



UNIVERSIDAD DE JAÉN
Escuela Politécnica Superior (Jaén)

Trabajo Fin de Máster

INFLUENCIA DE LA CALIDAD DE MALLA Y MDE EN LA SIMULACIÓN MULTIFÁSICA 3D DEL FLUJO DEL RÍO GUADALQUIVIR

Alumno/a: Alcázar Vargas, Manuel Gonzalo

Tutor/a: Prof. D. Patricio Bohórquez Rodríguez de Medina
Dpto.: Ingeniería Mecánica y Minera

Noviembre, 2017

Índice

1	Introducción	2
2	Descripción de los datos de entrada disponibles	9
2.1	Estereocorrelación fotogramétrica	9
2.2	Tecnología LiDAR.....	10
2.3	Ventajas e inconvenientes del LiDAR con respecto a la cartografía tradicional .	12
2.4	PNOA	13
2.5	Comparativa MDTs.....	15
2.6	Modificación del MDT: eliminación de puentes	18
3	Descripción detallada del proceso seguido en la generación de la malla	22
3.1	Generación de un archivo STL con los datos del MDT	22
3.2	Generación de la malla a partir del MDT en formato STL	29
4	Parámetros de calidad de malla.....	41
4.1	Skewness	41
4.2	Size change.....	44
4.3	Aspect ratio.....	44
4.4	Ortho Skew.....	45
4.5	Ejemplo de aplicación.....	48
5	Análisis de calidad de malla y comparativa con estudios previos.....	52
5.1	Comparativa de las mallas empleando CheckMesh.....	52
5.2	Comparativa de las mallas empleando ANSYS Fluent	58
5.3	Comparativa de las mallas empleando PointWise	62
5.4	Comparativa de las mallas mediante inspección visual	66
5.5	Análisis global de la calidad de la malla propuesta	70
6	Funcionamiento de un código CFD.....	71
6.1	Pre-procesador.....	71
6.2	Procesador o “solver”	72
6.3	Post-procesador	74
7	Descripción del proceso seguido en la configuración de la simulación	75
8	Resultados de la simulación y comparativa con estudios previos	83
8.1	Software empleado para el postprocesado: EnSight.....	83
8.2	Resultados de la simulación. Q = 500 m ³ /s.....	84
8.3	Comparativa con estudios previos	89

9	Conclusiones	91
10	Anejo 1: Generación de mallas mediante SnappyHexMesh	96
11	Referencias	98

1 Introducción

El presente trabajo fin de máster (TFM en adelante) tiene por objeto la aplicación de sofisticadas técnicas de simulación numérica 3D (modelo tipo Volumen Fluido para flujo agua-aire en régimen turbulento) en el cálculo de avenidas de ríos y delimitación de áreas inundables usando una malla computacional óptima basada en datos LiDAR. Se presentan como retos el mallado de geometrías tipo codo en valles confinados tomando como datos de partida modelos digitales del terreno obtenidos a partir de datos LiDAR (vuelo PNOA 2014) en lugar de ortofotogrametría (PNOA 2011), así como la configuración y ejecución de la simulación CFD correspondiente.

Para ello, se estudia el cauce del río Guadalquivir aguas abajo de la presa de Marmolejo, localidad perteneciente a la provincia de Jaén. Los motivos que llevan a analizar esta zona son diversos.

Esta zona presenta un interés hidráulico destacado; cada cierto tiempo se inunda (Ilustración 1.1), ocasionando importantes daños tanto a la vegetación, como a los edificios colindantes, obras lineales, cortes de carreteras, etc...

Aparte del daño inmediato de la inundación, el transporte de sedimentos, procedentes fundamentalmente de la presa, es muy importante: las márgenes del río se cubren con metros de sedimento procedente de las inundaciones (Ilustración 1.5) estos sedimentos, además, reducen la capacidad de desagüe del cauce, lo que provoca más inundaciones y afectan a la estabilidad de los taludes de las márgenes del río. (Bohórquez y Moral, 2017), (La contra de Jaén, 2017)

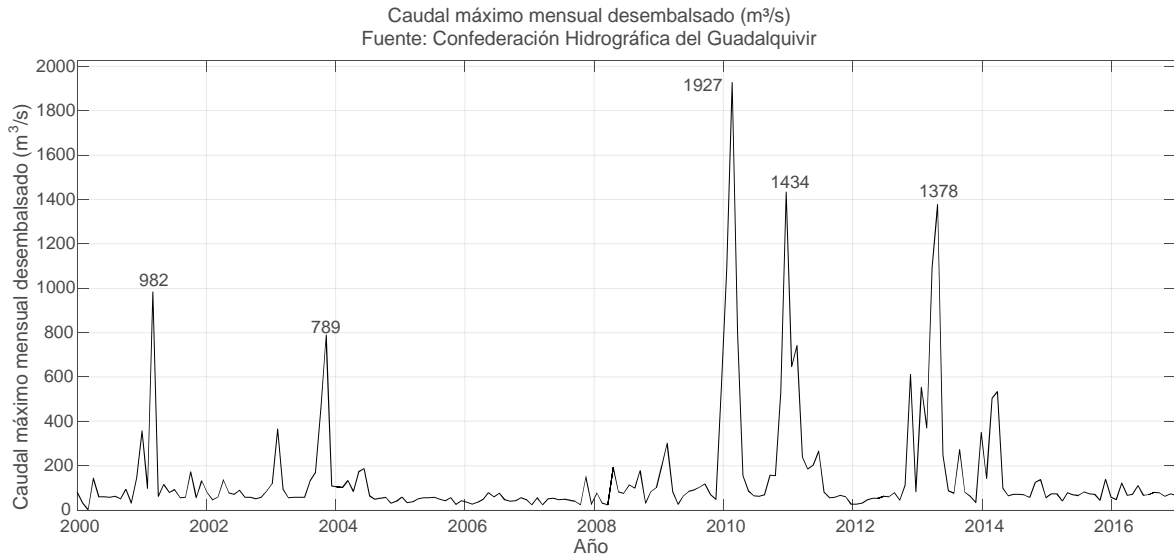


Ilustración 1.1.- Caudal máximo mensual desembalsado en la presa de Marmolejo.

