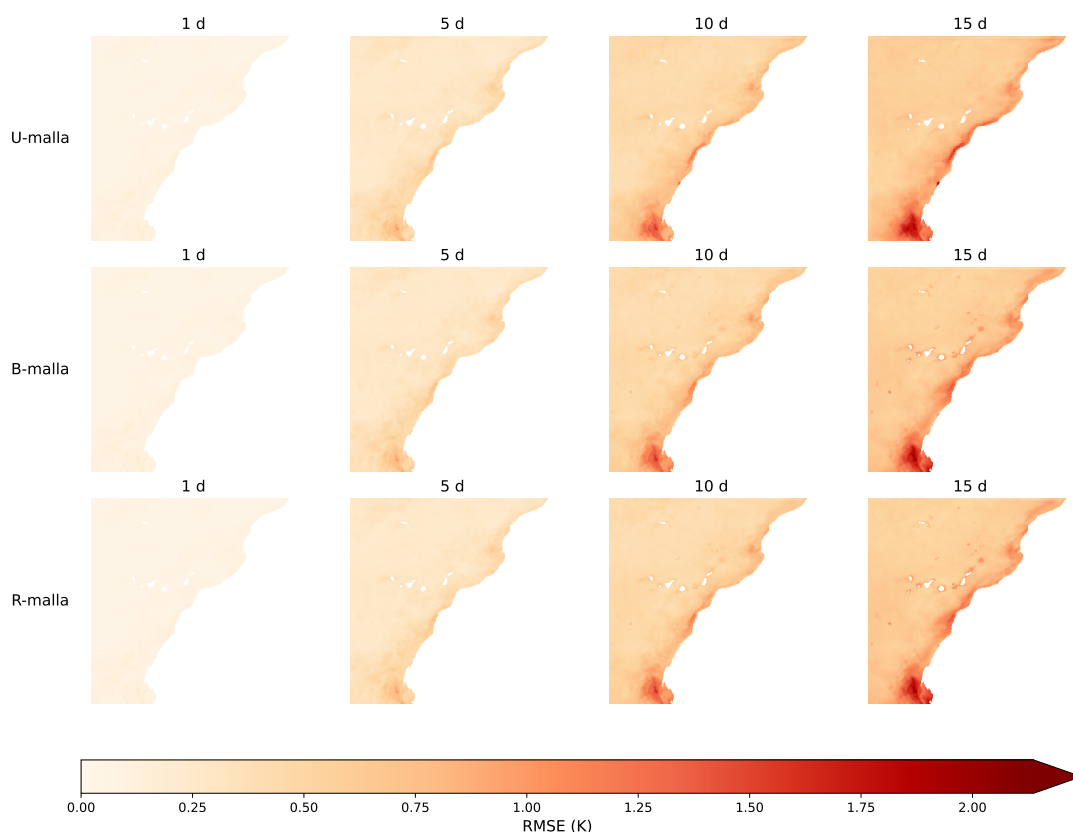


Figura 7.3: Distribución espacial del RMSE en distintos horizontes de predicción (1, 5, 10 y 15 días) para cada tipo de malla evaluada. Se observa cómo la U-malla presenta errores más localizados en zonas costeras, mientras que la B-malla y la R-malla ofrecen patrones más homogéneos.

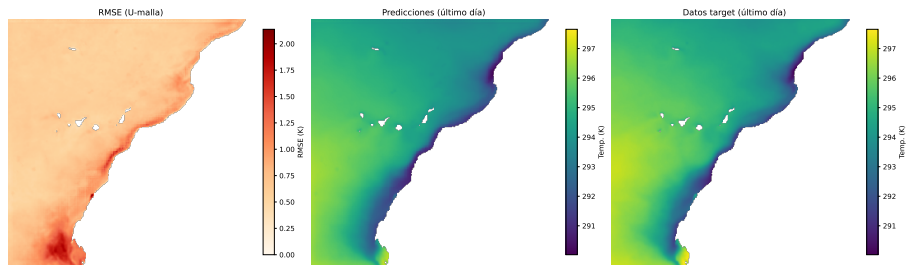
En contraste, las mallas no estructuradas —como la R-malla y la B-malla— muestran un mejor ajuste a estas regiones, lo que podría atribuirse a su capacidad para evitar la pérdida de conectividad derivada de la eliminación de nodos ubicados en tierra. En el caso de la U-malla, dicha eliminación puede provocar deformaciones locales en la malla, especialmente en los bordes de los cuadrados que quedan incompletos, lo que impacta negativamente en la fase de decodificación del modelo.

Para profundizar en este análisis, las Figuras 7.4a, 7.4b y 7.4c muestran una comparación entre los valores predichos, los valores objetivo y los respectivos RMSE para el último día del pronóstico. Aunque las diferencias visuales entre las mallas son sutiles, tanto la R-malla como la B-malla presentan campos de error y de predicción más suaves, con una menor aparición de artefactos localizados. Este comportamiento refuerza la hipótesis de que las mallas no estructuradas pueden ofrecer ventajas en la representación de dominios costeros complejos, y motiva la formulación de nuevas hipótesis que serán exploradas en los experimentos posteriores.

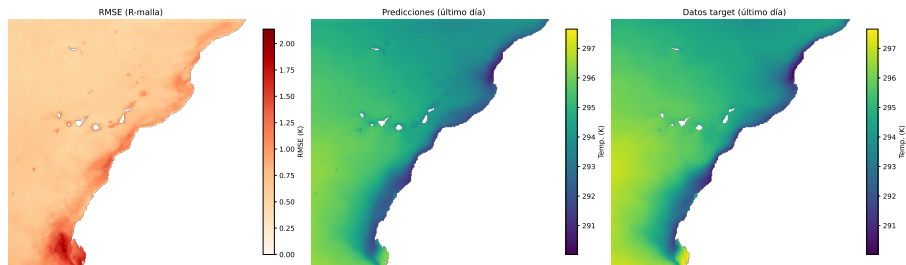
Por otro lado, cabe destacar ligeras mejoras en los tiempos de entrenamiento obtenidos por los modelos basados en mallas no estructuradas, tal como se recoge en la Tabla 7.2. A pesar de alcanzar resultados casi idénticos en cuanto a precisión, estos modelos han requerido menos recursos computacionales que el modelo basado en la malla uniforme. Este hallazgo, sumado a la equivalencia en el rendimiento predictivo, sugiere que la cantidad de nodos

generada por defecto en la configuración original de SeaCast podría ser excesiva para el caso concreto de estudio, reduciendo el impacto real de las diferencias morfológicas introducidas entre mallas.

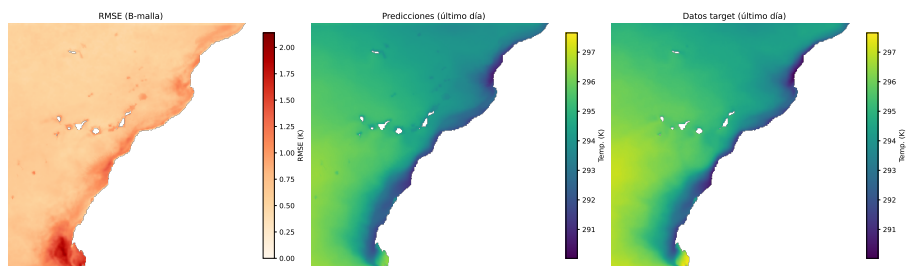
Este resultado motivó la realización de un segundo experimento, en el que se redujo progresivamente el número de nodos de las mallas con el objetivo de analizar cómo afecta esta disminución tanto al rendimiento predictivo como al coste computacional. La finalidad de dicho experimento es identificar un equilibrio óptimo entre la eficiencia y la precisión del modelo, evaluando si una malla menos densa, pero estructuralmente adecuada, puede mantener un rendimiento comparable al de configuraciones más complejas.



(a) U-malla: RMSE espacial, predicción y valor objetivo en el último día de los pronósticos.



(b) R-malla: RMSE espacial, predicción y valor objetivo en el último día de los pronósticos.



(c) B-malla: RMSE espacial, predicción y valor objetivo en el último día de los pronósticos.

Figura 7.4: Comparación detallada entre las tres configuraciones de malla (U-malla, R-malla y B-malla) en el último día de predicción. Se muestran tres mapas por cada caso: el RMSE espacial, los valores predichos por el modelo y los valores objetivos del conjunto de test. Esta visualización permite detectar errores localizados y evaluar la fidelidad espacial de las predicciones.

7.3. Impacto de la reducción de resolución de la malla

A partir de los resultados obtenidos en el experimento anterior, en el que las diferencias de rendimiento entre las distintas mallas resultaron sorprendentemente pequeñas, se plantea la hipótesis de que dicha similitud podría deberse a un exceso de nodos en los distintos niveles de resolución. Alternativamente, esta falta de variabilidad en los errores podría estar relacionada con la limitada cantidad de datos utilizada para el entrenamiento. Con el objetivo de investigar la primera de estas posibilidades, el presente experimento propone una degradación controlada de la malla uniforme (U-malla), reduciendo progresivamente su resolución para identificar un número de nodos adecuado que permita conservar la capacidad predictiva del modelo sin incurrir en un mayor coste computacional innecesario.

Debido a las limitaciones computacionales del entorno de trabajo, el entrenamiento se ha mantenido acotado a un periodo de tres años. No obstante, se ha optado por explorar la reducción de la resolución espacial como estrategia alternativa para optimizar el modelo. Esta medida no solo permite disminuir los recursos requeridos, sino que también facilita el análisis de la sensibilidad del modelo frente a modificaciones estructurales en la malla.

Con este enfoque, se incorpora como objetivo adicional del experimento la mejora de la eficiencia computacional del sistema, sin comprometer significativamente la precisión de las predicciones. Para ello, se ha reducido también el número de aristas presentes tanto en el codificador como en el decodificador, limitando las conexiones salientes a una por nodo en la grilla, así como a una conexión por nodo desde las mallas de mayor resolución en los pasos intermedios de interconexión entre niveles.

Asimismo, se ha investigado el efecto de permitir o restringir el cruce de aristas diagonales en la U-malla, utilizada como base para esta experimentación. Se han evaluado distintas variantes construidas a partir de esta malla de referencia, con el fin de alcanzar una configuración que minimice el número de nodos y conexiones sin deteriorar el rendimiento del modelo.

Las configuraciones analizadas se agrupan en dos categorías principales: aquellas que permiten cruces diagonales entre celdas adyacentes, denominadas **ce**, del *crossing edges*, y aquellas que los restringen, siendo estas denominadas en este caso como **nce**, del inglés *non-crossing edges*), donde se fusionará con la resolución de la malla para identificar claramente cada modelo de malla. Por otra parte, cabe destacar que los cambios de resoluciones de las mallas se harán principalmente con el enfoque que evita cruces diagonales de aristas. Las resoluciones utilizadas en cada nivel de malla, así como las diferencias topológicas entre las configuraciones, se recogen en la Tabla 7.3 y se ilustran visualmente en las Figuras 7.5 y 7.6 y 7.7. Dado el volumen visual y la estructura multinivel de las mallas representadas, las Figuras 7.5 y 7.6 y 7.7 se han agrupado al final de esta sección, antes de los resultados. Esta disposición permite visualizar de forma continua las distintas configuraciones analizadas sin interrumpir el desarrollo argumental del texto, donde se ha tenido que dividir en varias figuras para conseguir un tamaño mínimo para su visualización, favoreciendo así una lectura más fluida y una comparación directa entre variantes.

Tabla 7.3: Características de las mallas por nivel y tipo en el experimento sobre el impacto de la reducción de resolución de la malla.