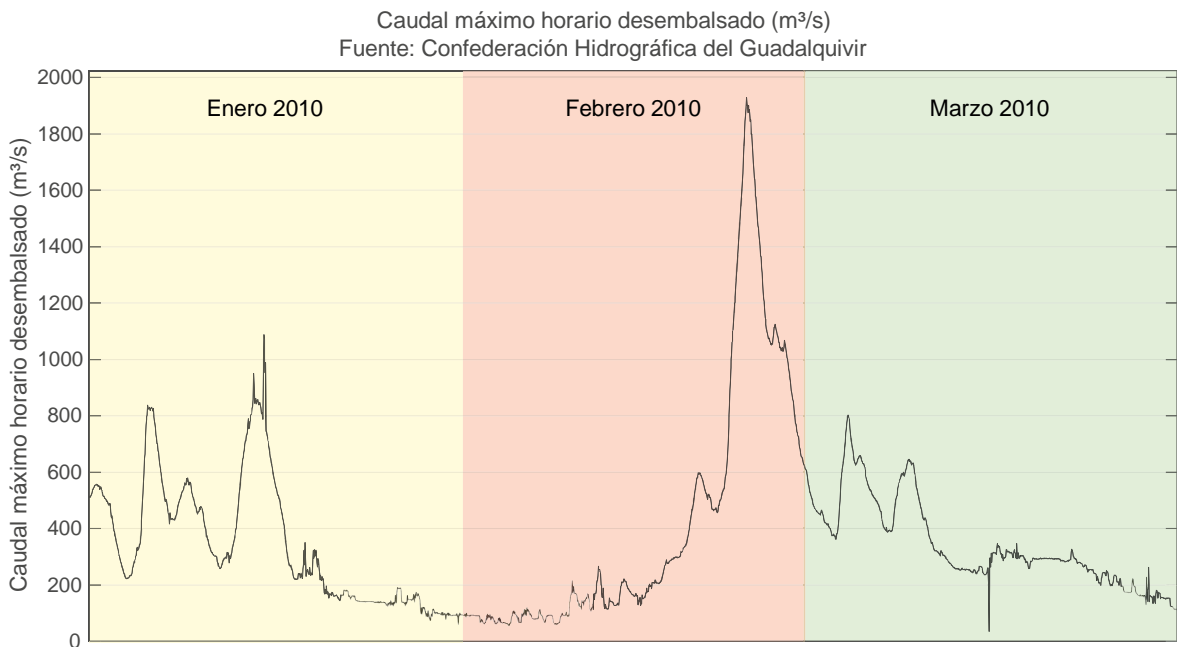


Ilustración 1.2.- Caudal máximo horario desembalsado en la presa de Marmolejo. Enero-marzo de 2010

En la página web del Centro Informático Científico de Andalucía (CICA en adelante), se puede consultar el histórico de ortofotos de la zona, donde se ven los efectos sobre el terreno y la vegetación que ocasionan las inundaciones.

Entre los años 2011 y 2013, se puede apreciar como la vegetación, fundamentalmente olivos, se ha visto afectada, incluso arrancada a causa de la riada que tuvo lugar en el año 2013 (1.378 m³/s).



Ilustración 1.3 Meandro aguas abajo de la presa. Año 2011



Ilustración 1.4 Meandro aguas abajo de la presa. Año 2013

Ampliando la zona del meandro, se observan los sedimentos depositados y los daños provocados (Ilustración 1.5):

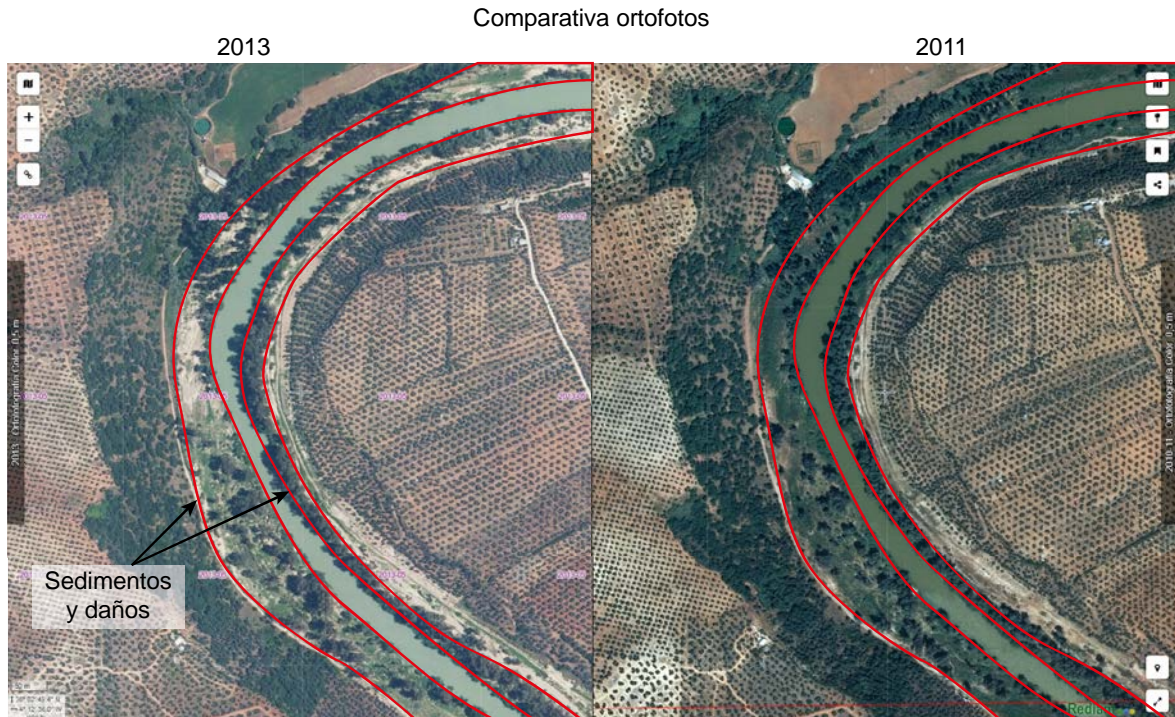


Ilustración 1.5.- Meandro aguas abajo de la presa. Izquierda: 2013. Derecha: 2011

Por otro lado, es una topografía muy compleja, con geometrías tipo codo en valles confinados, lo que supone un reto de cara a la generación de la malla computacional, que se recuerda, es junto con la posterior simulación, el principal objetivo del presente trabajo.

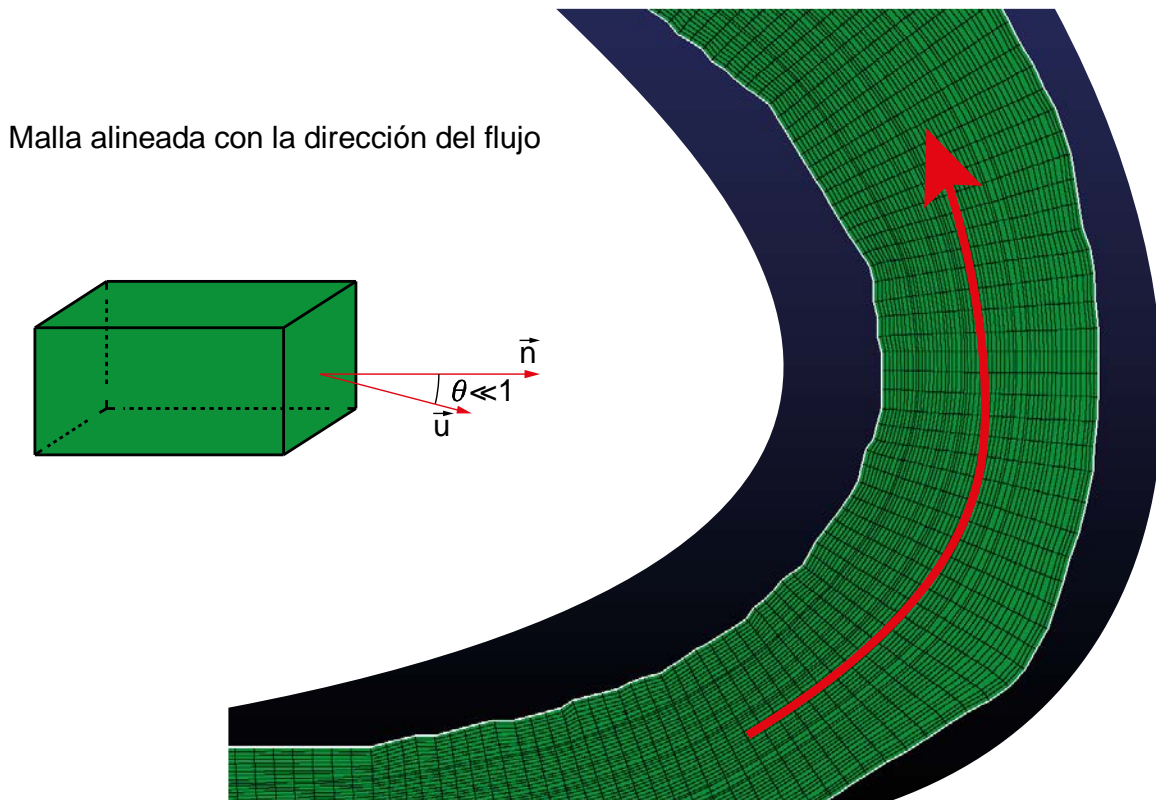
Se presenta también como objetivo secundario un análisis de la calidad del MDT en función de la técnica empleada, a saber: correlación fotogramétrica de imágenes o datos procedentes de vuelo LiDAR. Estos datos proceden de la página web del ministerio de fomento.

En cuanto al reto que supone la generación de una malla computacional óptima la para simulación numérica de una inundación mediante CFD (*Computational Fluid Dynamics*), es interesante destacar varios aspectos.

Por un lado, la generación de una malla tridimensional permite obtener información que mediante mallas uni o bidimensionales no es posible realizar, ya que el movimiento vertical del fluido sólo se captura con mallas tridimensionales. Esta componente de la velocidad, aunque pequeña en magnitud, es determinante

en los fenómenos de transporte de sedimentos. La captura de la capa límite turbulenta en el lecho del cauce es un fenómeno de notable interés.

Para poder capturar la capa límite, se genera una malla *estructurada* de aproximadamente un millón de celdillas de toda la zona de estudio (6 km aguas abajo de la presa de Marmolejo). Las ventajas de una malla estructurada con respecto a una no-estructurada son múltiples, siendo la principal el inmenso control que se tiene sobre la geometría de la malla, especialmente en el interior de la misma.



Este último punto es uno de los más importantes del presente TFM. Con la malla tridimensional estructurada que aquí se propone, genera y analiza; se persigue poder capturar, además de los fenómenos que ya se capturan con mallas bidimensionales, la componente vertical de la velocidad a lo largo del meandro. Esta componente de la velocidad condiciona fuertemente los fenómenos de transporte de sedimentos asociados a la capa límite turbulenta en el cauce. Para ello es crucial que la malla sea suficientemente fina en la zona próxima al terreno.