Malla	Nivel 1		Nivel 2		Nivel 3	
	Nodos	Aristas	Nodos	Aristas	Nodos	Aristas
ce_81_27_9	3568	27578	394	2848	45	268
nce_81_27_9	3568	20944	394	2204	45	212
nce_63_21_7	2158	12598	243	1310	27	122
nce_45_15_5	1103	6336	122	642	14	54
nce_27_9_3	394	2204	45	212	5	14
nce_12_6_3	80	396	20	82	5	14
nce_5_4_3	14	54	9	30	5	14
nce_5_3	14	54	5	14	—	—

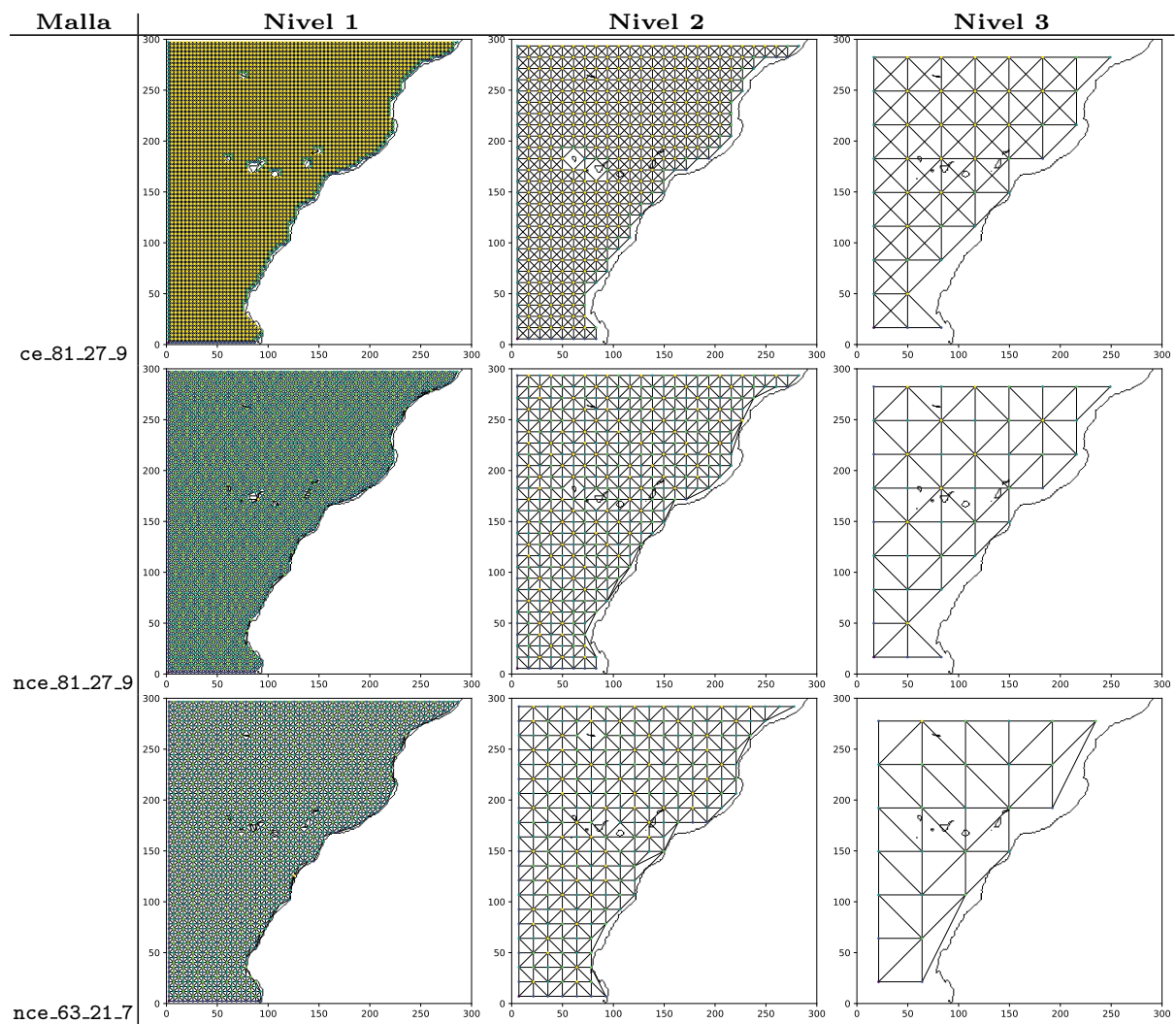


Figura 7.5: Comparativa visual (parte 1) de las mallas con mayor resolución utilizadas en el experimento sobre la reducción progresiva de la malla. Cada fila muestra una configuración concreta y cada columna representa un nivel jerárquico (del 1 al 3).

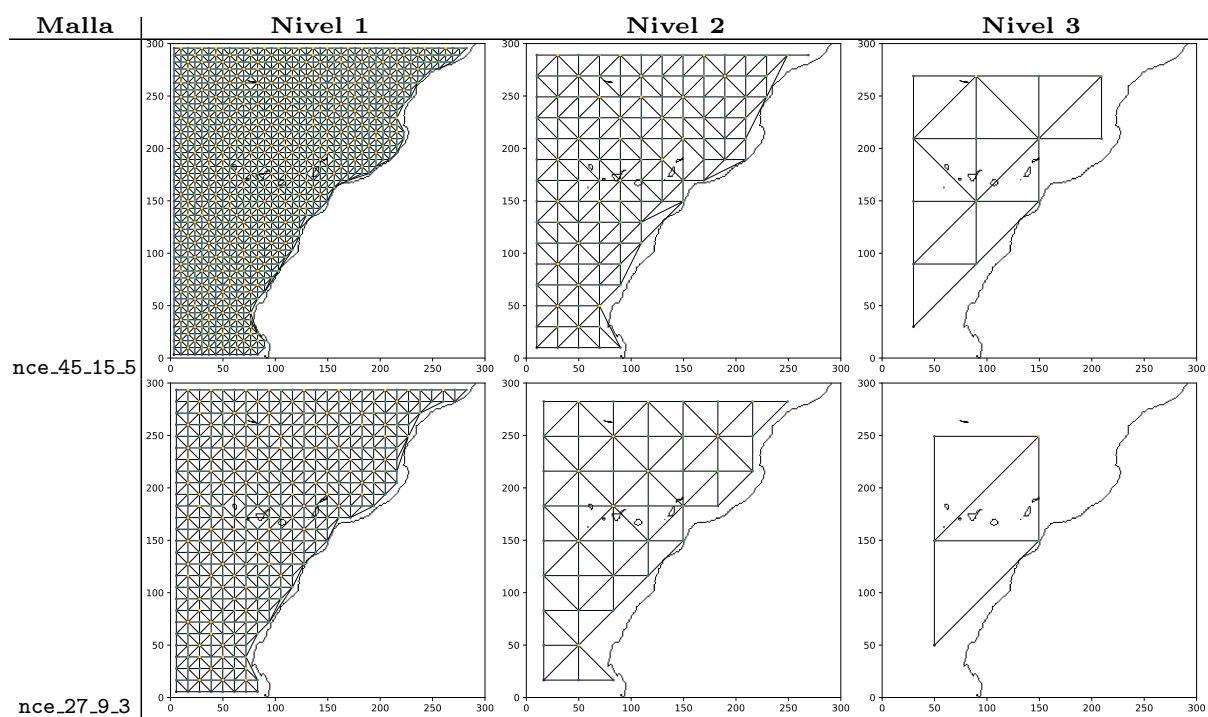


Figura 7.6: Comparativa visual (parte 2) de las mallas con mayor resolución utilizadas en el experimento sobre la reducción progresiva de la malla. Esta segunda parte incluye las configuraciones de menor tamaño evaluadas.

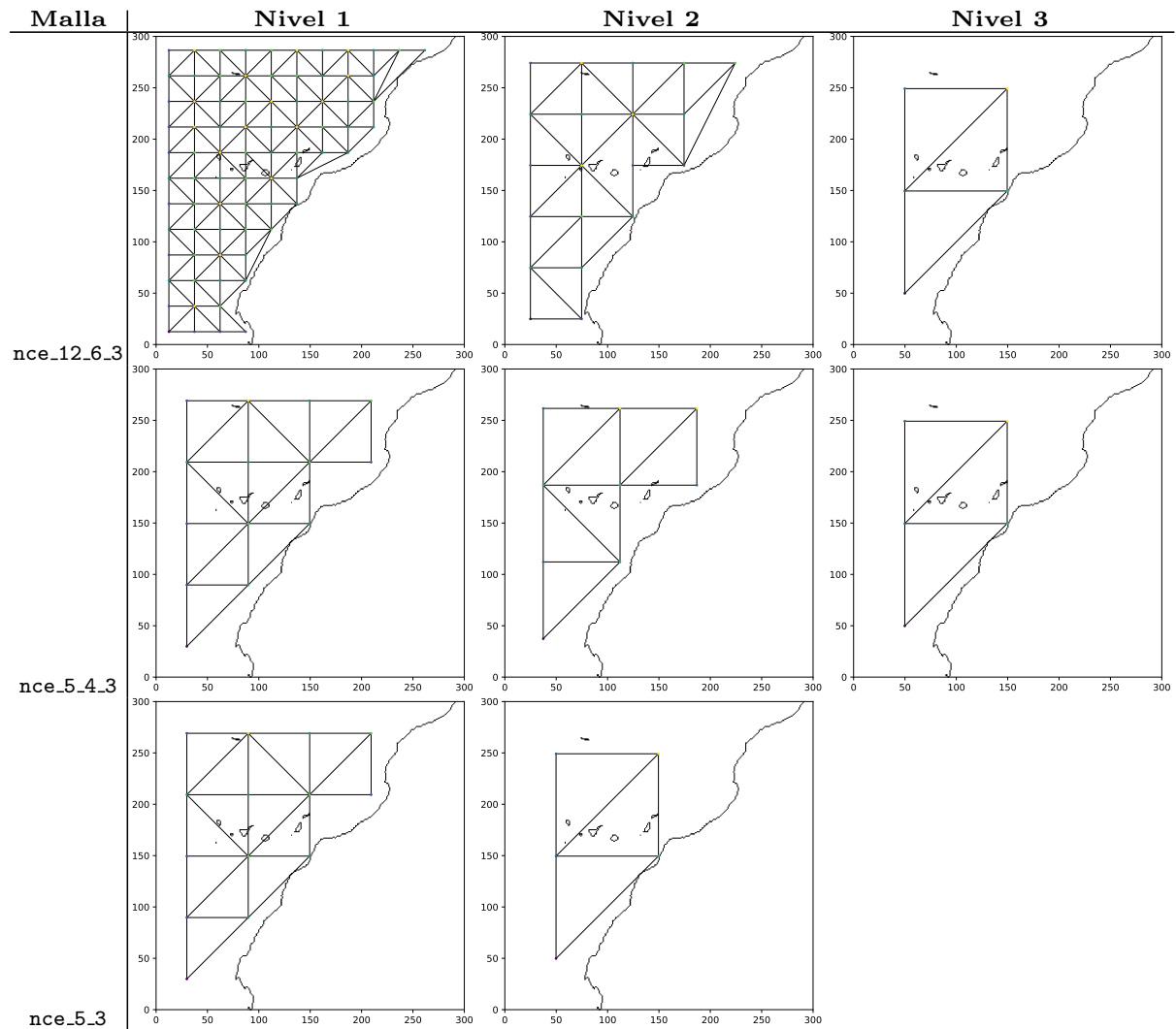


Figura 7.7: Comparativa visual de las mallas con resoluciones degradadas utilizadas en el mismo experimento. Esta figura complementa la anterior, mostrando configuraciones más compactas. Cada fila representa una malla específica y cada columna, su correspondiente nivel de resolución.

Resultados obtenidos de la evaluación del impacto de la reducción de resolución de la malla

En primer lugar, se ha llevado a cabo un análisis comparativo para evaluar el impacto de permitir o restringir el cruce de aristas diagonales en la U-malla. Las Figuras 7.8 y 7.9 ilustran los resultados obtenidos. Aunque las diferencias globales no son especialmente pronunciadas, la Figura 7.8 muestra que la configuración sin aristas diagonales cruzadas (*nce*) presenta un rendimiento ligeramente superior en términos de RMSE acumulado durante los 15 días de pronóstico. Asimismo, la representación espacial del error en la Figura 7.9 evidencia una reducción en la aparición de artefactos en las zonas costeras, así como una predicción algo más homogénea en el resto del dominio. Estos resultados sugieren que eliminar el cruce diagonal entre celdas puede contribuir a una malla más regular y estable en regiones con geometría costera compleja, mejorando levemente la capacidad de generalización del modelo, donde a partir de estos resultados podemos inferir que el cruce de aristas en esta malla estaba añadiendo más conexiones de las necesarias para la representación de la temperatura superficial del mar en la zona de estudio, permitiendo, no solo conseguir una ligera mejora en la predicción del modelo, sino también una reducción del coste computacional necesario para el entrenamiento del modelo.