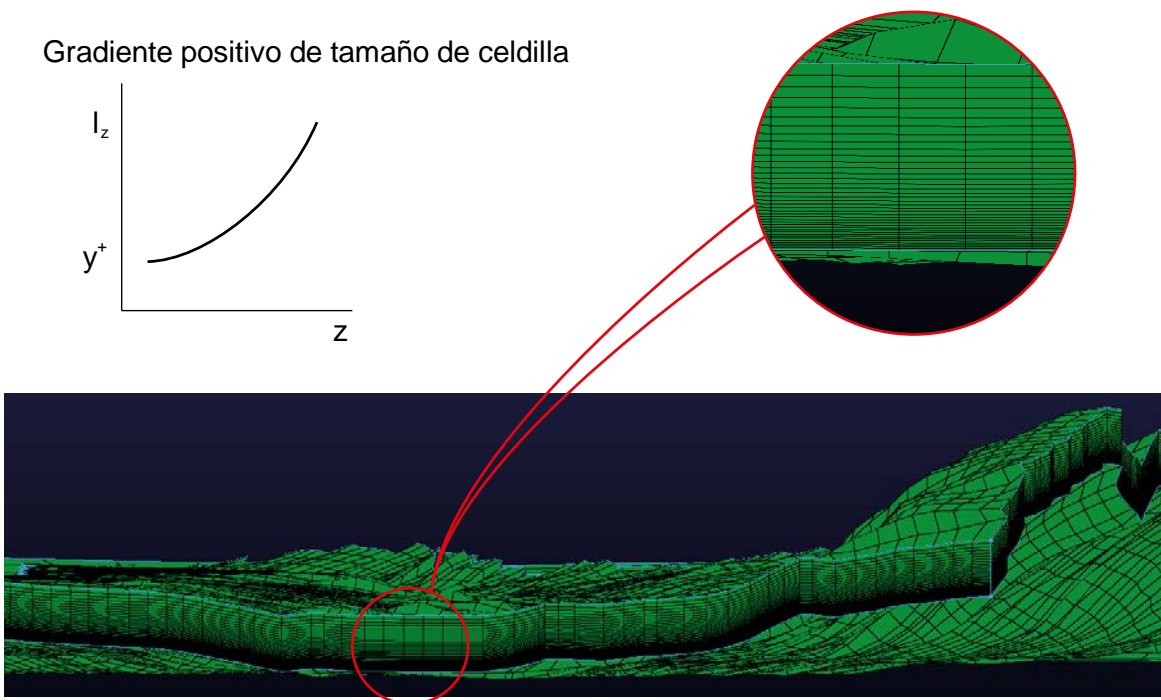


Ilustración 1.7.- Gradiente positivo del tamaño vertical de celdilla

De hecho, hoy en día para la gran mayoría de las simulaciones CFD que se realizan, el fenómeno de capa límite se captura con mallas estructuradas muy refinadas en las condiciones de contorno tipo *wall* (Ilustración 1.8)

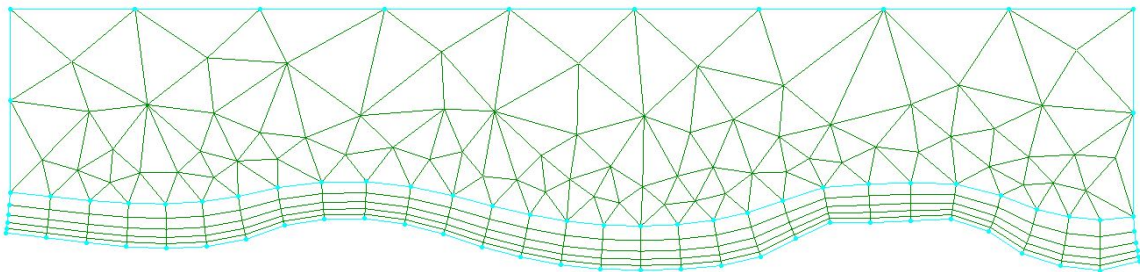


Ilustración 1.8.- Malla híbrida con zona estructurada en la capa límite.

Además del refinamiento de la malla en las proximidades del cauce, la alineación de las celdillas con la dirección del flujo, o más concretamente, el vector normal de cada cara con el vector velocidad del flujo que atraviesa dicha cara, mejora muchísimo la convergencia y la precisión de los métodos numéricos de simulación, además de disminuir drásticamente la difusión numérica. Se recuerda que el flujo convectivo de una magnitud fluida (ϕ) que atraviesa una superficie Σ_0 viene dado por la expresión:

$$\int_{\Sigma_0} \phi(\bar{v} \cdot \bar{n}) d\sigma \quad (1)$$

Por lo que, debido a la naturaleza de los métodos numéricos empleados en CFD, cuanto menor sea el ángulo que forman los vectores normal (\bar{n}) y velocidad (\bar{v}), mucho más rápido convergerá la simulación y mayor será la precisión.

Otra ventaja añadida de las mallas estructuradas frente a las no estructuradas es que requieren mucha menos memoria en la computadora, puesto que la conectividad de celdillas se define implícitamente en la malla, algo que no ocurre con las mallas no estructuradas.

Por último, existen multitud de algoritmos de CFD que son aplicables exclusivamente a mallas estructuradas, por lo que el abanico de herramientas con este tipo de mallas es más amplio.

Aparte de la generación de la malla, también es objeto de estudio del presente trabajo la simulación por CFD de la misma, y la comparación de los resultados con estudios previos. Es necesario analizar los métodos numéricos que se emplean en este tipo de simulaciones, así como los parámetros más importantes que gobiernan el flujo para poder configurar la simulación de modo que converja adecuadamente, con un coste computacional y una precisión asumibles.

2 Descripción de los datos de entrada disponibles

Tal y como se ha descrito en el objetivo del presente trabajo fin de máster, es crucial disponer de un adecuado modelo digital del terreno (MDT en adelante) para poder realizar la malla computacional de la zona de estudio, que es uno de los objetivos principales de este trabajo.

Se entiende por MDT un conjunto de puntos, estructurados en forma de malla, que tienen como propiedad la altitud respecto al nivel medio del mar (ortométrica). El parámetro que caracteriza a los MDT es el “paso de malla”, o más concretamente la “densidad” de la malla. El primero es la distancia promedio que hay entre dos puntos, mientras que el segundo es el número de puntos por metro cuadrado. La calidad del MDT generalmente viene determinada por la utilización que se requiera del modelo: modelos hidrológicos, generación de ortofotografías, etc... Actualmente los MDT pueden llegar a tener densidades superiores a los 50 puntos/m²

Existen fundamentalmente dos técnicas para la generación de MDT: a partir de un vuelo fotogramétrico o LiDAR, o calculado a partir de la cartografía ya existente mediante técnicas de postprocesado de los datos.

2.1 Estereocorrelación fotogramétrica

A partir de los datos de un vuelo fotogramétrico, para el levantamiento de los MDTs las técnicas que se utilizan están basadas en la correlación automática de imágenes digitales y la posterior depuración de los MDTs generados. La ortofotografía digital de Andalucía ha sido generada siguiendo esta metodología. Los pasos para la generación de MDTs según esta técnica son, *grosso modo*, los siguientes:

1. Generación del TIN. Son necesarios puntos distribuidos aleatoriamente, de los cuales se conozca la cota, así como líneas de ruptura reintegradas en la ortofotografía con objeto de obtener la geometría del MDT. Para obtener dichos puntos, se emplean curvas de nivel, caminos, líneas de ferrocarril y de carretera, sendas, etc...