En segundo lugar, se ha llevado a cabo una evaluación detallada del impacto que tiene la resolución espacial de la malla sobre el rendimiento del modelo. Para ello, se han comparado varias configuraciones progresivamente degradadas, conservando la estructura general de la U-malla sin aristas cruzadas, pero con resoluciones descendentes en cada uno de los niveles jerárquicos. El análisis se ha centrado en el último día del pronóstico, dado que este suele concentrar los mayores errores acumulados y, por tanto, maximizar las diferencias entre configuraciones. La Figura 7.10 muestra la evolución del RMSE para cada resolución evaluada, tanto a escala global como desglosada por regiones de interés: el borde costero africano, caracterizado por el fenómeno de *upwelling*, y el mar abierto.

De la comparación de curvas se desprende que la configuración con resolución 27,9,3 ofrece un rendimiento óptimo antes de que el error comience a estabilizarse en configuraciones más simples. Esta configuración representa un punto de equilibrio entre complejidad estructural y precisión predictiva, ya que mantiene un número razonable de nodos sin incurrir en sobrecostes computacionales innecesarios. Por tanto, se considera esta resolución como óptima, donde podríamos ver diferencias más notables entre las diferentes mallas propuestas.

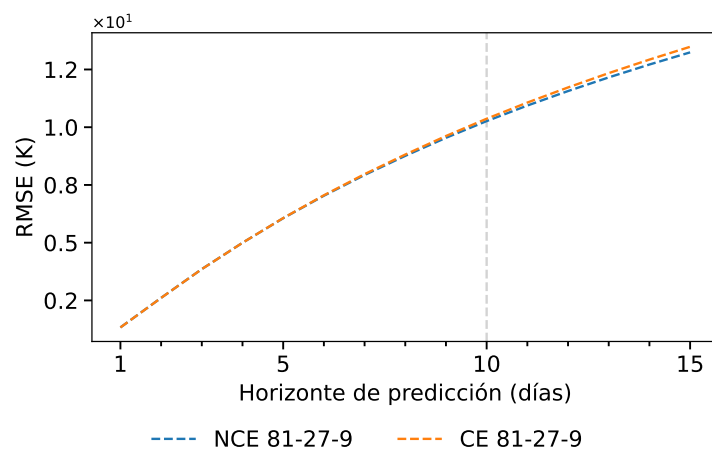
Finalmente, se destaca que las configuraciones de malla utilizadas en este experimento han supuesto una mejora sustancial en los tiempos de entrenamiento. La Tabla 7.4 recoge los tiempos totales, expresados en horas y minutos, junto con los porcentajes de reducción obtenidos respecto a dos referencias: la U-malla original del primer experimento y la configuración *ce_81_27_9*, que incluye aristas cruzadas. Si bien el RMSE global se ha mantenido relativamente estable entre configuraciones, tal y como podemos ver en la curvatura de la Figura 7.10, la disminución del coste computacional ha sido considerable. Este resultado no solo valida la utilidad de la degradación controlada de la malla como estrategia de optimización, sino que también abre nuevas posibilidades para ampliar el periodo de entrenamiento en futuros estudios, sin comprometer la viabilidad operativa del proceso, siendo este punto uno de los más importantes durante el desarrollo de este Trabajo de Fin de Grado, donde la mayor disminución de tiempo de entrenamiento se

ha visto en el cambio del número de aristas que interconectan los diferentes niveles de la malla, donde podemos ver que el número de conexiones totales del modelo puede ser un factor determinante para el rendimiento del modelo y un aspecto a mejorar en futuras líneas de investigación.

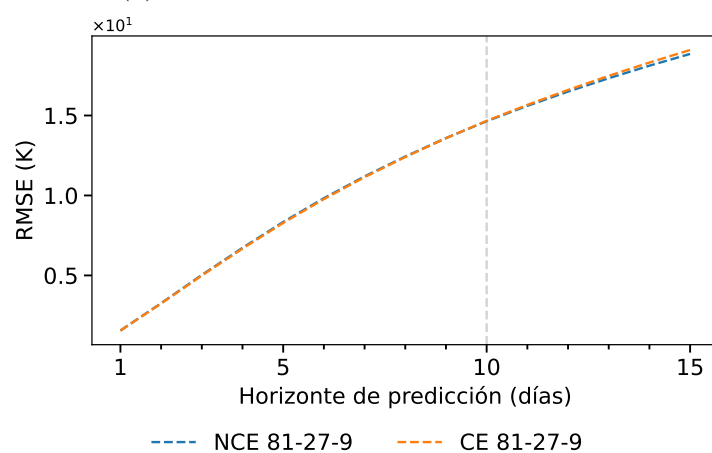
En conjunto, estos hallazgos respaldan la idea de que un diseño eficiente de la malla —que equilibre resolución y conectividad— puede ofrecer ventajas significativas tanto en términos de precisión como de eficiencia computacional, lo que resulta especialmente relevante en contextos donde los recursos disponibles son limitados.

Tabla 7.4: Comparativa de los tiempos de entrenamiento (en horas y minutos) y mejora porcentual obtenida respecto a la U-malla original y la malla con aristas cruzadas `ce_81_27_9`, en el experimento sobre reducción de resolución.

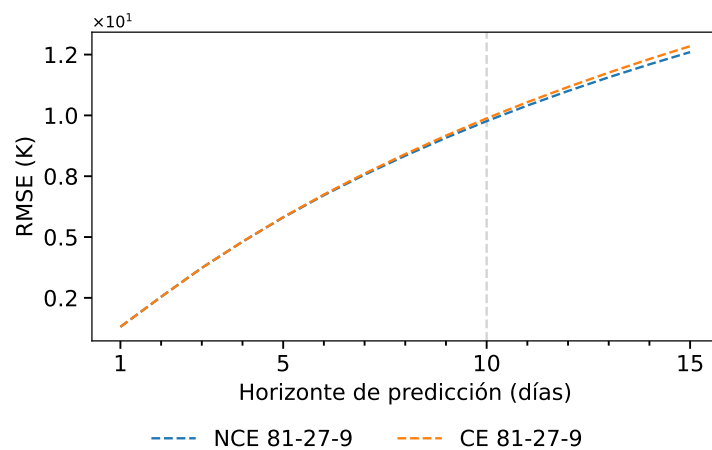
Modelo	Tiempo	Mejora vs U-malla (%)	Mejora vs <code>ce_81_27_9</code> (%)
U-malla	41 h 54 min	0.00 %	—
<code>ce_81_27_9</code>	19 h 48 min	52.72 %	0.00 %
<code>nce_81_27_9</code>	18 h 25 min	56.03 %	6.31 %
<code>nce_63_21_7</code>	15 h 28 min	63.07 %	21.90 %
<code>nce_45_15_5</code>	13 h 31 min	67.73 %	31.72 %
<code>nce_27_9_3</code>	12 h 16 min	70.71 %	38.00 %
<code>nce_12_6_3</code>	12 h 06 min	71.11 %	38.89 %
<code>nce_5_4_3</code>	12 h 11 min	70.95 %	38.38 %
<code>nce_5_3</code>	9 h 42 min	76.83 %	50.99 %



(a) RMSE en toda la región de estudio



(b) RMSE en la costa africana



(c) RMSE en el mar abierto

Figura 7.8: Evolución del error cuadrático medio (RMSE) a lo largo del horizonte de predicción de 15 días para las configuraciones `nce_81_27_9` y `ce_81_27_9`. Se muestran los resultados globales (a), en la región costera (b) y en el mar abierto (c). La comparación permite evaluar el impacto del cruce de aristas diagonales sobre la precisión de las predicciones.

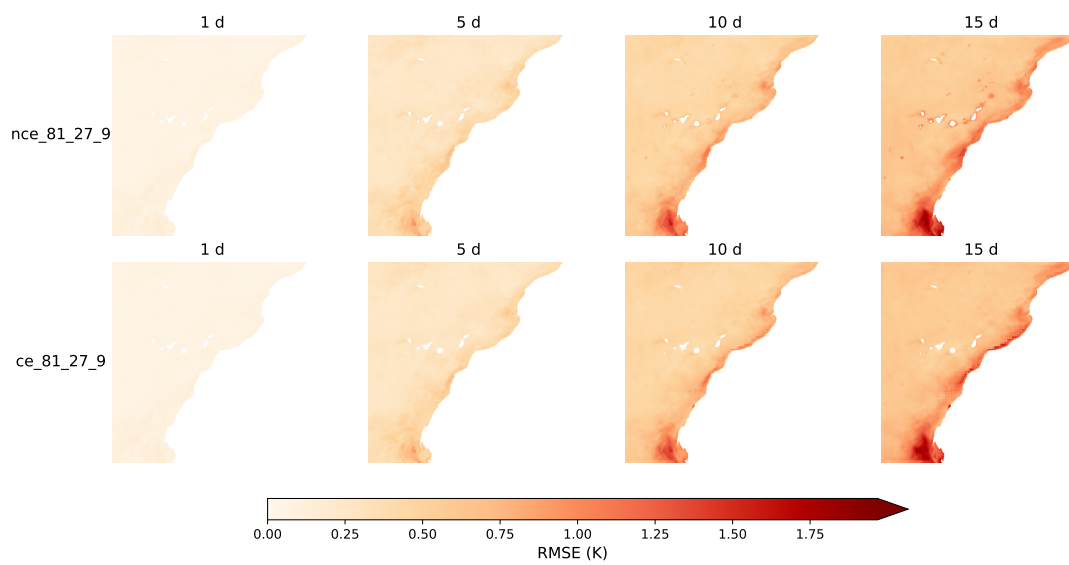
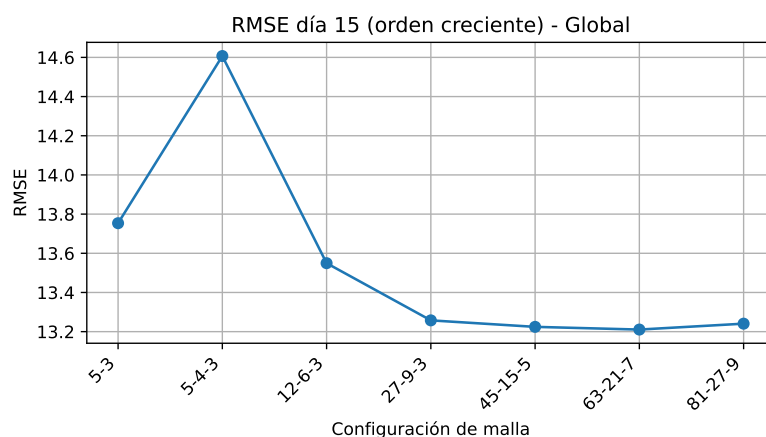
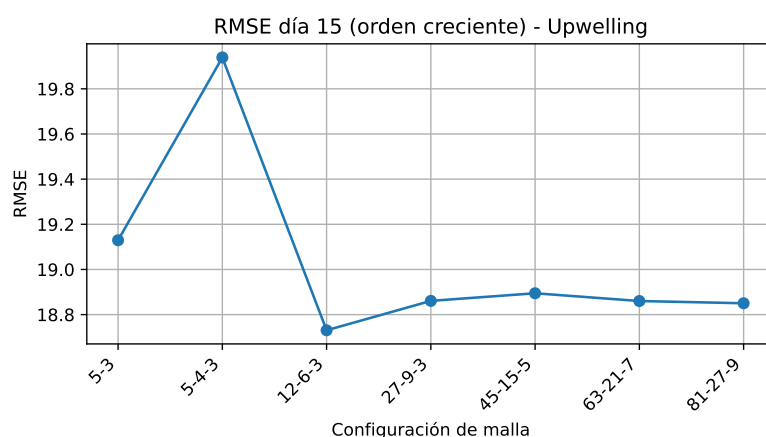


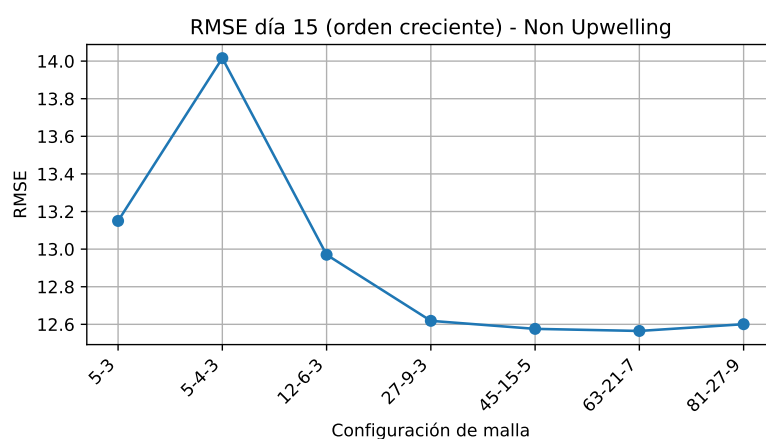
Figura 7.9: Distribución espacial del RMSE a distintos días del horizonte de predicción (1, 5, 10 y 15) para las configuraciones `nce_81_27_9` y `ce_81_27_9`. Esta representación permite observar diferencias en el comportamiento espacial del error, especialmente en zonas costeras.