2. Se coloca el primer punto de la malla (grid), habitualmente en el punto situado más al suroeste.
3. Se determina el paso de malla, que normalmente toma el mismo valor, tanto en X como en Y. Se expresa en metros
4. Se calcula la altitud respecto al nivel medio del mar de cada punto (cota). El proceso seguido para el cálculo de la cota, cuando los datos se obtienen de un vuelo LiDAR se explica más adelante.

Debido a la naturaleza del MDT, es necesario realizar un mantenimiento del mismo, pues algunas zonas pueden ver modificada su cota debido a multitud de circunstancias. Esto es de especial importancia si se pretende convertir una foto a una ortofoto empleando el MDT: un error en la cota puede deformar sustancialmente la ortofoto posterior:

- Debido a la magnitud de las obras lineales (autovías, ferrocarriles, etc..) así como otras grandes obras públicas, es necesario la captura de nuevas líneas de rotura, así como de las nuevas estructuras construidas
- A causa del solape que tiene lugar a la hora de generar MDT, especialmente los datos procedentes de diferentes pasadas de un vuelo LiDAR, es importante corregir cualquier error que pudiese aparecer al combinar dichos MDTs.
- Las cotas negativas deben ser corregidas
- Existirán zonas en las que no haya puntos, debido fundamentalmente a errores en la toma de datos. Las cotas de dichas zonas deberán obtenerse a partir de interpolación de los datos de los alrededores

2.2 Tecnología LiDAR

El LiDAR (*Light Detection And Ranging*) es un sistema de medida, compuesto por un emisor de pulsos láser y un receptor de los mismos. Este aparato se instala en un avión desde el que se realiza el vuelo LiDAR. El funcionamiento del sistema se describe a continuación.

El emisor de pulsos láser emite ondas que llegan a la superficie terrestre y vuelven al avión. El tiempo que tardan dichas ondas en hacer el viaje de ida y vuelta se mide con mucha precisión. Puesto que se conoce la velocidad de propagación de las ondas en el aire, es posible determinar la distancia que existe entre el avión y el terreno. Además, según los pulsos láser reboten de una u otra manera, el terreno se puede clasificar en:

- Agua: el rayo láser es absorbido y no se refleja nada. No hay puntos en dicha zona
- Superficie sólida: el rayo láser se refleja correctamente. Este sería el caso de edificios, rocas, carreteras, tableros de puentes, etc...
- Vegetación: parte del rayo se refleja, y parte atraviesa la vegetación, reflejándose en el suelo. En este caso se tendrían dos medidas, y ambas son almacenadas (Ilustración 2.1).



Ilustración 2.1.- Toma de datos del LiDAR. Fuente: INCLAM

El sistema no solo toma un punto en la vertical, sino que hace un “barrido” con un cierto ángulo (Ilustración 2.2). Esto permite reducir considerablemente el número de pasadas que debe realizar el avión para medir una determinada área.

De este modo, se obtiene una nube de puntos y una clasificación entre ellos, a saber:

- Terreno base
- Vegetación
- Edificios

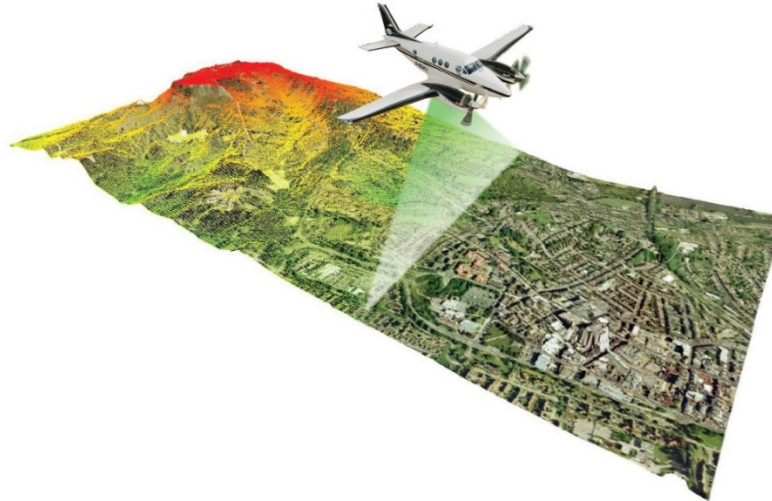


Ilustración 2.2.- Esquema de funcionamiento de un vuelo LiDAR (blog.tecnoceano.com)

Habitualmente será necesario un cierto solape entre los vuelos para poder unir las nubes de puntos y poder elaborar un MDT adecuado. Se suele emplear como parámetro indicador de la resolución del LiDAR el número de puntos medio por metro cuadrado. La resolución (o densidad) suele variar desde 0,33 puntos/m² en vuelos de avión a alta distancia hasta 50 puntos/m² en vuelos de helicóptero. Es necesario llegar a un equilibrio en la densidad de puntos. Por un lado, el obtener una nube de puntos muy densa encarece considerablemente el estudio, mientras que una nube poco densa reduce la calidad del MDT notablemente. Para el Sistema Nacional de Cartografía de Zonas Inundables se recomienda emplear la densidad que aporta el LiDAR PNOA del IGN.

Por último, el formato para almacenar y transmitir el MDT puede ser:

- Una malla en la que el atributo de cada celda es la altitud (grid)
- Un fichero ASCII, de texto plano, con las coordenadas X, Y, Z para cada punto
- Un formato CAD, como puede ser .stl

2.3 Ventajas e inconvenientes del LiDAR con respecto a la cartografía tradicional

El LiDAR presenta ventajas muy claras en lo que a la determinación de zonas inundables se refiere. Sin embargo, como se muestra en la Tabla 1, también cuenta con algunas desventajas que pueden obligar al empleo de los datos LiDAR junto a la topografía clásica.

Tabla 1.- Ventajas e inconvenientes de la tecnología LiDAR frente a la cartografía tradicional.
Fuente: guía metodológica para el desarrollo del SNCZI

Ventajas	Desventajas
<ul style="list-style-type: none"> ▪ Mayor precisión en cotas ▪ Homogeneidad de resultados ▪ Facilidad de manejo para la obtención de datos para la modelización hidráulica ▪ Reducción considerable del trabajo de delineación posterior al cálculo hidráulico ▪ Menos coste 	<ul style="list-style-type: none"> ▪ Necesidad de realización de batimetrías en aquellos casos en que los cauces lleven agua ▪ Falta de información asociada (toponimia, etc...) ▪ Dificultad de cartografiar superficies cubiertas de vegetación muy densa, lo que puede dar lugar a errores en MDTs finales ▪ Fallos de precisión en la delineación de líneas límites o aristas visibles en imágenes fotográficas (motas, orillas, etc...)