(a) Comparativa de la media de RMSE en toda la región para las diferentes configuraciones.



(b) Comparativa de la media de RMSE en la región costera para las diferentes configuraciones.



(c) Comparativa de la media de RMSE en el mar abierto para las diferentes configuraciones.

Figura 7.10: Comparativa del error cuadrático medio (RMSE) en el último día del horizonte de predicción para las distintas configuraciones de malla degradadas jerárquicamente. Se muestra la media del RMSE ordenada de menor a mayor en tres regiones: (a) toda la zona de estudio, (b) región costera afectada por el fenómeno de afloramiento, y (c) mar abierto. Esta comparación permite evaluar el impacto de la reducción de resolución en distintos contextos geográficos.

7.4. Comparativa de mallas con resolución optimizada

El tercer experimento de este trabajo tiene como objetivo comparar el rendimiento de distintas configuraciones de malla generadas mediante técnicas más estructuradas y controladas. A diferencia de los experimentos anteriores, en este caso se ha optado por excluir las mallas aleatorias y aquellas basadas exclusivamente en distribuciones de probabilidad. Esta decisión se ha tomado principalmente por limitaciones temporales y por la menor capacidad de estas mallas para garantizar una cobertura homogénea del dominio oceanográfico; sin embargo, hemos conseguido obtener resultados positivos en su uso según el artículo Reyes et al. [2025], que sigue también las líneas del primer experimento mostrado.

En su lugar, el enfoque se ha centrado en evaluar mallas construidas mediante el método de muestreo por el punto más lejano (*Farthest Point Sampling* (FPS)), debido a su capacidad para generar distribuciones de nodos más uniformes y estructuradas, especialmente adecuadas para representar con mayor fidelidad las complejidades geomorfológicas de la región de estudio.

Por otra parte, se ha decidido eliminar el tercer nivel de resolución de las mallas utilizadas, dado que en experimentos previos se observó una escasa densidad de nodos en dicho nivel, lo cual dificultaba la conectividad y generaba nodos aislados. Esta falta de coherencia topológica compromete el aprendizaje del modelo y, por tanto, su eliminación busca mejorar la estabilidad y validez de las configuraciones evaluadas.

En lo que respecta a la nomenclatura adoptada, se mantiene el nombre `nce` para referirse a la malla uniforme sin aristas diagonales cruzadas, utilizada como punto de referencia. Por su parte, la malla generada mediante FPS se identifica como `fps`, y se han evaluado dos variantes densificadas de esta última: la primera, denominada `fps_base`, ha sido construida mediante una función de densidad basada en la profundidad inversa suavizada; la segunda, llamada `fps_sigmoide_mixto`, emplea una función de densidad mixta de tipo sigmoide que permite reforzar regiones específicas del dominio. Estos nombres han sido seleccionados y usados con el objetivo de referenciar con brevedad cada modelo de malla con el algoritmo utilizado para la generación de cada una.

Las características estructurales de todas las mallas analizadas en este experimento se resumen en la Tabla 7.5, y su representación visual puede observarse en la Figura 7.11.

Tabla 7.5: Características estructurales de las mallas por nivel utilizadas en el experimento de resolución optimizada.

Malla	Nivel 1		Nivel 2	
	Nodos	Aristas	Nodos	Aristas
<code>nce</code>	394	2204	45	212
<code>fps</code>	394	2228	45	212
<code>fps_base</code>	394	2208	45	190
<code>fps_sigmoide_mixto</code>	394	2234	45	216

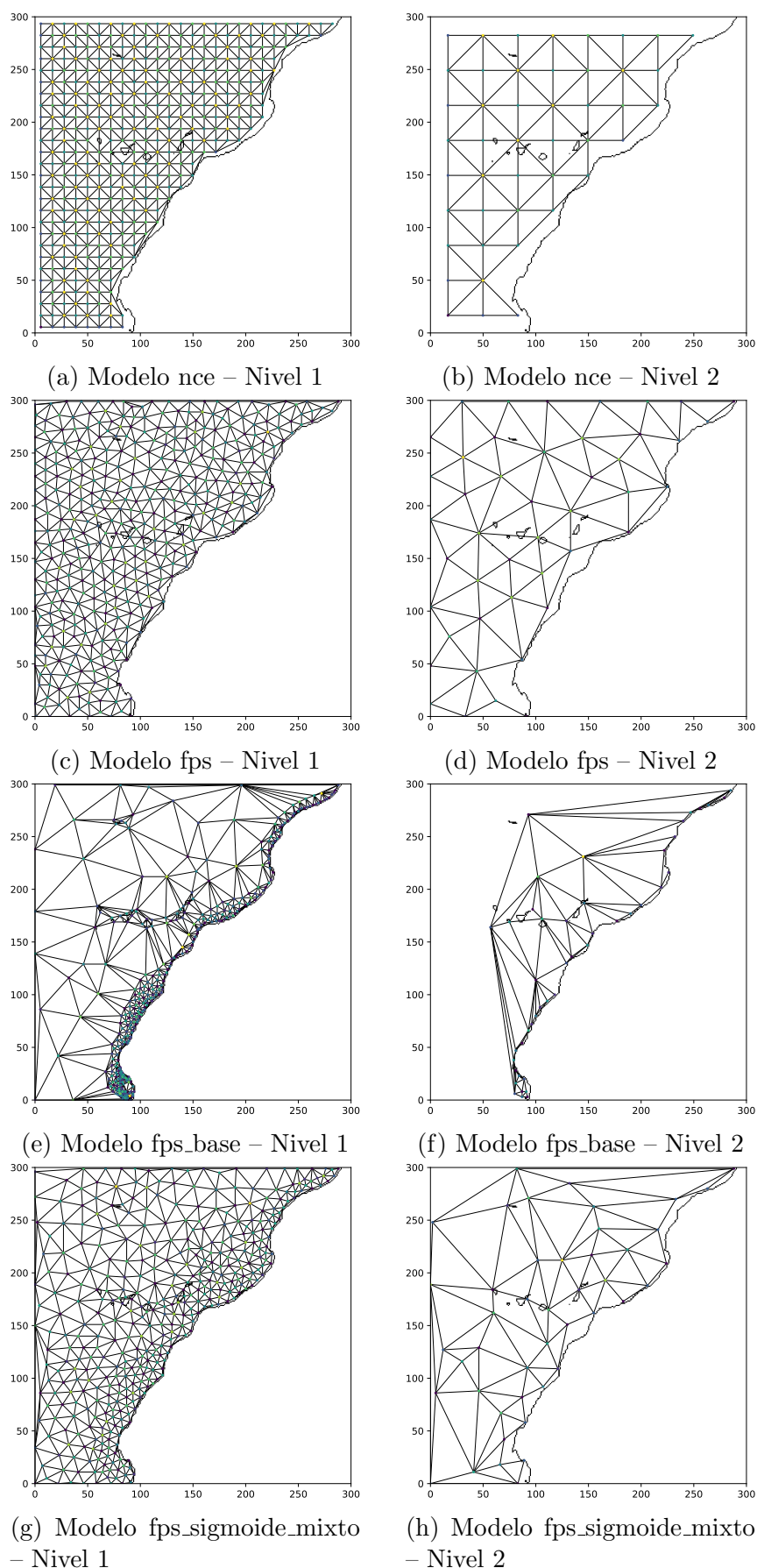


Figura 7.11: Comparativa visual de las cuatro configuraciones de malla utilizadas en la comparativa de mallas con resolución optimizada. Cada fila representa un modelo diferente y cada columna un nivel de resolución.

Resultados obtenidos de la evaluación comparativa de mallas basadas en muestreo y densificación controlada

Los resultados obtenidos en este experimento indican que las mallas generadas mediante muestreo del punto más lejano (FPS) no presentan una mejora sustancial en términos de error medio global respecto a la malla de referencia *nce* (uniforme sin cruces diagonales), tal como se observa en la Figura 7.12. A pesar de ello, el análisis espacial del error proporciona información más relevante.

En la Figura 7.13, se puede apreciar que los modelos entrenados con las mallas generadas por el muestreo del punto más lejano no presentan artefactos, donde podemos concluir que este fenómeno se da principalmente en mallas regulares estructuradas cuadrangulares, siendo este detalle especialmente notable en las regiones costeras, donde la geometría del dominio es más irregular y donde mejor se adaptan las mallas propuestas en este Trabajo de Fin de Grado. Esta mejora en la distribución espacial del error se ve reforzada en la Figura 7.14, que compara directamente las predicciones y los errores localizados para el último día de pronóstico. Estas observaciones sugieren que, aunque las métricas globales no reflejan diferencias significativas, las mallas denominadas *fps* presentan una mayor capacidad para adaptarse a las particularidades topográficas de la región de estudio, ofreciendo predicciones más suaves y realistas en zonas complejas.

Para cuantificar con mayor detalle estas diferencias, se ha elaborado la Tabla 7.6, en la que se presentan los valores medios del RMSE correspondientes al último día de predicción, así como los porcentajes de mejora respecto al modelo base *nce* en distintas subregiones: los cabos Blanco, Ghir y Juby, la franja costera africana en su conjunto y el mar abierto. En esta comparación, el modelo construido con la malla *fps_sigmoides_mixto*, densificada según una función de distribución sigmoide mixta, destaca como la configuración más eficaz, alcanzando los mayores niveles de mejora relativa en la mayoría de zonas evaluadas. Respecto a la anteriormente mencionada Tabla 7.6, podemos apreciar como la zona donde nuestros modelos de malla han empeorado ha sido en el cabo Blanco, donde achacamos esta pérdida de precisión a la complejidad propia de la predicción de la temperatura superficial del mar, donde consideramos que sería necesario poder tomar un espacio mayor al sur para conseguir una mejor predicción en dicha zona, donde destacamos como en las demás zonas de interés propuestas si que se ha mejorado ligeramente la predicción, sobre todo en los cabos Ghir y Juby.

Finalmente, en la Tabla 7.7 se resumen los tiempos de entrenamiento requeridos por cada uno de los modelos. Los resultados muestran una alta consistencia entre las distintas configuraciones, con una variación máxima de apenas seis minutos entre el modelo más lento y el más rápido. Esta estabilidad en el coste computacional refuerza la viabilidad práctica de utilizar mallas estructuradas basadas en el muestreo por el punto más lejano, especialmente cuando se busca optimizar la representación espacial sin incurrir en penalizaciones significativas en tiempo de entrenamiento.

En conjunto, este experimento pone de manifiesto que las mallas generadas mediante muestreo por el punto más lejano, destacando especialmente sus variantes densificadas, ofrecen una representación más adecuada del dominio costero oceánico, mejorando la calidad espacial de las predicciones sin comprometer la eficiencia computacional, eliminando a su vez la presencia de artefactos tanto en las predicciones como en las representaciones

del RMSE espacial.