2.4 PNOA

El Plan Nacional de Ortofotografía Aérea (PNOA en adelante) tiene como objetivo generar un modelo de todo el territorio español mediante vuelos LiDAR aerotransportados. El modelo será una nube de puntos (X, Y, Z), con una resolución promedio de 0,5 puntos/m². En cuanto al error cuadrático medio en la vertical (RMSE_z) se estima en 0,20 m. Estos modelos o bases de datos, no solamente sirven para la generación de MDTs, también sirven para generar MDSs, que incluyen edificaciones y demás obras; realizar estudios de inundabilidad, detección de nuevos edificios, aplicaciones en telecomunicaciones (cobertura de antenas), y un largo etcétera.

La distribución de los datos se realiza a través del centro de descarga del centro nacional de información geográfica (CNIG en adelante). Los archivos, con extensión .laz, cubren cada uno una extensión de 2 x 2 km. Los puntos proceden de vuelos LiDAR, y se han clasificado de forma automática, coloreándose, en RGB, cada punto según las ortofotos del PNOA. El tamaño de pixel se encuentra comprendido entre 25 y 50 cm. En cuanto al sistema geodésico de referencia, en la península Islas Baleares, Ceuta y Melilla es ETRS89; mientras que en las Islas

Canarias es REGCAN95, siendo ambos sistemas compatibles con WGS84 y proyección UTM, según el huso al que corresponda cada fichero. En cuanto a la cota Z, las altitudes son ortométricas

La malla computacional que se genera en el presente trabajo toma como datos de partida los ficheros .LAZ que ofrece el Ministerio de Fomento.

Según establece la guía metodológica para el desarrollo del SNCZI, la precisión del LiDAR viene condicionada por dos factores: la precisión durante la toma de datos y la precisión durante el procesado de los datos. A continuación, se desarrolla cada una de ellas.

Precisión durante la toma de datos

Fundamentalmente viene condicionada por dos factores:

- Sistema de Navegación Inercial (INS): es indispensable conocer la posición y *orientación* del sensor LiDAR. Esta orientación viene determinada por el INS. Como cualquier instrumento de medida, presenta un error que debe ser tenido en cuenta
- GPS diferencial: Determina la posición del sensor LiDAR y, al igual que el anterior instrumento, tiene un error.

Además, de la precisión durante la toma de datos, también es una fuente de errores el procesado de los mismos

Precisión aportada por el procesado de los datos de campo

Existen multitud de algoritmos que permiten generar el MDT a partir de los datos de campo, teniendo algunas empresas sus propios algoritmos. Se podrían dividir estos algoritmos en dos bloques:

- Los que mantienen la cota de los puntos de campo y, mediante interpolación, obtienen la cota de puntos interiores

- Los que tan solo emplean puntos que minimicen el error de la interpolación, modificando, si procede, la cota de algunos puntos de campo

Así, a partir de los mismos datos de campo se pueden generar distintos MDTs, en función del algoritmo empleado. La importancia del método radica en la captura de estructuras singulares: aristas de edificios, muros, etc...

2.5 Comparativa MDTs

A continuación, se procede a describir, de un modo cualitativo las diferencias entre los MDTs obtenidos con las tecnologías anteriores, a saber: estereocorrelación fotogramétrica (2011) y vuelo LiDAR (2014). Se procede de la siguiente manera para ambos casos:

1. Se muestran (diversos) cortes de la sección de estudio y se superponen los perfiles del terreno. (Ilustración 2.4)
2. Se muestra un MDT tridimensional, representado mediante el software *Global Mapper*, re-escalado en el eje Z por motivos ilustrativos. (Ilustración 2.5)

Donde las imágenes 3D se obtienen desde la misma ubicación; así como los cortes 2D tienen las mismas coordenadas de inicio y fin.



Ilustración 2.3.- Ortofotografía de la zona de estudio.