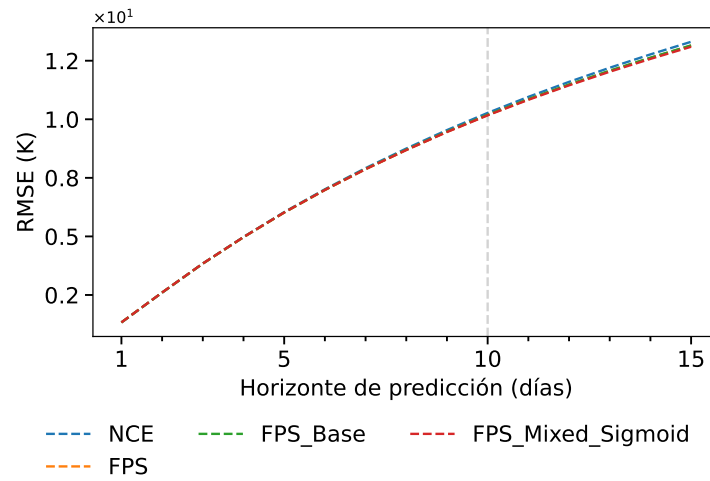
Por último, cabe señalar que la muestra de datos utilizada para el entrenamiento del modelo ha sido limitada, lo que probablemente ha condicionado la capacidad del sistema para alcanzar resultados más concluyentes. Por ello, se considera altamente recomendable ampliar el conjunto de datos empleado, de modo que se puedan confirmar o refinar las observaciones obtenidas en este capítulo. Esta ampliación será contemplada como una de las principales líneas de trabajo futuro asociadas a este proyecto.

Tabla 7.6: Comparativa del RMSE y mejora porcentual por zonas para distintas configuraciones de malla respecto a la malla *nce*. Se presentan los valores medios del RMSE en el último día de predicción, así como las mejoras relativas (en porcentaje) alcanzadas en distintas subregiones: cabos Blanco, Ghir y Juby, franja costera africana y mar abierto.

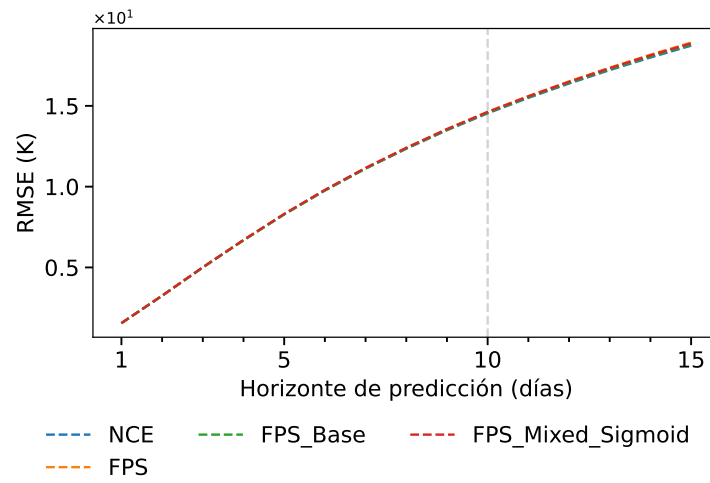
Malla	RMSE global (K)	Cabo Blanco (%)	Cabo Ghir (%)	Cabo Juby (%)	Costa Africana (%)	Mar Abierto (%)
<i>nce</i>	13.31	–	–	–	–	–
<i>fps</i>	13.15	-0.60	3.68	1.66	-0.31	1.84
<i>fps_base</i>	13.16	-0.50	3.83	2.42	0.47	1.49
<i>fps_sigmoide_mixto</i>	13.10	-0.63	4.72	2.15	0.09	2.27

Tabla 7.7: Tiempos de entrenamiento requeridos por cada modelo de malla en la comparativa de configuraciones con resolución optimizada. La duración total (en horas y minutos) refleja que las diferencias entre configuraciones son mínimas, lo que sugiere una elevada estabilidad computacional incluso al introducir mecanismos de densificación espacial.

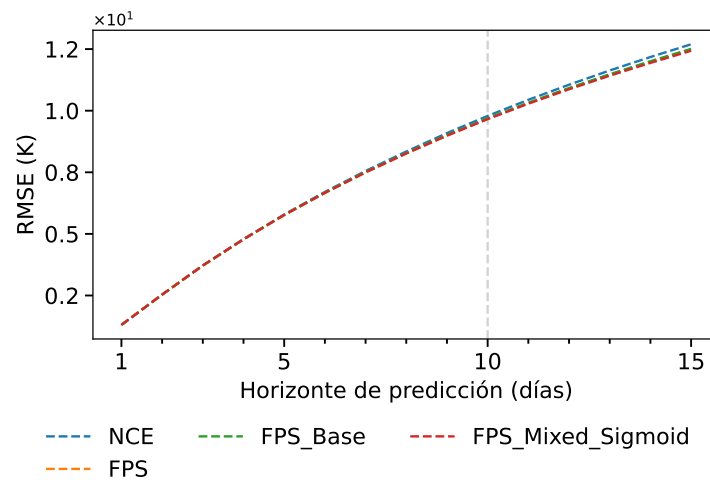
Malla	Tiempo de entrenamiento
<i>nce</i>	9 h 29 min
<i>fps</i>	9 h 30 min
<i>fps_base</i>	9 h 35 min
<i>fps_sigmoide_mixto</i>	9 h 34 min



(a) RMSE en toda la región de estudio



(b) RMSE en la costa africana



(c) RMSE en el mar abierto

Figura 7.12: Evolución del error cuadrático medio (RMSE) a lo largo del horizonte de predicción de 15 días para las distintas configuraciones de malla con resolución optimizada. Se presentan resultados en tres ámbitos espaciales: (a) toda la región de estudio, (b) la zona costera afectada por el fenómeno de afloramiento, y (c) el mar abierto. Esta representación permite evaluar la capacidad de generalización de cada configuración a lo largo del tiempo y en distintas regiones del dominio.

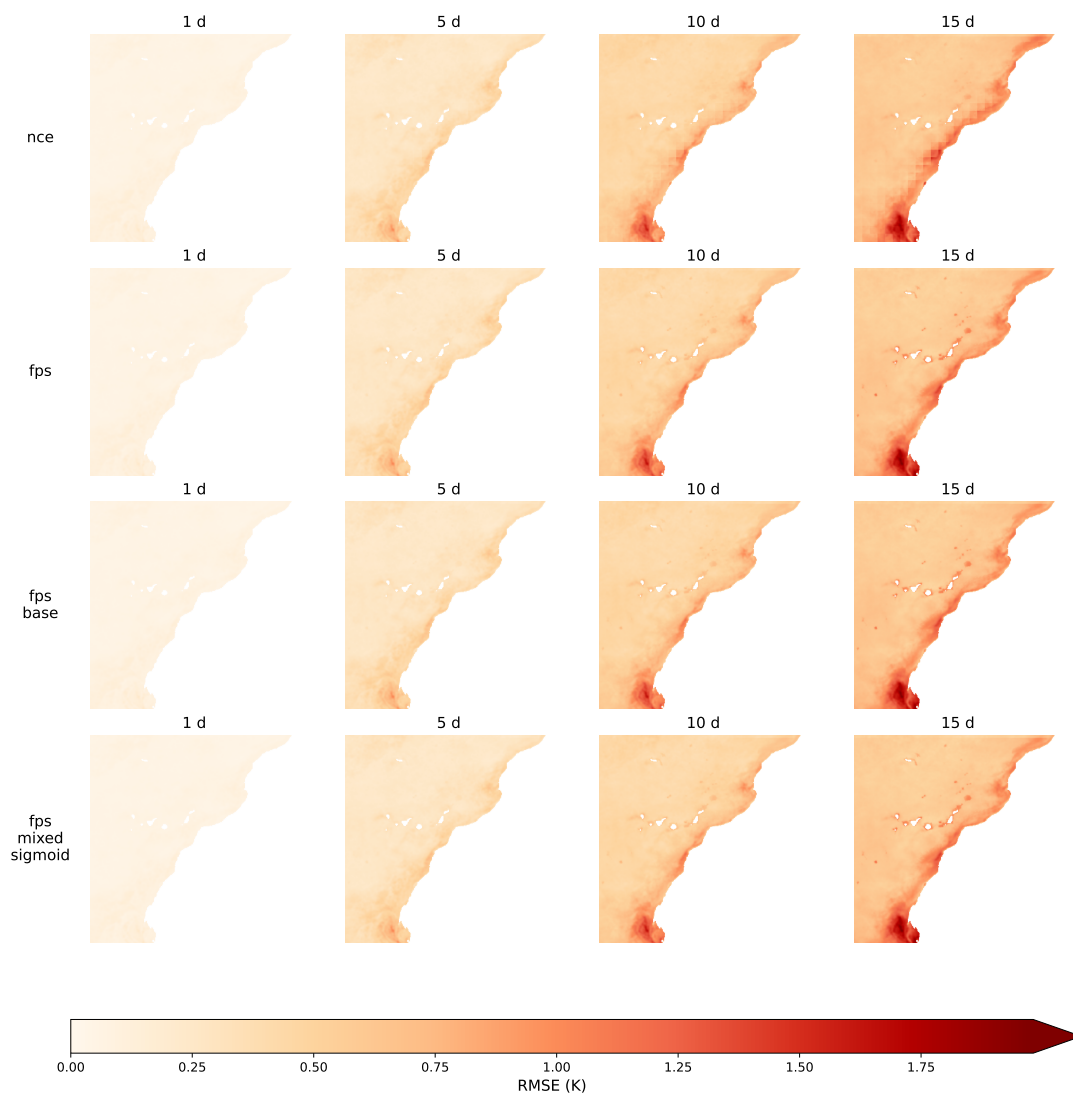
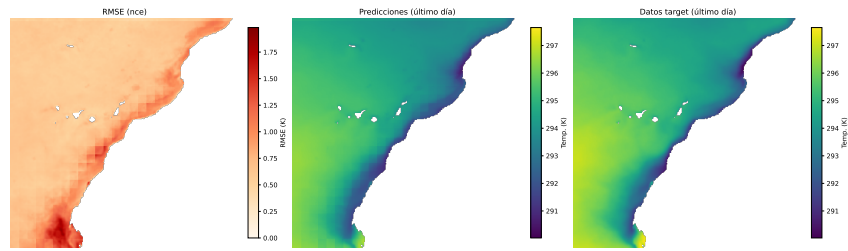
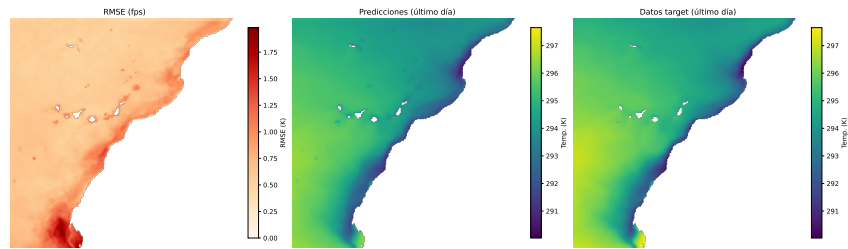


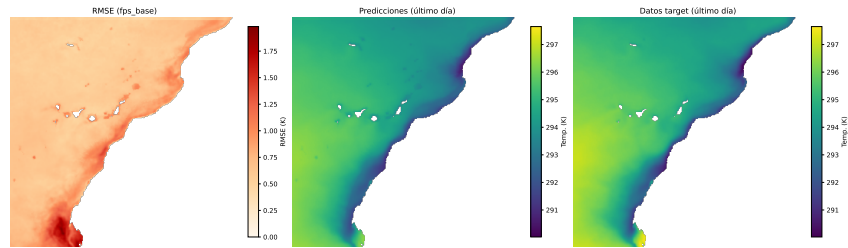
Figura 7.13: Distribución espacial del RMSE para los días 1, 5, 10 y 15 del horizonte de predicción, comparando las configuraciones de malla `nce`, `fps`, `fps_base` y `fps_mixed-sigmoid`. Donde se puede apreciar la eliminación de la presencia de artefactos en las zonas costeras.



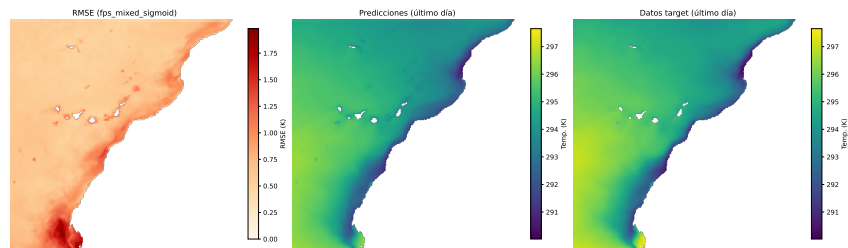
(a) `nce`: RMSE espacial, predicción y valor objetivo en el último día.



(b) `fps`: RMSE espacial, predicción y valor objetivo en el último día.



(c) `fps_base`: RMSE espacial, predicción y valor objetivo en el último día.



(d) `fps_sigmoide_mixto`: RMSE espacial, predicción y valor objetivo en el último día.

Figura 7.14: Comparativa espacial entre modelos de malla en el último día del horizonte de predicción. Para cada configuración (`nce`, `fps`, `fps base` y `fps sigmoide mixto`) se muestran tres mapas: el RMSE espacial, la predicción del modelo y los datos objetivos. Esta representación permite evaluar visualmente el ajuste del modelo y las diferencias en la distribución del error en función del diseño de la malla.

Capítulo 8

Conclusiones y trabajo futuro

8.1. Conclusiones

En este trabajo se ha puesto de manifiesto la relevancia que tiene el diseño de la malla subyacente en los modelos basados en redes neuronales de grafos (GNN, por sus siglas en inglés). A lo largo del desarrollo del proyecto, se ha evidenciado que la elección y configuración de dicha malla no es un aspecto menor o secundario, sino un componente fundamental que influye de manera directa en el rendimiento, la eficiencia y la escalabilidad de los modelos implementados.

En particular, se ha demostrado que una construcción adecuada de la malla puede traducirse en un ahorro significativo de recursos computacionales, permitiendo reducir tanto el tiempo de entrenamiento como la necesidad de capacidad de cómputo sin comprometer la calidad de los resultados. Esta optimización abre la puerta a dos beneficios clave: por un lado, posibilita el incremento de la complejidad del modelo (por ejemplo, añadiendo más capas o parámetros) sin superar las restricciones computacionales existentes; por otro, permite incluir un mayor volumen de datos durante el proceso de entrenamiento, lo que potencialmente mejora la capacidad generalizadora del modelo.

Por otro lado, los resultados obtenidos han evidenciado que los modelos basados en mallas densificadas han logrado mejorar la precisión de las predicciones en los alrededores de la mayoría de los cabos más relevantes de la zona de estudio. Este comportamiento demuestra que es posible optimizar el rendimiento de las GNNs en regiones geográficamente complejas sin recurrir a mecanismos de atención. Se trata de un hallazgo especialmente relevante en el contexto de este trabajo, ya que ofrece un enfoque eficiente desde el punto de vista computacional y constituye un punto de partida prometedor para el diseño de arquitecturas GNN aplicadas a otros ámbitos más allá del oceanográfico, manteniendo la eficacia predictiva sin incrementar los costes de inferencia.

Profundizando en las mejoras obtenidas en las predicciones, la eliminación de artefactos espaciales en los resultados generados por el modelo representa uno de los hallazgos más relevantes de los experimentos realizados. La aparición de este tipo de errores ha sido documentada de forma recurrente en trabajos previos [Cuervo-Londoño et al., 2025; Oskarsson et al., 2023], lo que refuerza la importancia del presente resultado. En este contexto, las estructuras de malla propuestas en este trabajo se consolidan como una solución

prometedora para mitigar este problema y mejorar la calidad espacial de las predicciones.

Además, estos avances no solo tienen implicaciones técnicas, sino también medioambientales. La eficiencia computacional se traduce directamente en una menor demanda energética, lo cual contribuye a mitigar la huella de carbono asociada al desarrollo y entrenamiento de modelos de aprendizaje profundo. En un contexto en el que la sostenibilidad tecnológica adquiere una importancia creciente, este tipo de enfoques eficientes representa una vía prometedora hacia una inteligencia artificial más respetuosa con el medio ambiente.

Por otra parte, dados los recursos disponibles para la elaboración de este proyecto, se ha tenido que utilizar una muestra pequeña de datos para el entrenamiento de los diferentes modelos, donde se considera que este factor ha sido altamente determinante en los resultados obtenidos, donde no se han conseguido mejoras significativas en cuanto a la predicción de variables oceanográficas por esta vía al no haber podido utilizar una mayor cantidad de años para el entrenamiento de los modelos.

8.2. Trabajo futuro

El presente trabajo ha permitido explorar el uso de redes neuronales de grafos para la predicción de variables meteorológicas en un entorno geoespacial. No obstante, como todo desarrollo experimental, el modelo actual presenta diversas oportunidades de mejora que podrían ser abordadas en futuras investigaciones. A continuación, se detallan algunas de las líneas más prometedoras para continuar ampliando y refinando los resultados obtenidos.

En primer lugar, una consideración relevante es el aumento del tamaño de la muestra empleada durante el entrenamiento. Una mayor cantidad de datos permitiría evaluar con mayor rigor la validez y robustez de los resultados obtenidos, al tiempo que podría mejorar la capacidad generalizadora del modelo.

Por otro lado, resulta pertinente explorar arquitecturas alternativas a la red neuronal de grafos utilizada en este trabajo. En lugar de emplear una arquitectura personalizada basada en interacciones, sería interesante analizar el desempeño de distintos modelos disponibles en la librería `PyTorch Geometric`, con especial atención a aquellos que incorporan mecanismos de atención. Estas técnicas podrían mejorar significativamente la capacidad del modelo para capturar relaciones espaciales complejas entre nodos.

Asimismo, se plantea como una línea de mejora el ajuste de los hiperparámetros del modelo, tales como el número de capas o la dimensionalidad de las capas ocultas. Dado que el número de nodos utilizados para representar la zona de estudio ha experimentado un aumento considerable, resulta razonable suponer que una reconfiguración de estos parámetros podría no solo mejorar la precisión de las predicciones, sino también optimizar los tiempos de cómputo.

Por otra parte, los resultados obtenidos abren la puerta al desarrollo de nuevos enfoques tanto en la estructuración de mallas como en los métodos de densificación, con el objetivo de superar el rendimiento alcanzado en este trabajo. Una posible línea de mejora consiste en sustituir la batimetría como variable guía del proceso de densificación por otro

factor más representativo de la dinámica oceánica, lo cual podría proporcionar una adaptación espacial más eficaz. Asimismo, sería factible explorar el diseño de mallas dinámicas que evolucionen en función del error observado durante la validación o de otras métricas relevantes. En este sentido, una alternativa intermedia podría consistir en generar diferentes mallas estáticas adaptadas a las condiciones estacionales, por ejemplo, utilizando la temperatura media de cada estación como criterio estructurante.

Una posible línea de mejora futura consistiría en la incorporación de métricas complementarias al RMSE para la evaluación del rendimiento del modelo. Aunque en este trabajo se ha optado por utilizar principalmente el RMSE debido a su amplia aceptación en la literatura y su capacidad para ofrecer una medida cuantitativa clara del error, donde la inclusión de otras métricas podrían proporcionar una visión más completa y matizada sobre la precisión alcanzada por las nuevas configuraciones de malla. Esto permitiría identificar mejor las fortalezas y limitaciones del modelo en distintos contextos espaciales o temporales, enriqueciendo así el análisis comparativo de los resultados.

Otra cuestión destacable hace referencia al problema de las predicciones excesivamente suavizadas, un fenómeno común en modelos de predicción meteorológica. Tal como se indica en Oskarsson et al. [2023], la incorporación de enfoques probabilísticos o la inyección de ruido en diferentes etapas del proceso de entrenamiento puede contribuir a generar predicciones más realistas al introducir un grado controlado de incertidumbre.

En esta misma línea, se considera que una optimización más cuidadosa de las etapas de codificación y decodificación, haciendo especial énfasis tanto en el número de aristas como en el mecanismo de conexión entre los nodos. Esto podría favorecer una transmisión de información más efectiva entre las distintas representaciones internas. Esto redundaría en una mejor capacidad del modelo para reflejar con fidelidad la evolución de fenómenos geoespaciales.

Finalmente, se sugiere mejorar el proceso de paso de mensajes entre nodos en la red neuronal de grafos. Una posible vía sería la incorporación de formulaciones basadas en principios físicos, concretamente de la termodinámica, con el objetivo de modelar de manera más realista el intercambio de temperatura entre regiones. Este enfoque podría aportar una mayor coherencia física al comportamiento del modelo y mejorar la calidad de las predicciones generadas.

Apéndice A

Competencias específicas cubiertas

En este capítulo se recogen las competencias específicas desarrolladas a lo largo del Trabajo de Fin de Grado, junto con una justificación argumentada de su cumplimiento:

- **ED3: Capacidad para conocer y desarrollar técnicas de aprendizaje computacional y diseñar e implementar aplicaciones y sistemas que las utilicen, incluyendo las dedicadas a la extracción automática de información y conocimiento a partir de grandes volúmenes de datos.**

Esta competencia se ha desarrollado mediante el entrenamiento sistemático de modelos de aprendizaje profundo, lo cual ha requerido la implementación de estrategias automatizadas para la gestión, procesamiento y estructuración de los datos obtenidos desde distintas fuentes oceanográficas y atmosféricas.

- **ED4: Capacidad para identificar y analizar problemas y diseñar, desarrollar, implementar, verificar y documentar soluciones software en ámbitos de aplicación de la Inteligencia Artificial en Ciencia e Ingeniería de Datos.**

A lo largo del proyecto se ha abordado el diseño y evaluación de distintas estructuras de malla aplicadas a redes neuronales de grafos, proponiendo configuraciones no uniformes adaptadas a la complejidad del dominio oceánico. Este enfoque se ha orientado a mejorar la precisión de las predicciones mediante soluciones arquitectónicas específicas, desarrolladas y documentadas de forma rigurosa.

- **ED6: Capacidad para tener un conocimiento profundo de los principios fundamentales y modelos utilizados en Ciencia de Datos, particularmente los relacionados con el análisis, predicción y prospectiva de grandes volúmenes de datos.**

Esta competencia se ha cubierto mediante el estudio e implementación de técnicas avanzadas de modelado, no solo desde la perspectiva de la parametrización, sino también desde el diseño estructural de los datos de entrada al modelo. En particular, se ha profundizado en la influencia de la topología de las mallas sobre el rendimiento de redes neuronales de grafos, evaluando el impacto en la difusión de la información a través de los pasos de mensaje.

- **ED7: Capacidad para participar activamente en la especificación, diseño, implementación y mantenimiento de los sistemas de información y de las**

comunicaciones de uso común en Ciencia e Ingeniería de Datos, así como para diseñar e implementar software de aplicaciones para estos sistemas.

A lo largo del desarrollo se ha seguido la metodología CRISP-DM, combinada con una planificación iterativa y validación continua a través de reuniones técnicas con los tutores. Este enfoque ha permitido especificar y refinar con precisión los objetivos del sistema, así como plantear e implementar soluciones novedosas en el contexto del modelado de variables oceanográficas usando GNNs.

Apéndice B

Objetivos de desarrollo sostenible

Tras haber obtenido conclusiones del desarrollo de este trabajo y cubierto las competencias específicas, procederemos a detallar los ODS (objetivos de desarrollo sostenible) de la Agenda 2030. Para ello, cumplimentaremos la Tabla B.1 marcando aquellos objetivos que tengan cierta relación con la temática de este estudio, reflejando en el proceso la importancia que tiene la predicción de las variables oceanográficas.

Se justifica un alto grado de relación del TFT con los ODS14 (vida submarina), ODS13 (Acción por el clima) y ODS9 (Industria, Innovación e Infraestructuras) por los siguientes motivos:

- Respecto al ODS14, justificamos su relación dado que el estudio de variables oceanográficas es de vital importancia para el estudio del comportamiento de la vida marina y conocer, entre otros aspectos, cambios en los hábitats de las diferentes especies que habitan en la zona de estudio
- Respecto al ODS13, justificamos su relación dado que el estudio oceanográfico aporta datos significativos sobre el cambio climático así como del calentamiento global ya que es estado del mar refleja de forma muy precisa estos aspectos.
- Respecto al ODS9, justificamos su relación dado que el uso de este tipo de redes neuronales para la predicción oceanográfica es muy reciente, y, además, se ha detectado que el impacto de la malla en estas redes no está suficientemente explorado.