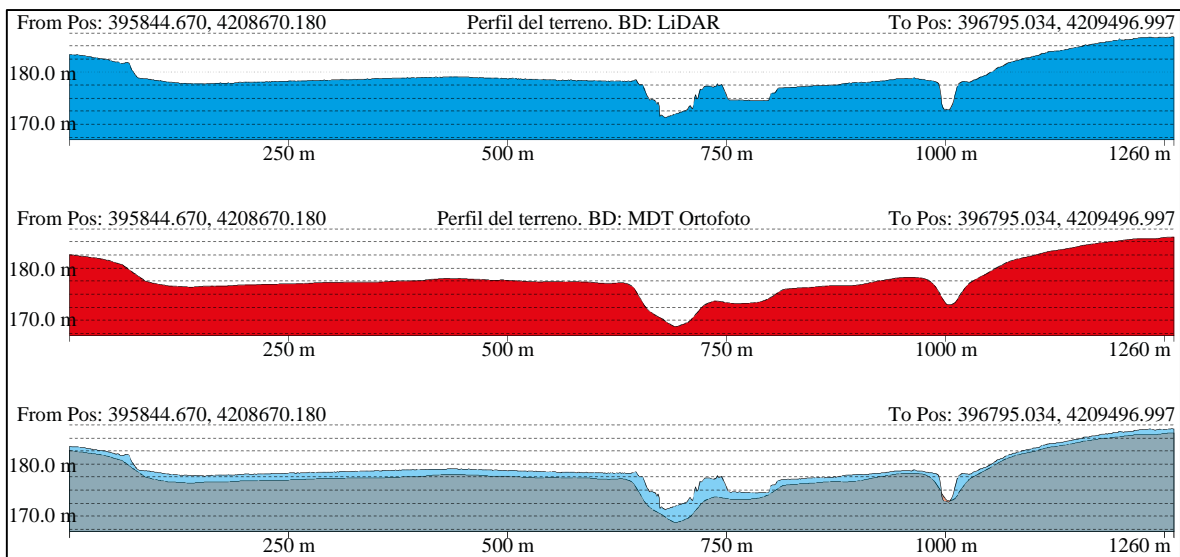
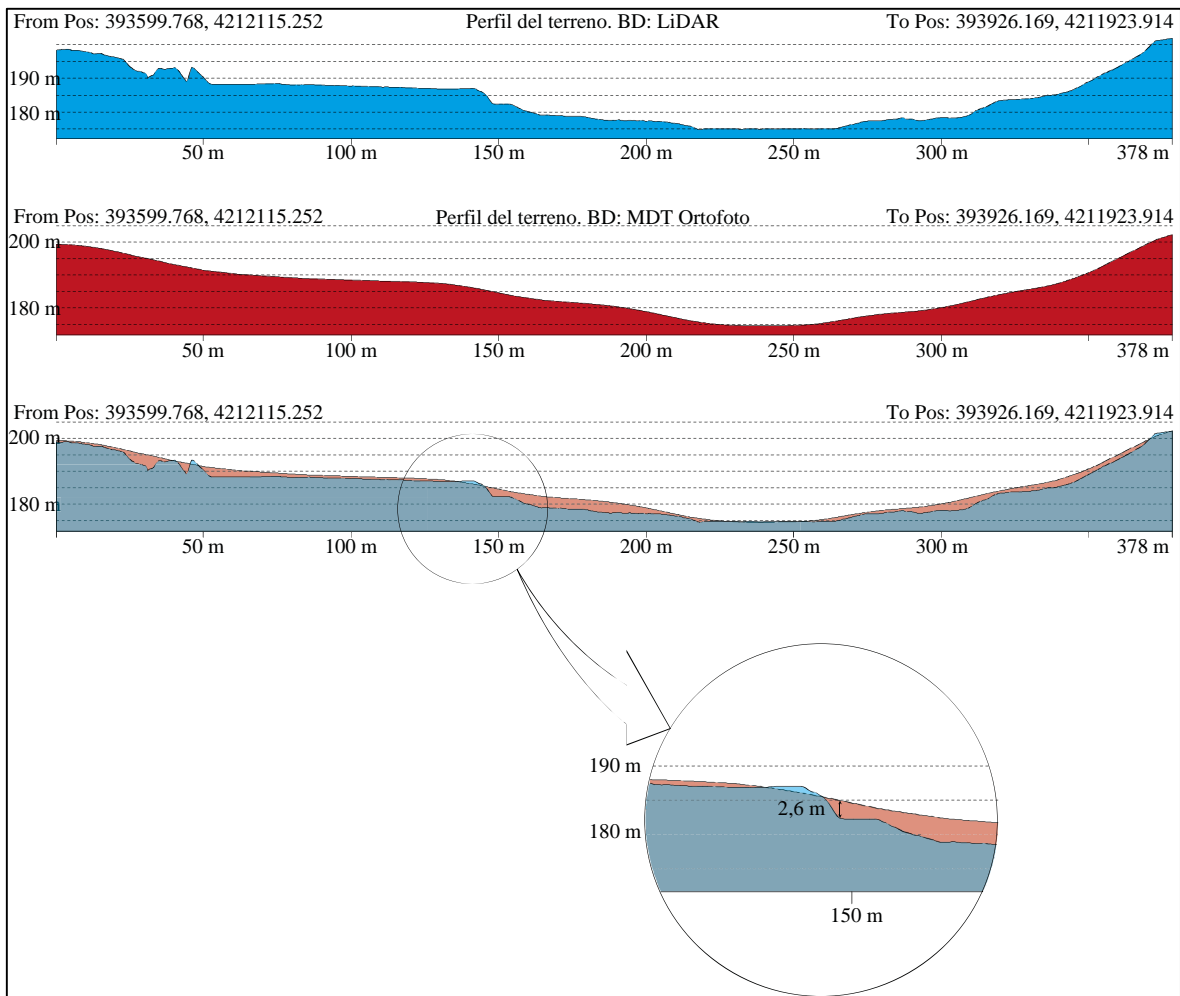


Ilustración 2.4.- Comparativa perfiles del terreno. Superior: Sección A-A'. Inferior: Sección B-B'. Cortes referidos a la Ilustración 2.3

A continuación, se muestra la vista 3D de la zona de estudio, según diferentes bases de datos