Se justifica un grado medio de relación con el TFT el ODS2 (2 Hambre cero) dado que, dentro de las variables oceanográficas estudiadas, nos hemos centrado principalmente en la predicción de la temperatura superficial del mar, siendo esta variable significativa para determinar zonas y momentos con abundantes cantidades de peces, siendo clave para el sector pesquero.

Se justifica un bajo grado de relación del TFT con los ODS3 (Salud y Bienestar) y ODS6(Agua limpia y saneamiento) por los siguientes motivos:

- Respecto al ODS3, justificamos su baja relación dado que el estudio del océano, como se ha indicado en la justificación del ODS2, puede ser útil para mejorar la

productividad del sector pesquero, mejorando el bienestar de la población con aumentos en la oferta de este producto.

- Respecto al ODS6, justificamos que su relación es baja con nuestro trabajo, donde, si bien la investigación oceanográfica puede contribuir a mejorar la calidad del agua, en este estudio no nos hemos enfocado en variables que inciden directamente en este aspecto del océano. Por ello, estimamos que el impacto sobre este objetivo de desarrollo sostenible es mínimo, teniendo únicamente un efecto colateral.

ODS	Nivel de relación			
	0	1	2	3
1. Fin de la Pobreza	X			
2. Hambre cero			X	
3. Salud y Bienestar	X			
4. Educación de calidad	X			
5. Igualdad de género	X			
6. Agua limpia y saneamiento		X		
7. Energía asequible y no contaminante	X			
8. Trabajo decente y crecimiento económico	X			
9. Industria, Innovación e Infraestructuras				X
10. Reducción de las desigualdades	X			
11. Ciudades y comunidades sostenibles	X			
12. Producción y consumo sostenibles	X			
13. Acción por el clima				X
14. Vida submarina				X
15. Vida de ecosistemas terrestres	X			
16. Paz, justicia e instituciones sólidas	X			
17. Alianzas para lograr objetivos	X			

Tabla B.1: Tabla con la valoración del grado de relación del TFG con los ODS, según una escala de 0 a 3 (no procede, bajo, medio, alto).

Bibliografía

- Barber, C. B., Dobkin, D. P., and Huhdanpaa, H. (1996). The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software (TOMS)*, 22(4):469–483.
- Battaglia, P. W., Hamrick, J. B., Bapst, V., Sanchez-Gonzalez, A., Zambaldi, V., Malinowski, M., Tacchetti, A., Raposo, D., Santoro, A., Faulkner, R., et al. (2018). Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*.
- Battaglia, P. W., Pascanu, R., Lai, M., Jimenez Rezende, D., and Kavukcuoglu, K. (2016). Interaction networks for learning about objects, relations and physics. In *Advances in Neural Information Processing Systems*, volume 29, pages 4502–4510.
- Ben Bouallègue, Z., Clare, M. C. A., Magnusson, L., Gascon, E., Maier-Gerber, M., Janoušek, M., Rodwell, M., Pinault, F., Dramsch, J. S., Lang, S. T. K., and et al. (2024). The Rise of Data-Driven Weather Forecasting: A First Statistical Assessment of Machine Learning–Based Weather Forecasts in an Operational-like Context. *Bulletin of the American Meteorological Society*, 105(6):E864–E883.
- Benedetti, F., Ayata, S.-D., Adloff, M., Hélaouët, P., Dodson, J. J., and Irisson, J.-O. (2023). Machine learning in marine ecology: an overview of techniques and applications. *ICES Journal of Marine Science*, 80(7):1829–1844.
- Blayo, E. and Debreu, L. (1999). Adaptive mesh refinement for finite-difference ocean models: First experiments. *Journal of Physical Oceanography*, 29(6):1239–1250.
- Bleck, R. (2002). An oceanic general circulation model framed in hybrid isopycnic-cartesian coordinates. *Ocean Modelling*, 4(1):55–88.
- Bodnar, C., Bruinsma, W. P., Lucic, A., Stanley, M., Vaughan, A., Brandstetter, J., Garvan, P., Riechert, M., Weyn, J. A., Dong, H., Gupta, J. K., Thambiratnam, K., Archibald, A. T., Wu, C.-C., Heider, E., Welling, M., Turner, R. E., and Perdikaris, P. (2024). A foundation model for the earth system.
- Bolton, T. and Zanna, L. (2019). Applications of deep learning to ocean data inference and subgrid parameterization. *Journal of Advances in Modeling Earth Systems*, 11(1):376–399.
- Bronstein, M. M., Bruna, J., Cohen, T., and Velicković, P. (2021). Geometric deep learning: Grids, groups, graphs, geodesics, and gauges.

- Chassignet, E. P., Hurlburt, H. E., Smedstad, O. M., Halliwell, G. R., Hogan, P. J., Wallcraft, A. J., Baraille, R., and Bleck, R. (2007). The hycom (hybrid coordinate ocean model) data assimilative system. *Journal of Marine Systems*, 65(1-4):60–83.
- Chen, Y., Huang, X., Luo, J.-J., Lin, Y., Wright, J. S., Lu, Y., Chen, X., Jiang, H., and Lin, P. (2023). Prediction of ENSO using Multivariable Deep Learning. *Atmospheric and Oceanic Science Letters*, 16(4):100350.
- Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation.
- Clementi, E., Drudi, M., Aydogdu, A., Moulin, A., Grandi, A., Mariani, A., Goglio, A. C., Pistoia, J., Miraglio, P., Lecci, R., Palermo, F., Coppini, G., Masina, S., and Pinardi, N. (2023). Mediterranean Sea physical analysis and forecast (version 1). Technical report, Copernicus Monitoring Environment Marine Service (CMEMS).
- Cuervo-Londoño, G., Sánchez, J., and Rodríguez-Santana, A. (2025). Deep learning weather models for subregional ocean forecasting: A case study on the canary current upwelling system. *arXiv preprint arXiv:2505.24429*.
- Cuervo-Londoño, G. A., Sánchez, J., and Ángel Rodríguez-Santana (2025). Forecasting sea surface temperature from satellite images with graph neural networks. In *Proceedings of the 21st International Conference in Computer Analysis of Images and Patterns (CAIP 2025)*, Lecture Notes in Computer Science (LNCS), Las Palmas de Gran Canaria, Spain. Springer Verlag. Accepted for publication.
- Cuomo, S., Schiano Di Cola, V., Giampaolo, F., Rozza, G., Raissi, M., and Piccialli, F. (2022). Scientific machine learning through physics-informed neural networks: Where we are and what’s next. *Journal of Scientific Computing*, 92(3):88.
- Debreu, L., Vouland, C., and Blayo, E. (2007). Agrif: Adaptive grid refinement in fortran. *Computers & Geosciences*, 33(4):395–400.
- Eldar, Y., Lindenbaum, M., Porat, M., and Zeevi, Y. (1997). The farthest point strategy for progressive image sampling. *IEEE Transactions on Image Processing*, 6(9):1305–1315.
- Escudier, R., Clementi, E., Cipollone, A., Pistoia, J., Drudi, M., Grandi, A., Lyubartsev, V., Lecci, R., Aydogdu, A., Delrosso, D., et al. (2021). A high resolution reanalysis for the Mediterranean Sea. *Frontiers in Earth Science*, 9:702285.
- European Union - Copernicus (2025). Sobre copernicus. <https://www.copernicus.eu/es/sobre-copernicus>. Accessed on 06-05-2025.
- European Union-Copernicus Marine Service (2019). European north west shelf/iberia biscay irish seas - high resolution l4 sea surface temperature reprocessed. https://resources.marine.copernicus.eu/product-detail/SST_ATL_SST_L4_REP_OBSERVATIONS_010_026/INFORMATION. Accessed on 17-02-2025.
- Fablet, R., Febvre, Q., Drumetz, L., and Rousseau, F. (2022). Recent developments in artificial intelligence in oceanography. *Ocean-Land-Atmosphere Research*, 2022:9870950.

- Feng, M., Boschetti, F., Ling, F., Zhang, X., Hartog, J. R., Akhtar, M., Shi, L., Gardner, B., Luo, J.-J., and Hobday, A. J. (2022). Predictability of sea surface temperature anomalies at the eastern pole of the Indian Ocean Dipole—using a convolutional neural network model. *Frontiers in Climate*, 4:925068.
- Feng, Z., Wang, R., Wang, T., Song, M., Wu, S., and He, S. (2024). A comprehensive survey of dynamic graph neural networks: Models, frameworks, benchmarks, experiments and challenges.
- Gattuso, J.-P., Magnan, A. K., Bopp, L., Cheung, W. W., Duarte, C. M., Hinkel, J., Mcleod, E., Micheli, F., Oschlies, A., Williamson, P., et al. (2018). Ocean solutions to address climate change and its effects on marine ecosystems. *Frontiers in Marine Science*, 5:337.
- Ge, L., Zhang, J., Zhao, X., Zhou, X., and Li, Q. (2024). A non-uniform grid graph convolutional network for sea surface temperature prediction. *Remote Sensing*, 16(17):3216.
- Goyal, P., Chhetri, S. R., and Canedo, A. (2020). dyngraph2vec: Capturing network dynamics using dynamic graph representation learning. *Knowledge-Based Systems*, 187:104816.
- Griffies, S. M., Winton, M., Anderson, W. G., Benson, R., Delworth, T. L., Dufour, C. O., Dunne, J. P., Goddard, P., Morrison, A. K., Rosati, A., et al. (2010). Problems and prospects in large-scale ocean circulation models. In *Proceedings of OceanObs'09: Sustained Ocean Observations and Information for Society*, volume 2, pages 437–458. ESA Publication WPP-306.
- Hersbach, H., Bell, B., Berrisford, P., Biavati, G., Horányi, A., Muñoz Sabater, J., Nicolas, J., Peubey, C., Radu, R., Rozum, I., Schepers, D., Simmons, A., Soci, C., Dee, D., and Thépaut, J.-N. (2023). Era5 hourly data on single levels from 1940 to present. <https://doi.org/10.24381/cds.adbb2d47>. Accessed on 18-02-2025.
- Hersbach, H., Bell, B., Berrisford, P., Hirahara, S., Horányi, A., Muñoz-Sabater, J., Nicolas, J., Peubey, C., Radu, R., Schepers, D., et al. (2020). The ERA5 global reanalysis. *Quarterly Journal of the Royal Meteorological Society*, 146(730):1999–2049.
- Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.
- Holmberg, D., Clementi, E., and Roos, T. (2024). Regional ocean forecasting with hierarchical graph neural networks. *arXiv preprint arXiv:2410.11807*.
- Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences of the United States of America*, 79(8):2554–2558.
- Hou, S., Li, W., Liu, T., Zhou, S., Guan, J., Qin, R., and Wang, Z. (2021). D2CL: A Dense Dilated Convolutional LSTM Model for Sea Surface Temperature Prediction. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:12514–12523.
- IBM Corporation (2024). CRISP-DM help overview. Último acceso: 9 de julio de 2025.

- IPCC (2019). Ipcc special report on the ocean and cryosphere in a changing climate. Pörtner, H.-O., D.C. Roberts, V. Masson-Delmotte, P. Zhai, M. Tignor, E. Poloczanska, K. Mintenbeck, A. Alegría, M. Nicolai, A. Okem, J. Petzold, B. Rama, N.M. Weyer (eds.).
- Izumi, T., Amagasaki, M., Ishida, K., and Kiyama, M. (2022). Super-Resolution of Sea Surface Temperature with Convolutional Neural Network- and Generative Adversarial Network-Based Methods. *Journal of Water and Climate Change*, 13(4):1673–1683.
- Kim, J., Kim, T., Ryu, J. G., and Kim, J. (2023). Spatiotemporal Graph Neural Network for Multivariate Multi-Step Ahead Time-Series Forecasting of Sea Temperature. *Engineering Applications of Artificial Intelligence*, 126:106854.
- Krishnapriyan, A. S., Gholami, A., Zhe, S. D., Kirby, R. M., and Mahoney, M. W. (2021). Characterizing Possible Failure Modes in Physics-Informed Neural Networks.
- Lam, R., Sanchez-Gonzalez, A., Willson, M., Wirnsberger, P., Fortunato, M., Alet, F., Ravuri, S., Ewalds, T., Eaton-Rosen, Z., Hu, W., Merose, A., Hoyer, S., Holland, G., Vinyals, O., Stott, J., Pritzel, A., Mohamed, S., and Battaglia, P. (2022). Graphcast: Learning skillful medium-range global weather forecasting. *arXiv preprint arXiv:2212.12794*.
- LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4):541–551.
- Levin, L. A., Brito-Morales, I., Schoeman, D. S., Hellberg, M. E., Hemery, L. G., Kudela, R. M., Meinvielle, M., Morato, T., Sogin, M. L., Thomas, C. J., et al. (2020). Climate velocity reveals increasing exposure of deep-ocean biodiversity to future warming. *Nature Climate Change*, 10(6):576–584.
- Liang, S., Zhao, A., Qin, M., Hu, L., Wu, S., Du, Z., and Liu, R. (2023). A graph memory neural network for sea surface temperature prediction. *Remote Sens.*, 15(14):3539.
- Lin, Y. D. and Zhong, G. Q. (2021). A multi-channel LSTM model for sea surface temperature prediction. *Journal of Physics: Conference Series*, 1880:012029.
- Liu, X., Wilson, T., Tan, P.-N., and Luo, L. (2019). Hierarchical LSTM Framework for Long-Term Sea Surface Temperature Forecasting. In *Proceedings of the 2019 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pages 41–50, Washington, DC, USA.
- Liu, Y., Wang, J., Xiao, C., Qi, H., Wang, Y., and Wang, P. (2023). A graph memory neural network for sea surface temperature prediction. *Remote Sensing*, 15(14):3539.
- Madec, G., Bourdallé-Badie, R., Bouttier, P.-A., Bricaud, C., Bruciaferri, D., Calvert, D., Chanut, J., Clementi, E., Coward, A., Delrosso, D., et al. (2016). Nemo ocean engine. Technical Report 27, Institut Pierre-Simon Laplace (IPSL), Paris, France.
- Manessi, F., Rozza, A., and Manzo, M. (2020). Dynamic Graph Convolutional Networks. *Pattern Recognition*, 97:107000.

- Mao, K., Liu, C., Zhang, S., and Gao, F. (2023). Reconstructing Ocean Subsurface Temperature and Salinity from Sea Surface Information Based on Dual Path Convolutional Neural Networks. *Journal of Marine Science and Engineering*, 11(5):1030.
- Marshall, J., Adcroft, A., Hill, C., Perelman, L., and Heisey, C. (1997a). A finite-volume, incompressible navier stokes model for studies of the ocean on parallel computers. *Journal of Geophysical Research: Oceans*, 102(C3):5753–5766.
- Marshall, J., Hill, C., Perelman, L., and Adcroft, A. (1997b). Hydrostatic, quasi-hydrostatic, and nonhydrostatic ocean modeling. *Journal of Geophysical Research: Oceans*, 102(C3):5733–5752.
- Miao, Y. L., Zhang, C. C., Zhang, X. F., and Zhang, L. X. (2023). A Multivariable Convolutional Neural Network for Forecasting Synoptic-Scale Sea Surface Temperature Anomalies in the South China Sea. *Weather and Forecasting*, 38(6):849–863.
- Mu, B., Qin, B., and Yuan, S. (2020). Multi-scale downscaling with Bayesian convolution network for ENSO SST pattern. In *Proceedings of the 2020 5th International Conference on Electromechanical Control Technology and Transportation (ICECTT)*, pages 359–362, Nanchang, China.
- Müller, M., Homleid, M., Ivarsson, K.-I., Køltzow, M. A. Ø., Lindskog, M., Midtbø, K. H., Andrae, U., Aspelien, T., Berggren, L., Bjørge, D., Dahlgren, P., Kristiansen, J., Randriamampianina, R., Ridal, M., and Vignes, O. (2017). AROME-MetCoOp: A Nordic convective-scale operational weather prediction model. *Weather and Forecasting*, 32(2):609–627.
- NOAA National Centers for Environmental Information (2022). ETOPO 2022 15 Arc-Second Global Relief Model. Accessed: 2025-06-26.
- Oskarsson, J., Landelius, T., and Lindsten, F. (2023). Graph-based neural weather prediction for limited area modeling. *arXiv preprint arXiv:2309.17370*.
- Ou, M., Xu, S., Luo, B., Zhou, H., Zhang, M., Xu, P., Zhu, H., et al. (2024). 3-D Ocean Temperature Prediction via Graph Neural Network with Optimized Attention Mechanisms. *IEEE Geoscience and Remote Sensing Letters*, 21:1503405.
- Pain, C. C., Piggott, M. D., Goddard, A. J., Fang, F., Gorman, G. J., Marshall, D. P., Eaton, M. D., Power, P. W., and de Oliveira, C. R. M. (2005). Optimisation based bathymetry approximation through constrained unstructured mesh adaptivity. *Ocean Engineering*, 32(13):1624–1654.
- Pan, X., Jiang, T., Sun, W., Xie, J., Wu, P., Zhang, Z., and Cui, T. (2024). Effective Attention Model for Global Sea Surface Temperature Prediction. *Expert Systems with Applications*, 254:124411.
- Racault, M.-F., Le Quéré, C., Buitenhuis, E., Sathyendranath, S., and Platt, T. (2012). Phytoplankton phenology in the global ocean. *Ecological Indicators*, 14(1):152–163.
- Rasp, S., Dueben, P. D., Scher, S., Weyn, J. A., Mouatadid, S., and Thuerey, N. (2020). Weatherbench: A benchmark data set for data-driven weather forecasting. *Journal of Advances in Modeling Earth Systems*, 12(11).

- Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., et al. (2019). Deep learning and process understanding for data-driven earth system science. *Nature*, 566(7743):195–204.
- Reyes, J. G., Cuervo-Londoño, G. A., and Sánchez, J. (2025). Adaptive meshes in graph neural networks for predicting sea surface temperature through remote sensing. In *Proceedings of the 21st International Conference in Computer Analysis of Images and Patterns (CAIP 2025)*, Lecture Notes in Computer Science (LNCS), Las Palmas de Gran Canaria, Spain. Springer Verlag. Accepted for publication.
- Rice, J., Xu, W. W., and August, A. (2020). Analyzing Koopman Approaches to Physics-Informed Machine Learning for Long-Term Sea-Surface Temperature Forecasting.
- Robinson, A. R. and Lermusiaux, P. F. (2002). *Ocean forecasting: conceptual basis and applications*. Springer, Berlin, Heidelberg.
- Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088):533–536.
- Samsi, S., Gadepally, V., Hurley, M., Jones, M., Kao, E., Mohindra, S., Monticciolo, P., Reuther, A., Smith, S., Song, W., Staheli, D., and Kepner, J. (2017). Static Graph Challenge: Subgraph Isomorphism. In *Proceedings of the 2017 IEEE High Performance Extreme Computing Conference (HPEC)*, pages 1–6, Waltham, MA, USA.
- Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., and Monfardini, G. (2009). The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80.
- Shchepetkin, A. F. and McWilliams, J. C. (2005). The regional oceanic modeling system (roms): a split-explicit, free-surface, topography-following-coordinate oceanic model. *Ocean Modelling*, 9(4):347–404.
- Shearer, C. (2000). The CRISP-DM model: the new blueprint for data mining. *Journal of data warehousing*, 5(4):13–22.
- Shi, X., Chen, Z., Wang, H., Yeung, D.-Y., kin Wong, W., and chun Woo, W. (2015). Convolutional lstm network: A machine learning approach for precipitation nowcasting.
- Sun, T., Feng, Y., Li, C., and Zhang, X. (2022). High precision sea surface temperature prediction of long period and large area in the indian ocean based on the temporal convolutional network and internet of things. *Sensors*, 22(4):1636.
- Team, C. (2023). Croco: Coastal and regional ocean community model. <https://www.croco-ocean.org>. Último acceso: julio de 2025.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, , and Polosukhin, I. (2017). Attention is all you need. *arXiv preprint arXiv:1706.03762*.
- Wang, J., Sun, Z., Yuan, C., Li, W., Liu, A.-A., Wei, Z., and Yin, B. (2024a). Dynamic graphs attention for ocean variable forecasting. *Engineering Applications of Artificial Intelligence*, 133:108187.
- Wang, X., Guo, X., Zhang, H., Luo, J., Jin, W., and Wang, X. (2025). A three-dimensional dynamic spatial-temporal graph neural network for ocean temperature field prediction. *Engineering Applications of Artificial Intelligence*, 127:107304.

- Wang, Y., Zhang, Y. M., and Wang, G. G. (2024b). Impact of Physical and Attention Mechanisms on U-Net for SST Forecasting. *Intelligent Marine Technology and Systems*, 2(1):11.
- Wu, S., Zhang, X., Bao, S., Dong, W., Wang, S., and Li, X. (2023). Predicting Ocean Temperature in High-Frequency Internal Wave Area with Physics-Guided Deep Learning: A Case Study from the South China Sea. *Journal of Marine Science and Engineering*, 11(9):1728.
- Xie, B., Qi, J., Yang, S., Sun, G., Feng, Z., Yin, B., and Wang, W. (2024). Sea Surface Temperature and Marine Heat Wave Predictions in the South China Sea: A 3D U-Net Deep Learning Model Integrating Multi-Source Data. *Atmosphere*, 15(1):86.
- Xie, J., Ouyang, J., Zhang, J., Jin, B., Shi, S., and Xu, L. (2021). An Evolving Sea Surface Temperature Predicting Method Based on Multidimensional Spatiotemporal Influences. *IEEE Geoscience and Remote Sensing Letters*, 19:1–5.
- Yang, H., Li, W., Hou, S., Guan, J., and Zhou, S. (2023). HiGRN: A Hierarchical Graph Recurrent Network for Global Sea Surface Temperature Prediction. *ACM Transactions on Intelligent Systems and Technology*, 14(4):1–19.
- Yang, Y., Dong, J., Sun, X., Lima, E., Mu, Q., and Wang, X. (2018). A cfcc-lstm model for sea surface temperature prediction. *IEEE Geoscience and Remote Sensing Letters*, 15(2):207–211.
- Yu, X., Shi, S., Xu, L., Liu, Y., Miao, Q. S., and Sun, M. (2020). A novel method for sea surface temperature prediction based on deep learning. *Mathematical Problems in Engineering*, 2020:6387173.
- Yuan, T., Zhu, J., Wang, W., Lu, J., Wang, X., Li, X., and Ren, K. (2023). A Space-Time Partial Differential Equation Based Physics-Guided Neural Network for Sea Surface Temperature Prediction. *Remote Sens.*, 15(14):3498.
- Zhang, K., Geng, X., and Yan, X.-H. (2020). Prediction of 3-D Ocean Temperature by Multilayer Convolutional LSTM. *IEEE Geoscience and Remote Sensing Letters*, 17(8):1303–1307.
- Zhang, Q., Wang, H., Dong, J., Zhong, G., and Sun, X. (2024a). Applications of deep learning in physical oceanography: a comprehensive review. *Frontiers in Marine Science*, 11:1396322.
- Zhang, S. Y., Yang, Y. Z., Xie, K. W., Gao, J. H., Zhang, Z. Y., Niu, Q. R., et al. (2024b). Spatial-Temporal Siamese Convolutional Neural Network for Subsurface Temperature Reconstruction. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–16.
- Zhao, X., Qi, J., Yu, Y., and Zhou, L. (2024). Deep learning for ocean temperature forecasting: a survey. *Intelligent Marine Technology and Systems*, 2(1):28.
- Zuo, X., Zhou, X., Guo, D., Li, S., Liu, S., and Xu, C. (2022). Ocean temperature prediction based on stereo spatial and temporal 4-d convolution model. *IEEE Geoscience and Remote Sensing Letters*, 19:1–5.



eii
ESCUELA DE INGENIERÍA
INFORMÁTICA