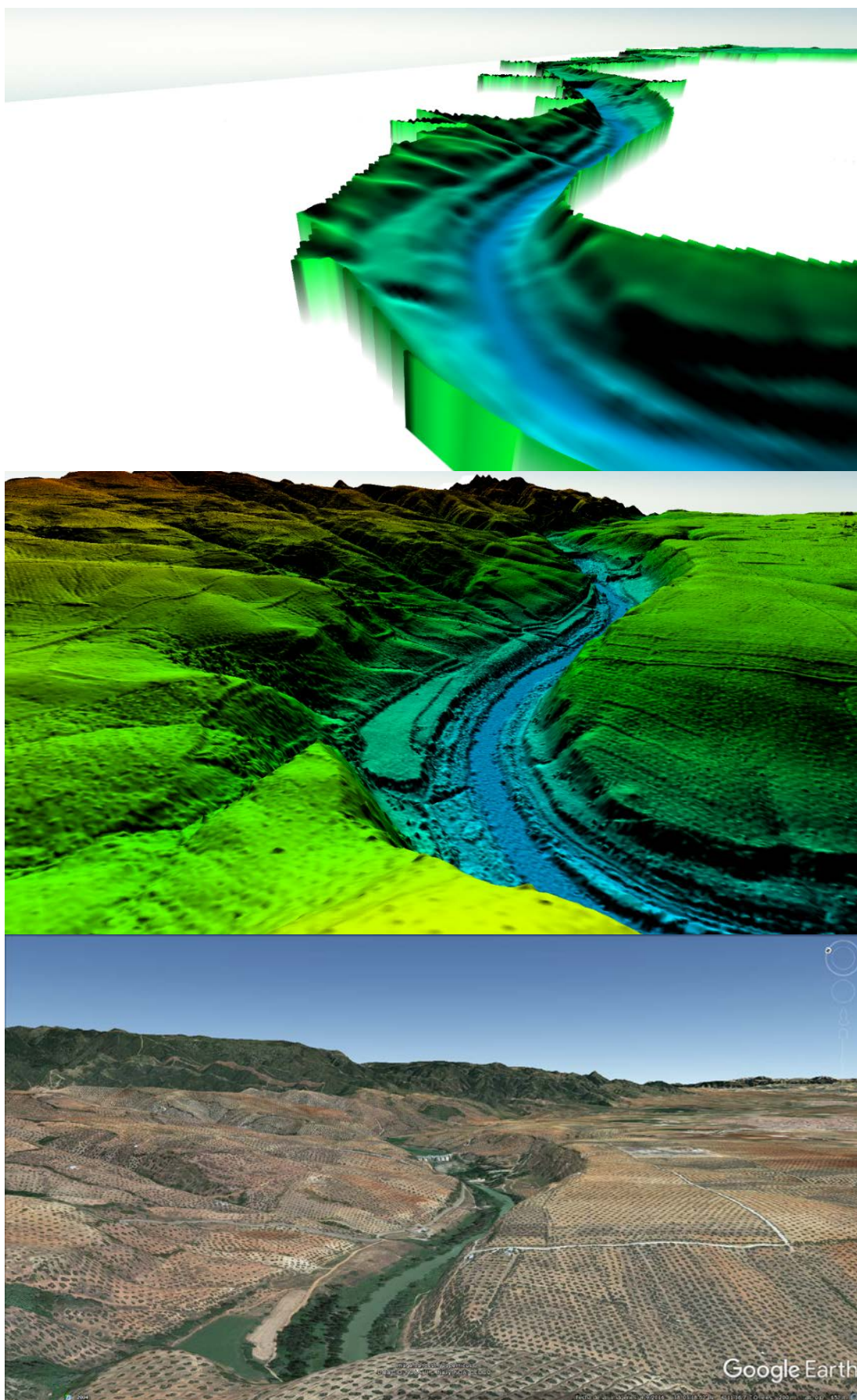


Ilustración 2.5.- Vista 3D de la zona de estudio. Superior: BD: correlación fotogramétrica. Medio: BD: LiDAR. Inferior: BD: Google Earth

De las ilustraciones 2.3, 2.4 y 2.5 se desprenden varias conclusiones:

1. El MDT obtenido mediante correlación fotogramétrica es mucho más suave. Esto es debido a que, al ser la densidad de la malla mucho menor, se pierden multitud de detalles del relieve. Además, la interpolación cúbica homogeneiza el perfil del terreno. El vuelo LiDAR, por el contrario, toma muchos más puntos por metro cuadrado, de modo que consigue capturar la orografía con mayor detalle.
2. Aunque el *error medio cuadrático* (RMEZ) del MDT realizado a partir de las ortofotos es de dos metros, se detectan desviaciones de casi tres metros. Esto no significa que, si el RME es de dos metros, no pueda haber desviaciones superiores esta distancia. Lo que indica esta manera de estimar el error es que la *desviación cuadrática media* de la magnitud considerada es de 2 metros. De este modo, se define el error según:

$$\text{RMSE}_z = \sqrt{\frac{\sum_{j=1}^n (z_j - \langle z \rangle)^2}{n}} \quad (2)$$

Donde se puede observar que el error medio cuadrático no es más que la variación esperada con respecto al valor “correcto”.

3. La variación entre ambos MDTs puede ser debido, a una variación en el perfil del terreno durante los tres años que transcurren desde una toma de datos a otra (2011 – 2014), ya que el transporte de sedimentos en las inundaciones que tienen lugar es importante.
4. En el propio cauce del río, las variaciones en la cota son debidas, fundamentalmente, a la variación del caudal instantáneo del río Guadalquivir entre la toma de ambos datos (2011 – 2014). Para corregir este fenómeno sería necesario realizar una batimetría del río, lo cual se sale del alcance del presente estudio.

2.6 Modificación del MDT: eliminación de puentes

El tramo objeto de estudio presenta un puente aguas abajo de la presa de Marmolejo. Este puente no ha sido detectado como tal en el postprocesado de los

datos LiDAR, por lo que debe eliminarse según establece la guía metodológica para el desarrollo del SNCZI. En dicho documento se establece el procedimiento para la eliminación de los puentes:

1. *Identificación de las zonas que son candidatas a ser clasificadas como puentes o pasos elevados.*
2. *Comprobación de si realmente se trata de un puente- paso elevado o, de lo contrario, se trata de un falso positivo.*
3. *En el caso de tratarse de un puente, retoque manual de la parte del mismo que se quiere eliminar (zona marcada con círculos en la Ilustración 2.7).*
4. *Eliminación de las alturas de los puntos del MDS correspondientes a la zona que se haya seleccionado como puente. Si se interpolan directamente los puntos que se han borrado, habrá problemas con la calidad del MDT obtenido porque también se expandirán las alturas de la parte alta del puente hacia la zona eliminada. Para evitar este problema, será necesario definir líneas de rotura por medio de puntos que representen geoméricamente la estructura (desde los puntos azules hasta los rojos en el puente principal de la Ilustración 2.7). Estos puntos se seleccionan de forma visual, con la seguridad de que pertenecen al punto más bajo del puente. Posteriormente, sobre el MDS al que se le han eliminado los puntos del puente, se dibujan las líneas de rotura desde el valor de altura de los puntos rojos hasta el valor de altura de los azules.*
5. *Generación del MDT final.*



Ilustración 2.6.- Puente aguas abajo de la presa de Marmolejo. Fuente: jaendesdemiatlaya.blogspot.com.es



Ilustración 2.7.- Detección y eliminación de puentes, pasos elevados y obras de drenaje. Fuente (INCLAM)

Aunque la descripción del proceso seguido para la eliminación del puente se realiza en apartados posteriores, es importante destacar la importancia de este elemento, pues afecta drásticamente al presente trabajo fin de máster por dos motivos:

1. Para la generación de una malla estructurada es necesario que la geometría no presente cambios bruscos de curvatura (aristas vivas) pues las celdillas computacionales se deformarían excesivamente, con todos los problemas de calidad de malla que ello implica, tal y como se describirá en el capítulo correspondiente.
2. La posterior simulación por CFD del proceso de inundación arrojaría resultados inválidos, puesto que la geometría del puente que captura la malla es totalmente distinta de la geometría real del mismo (Ilustración 2.9)

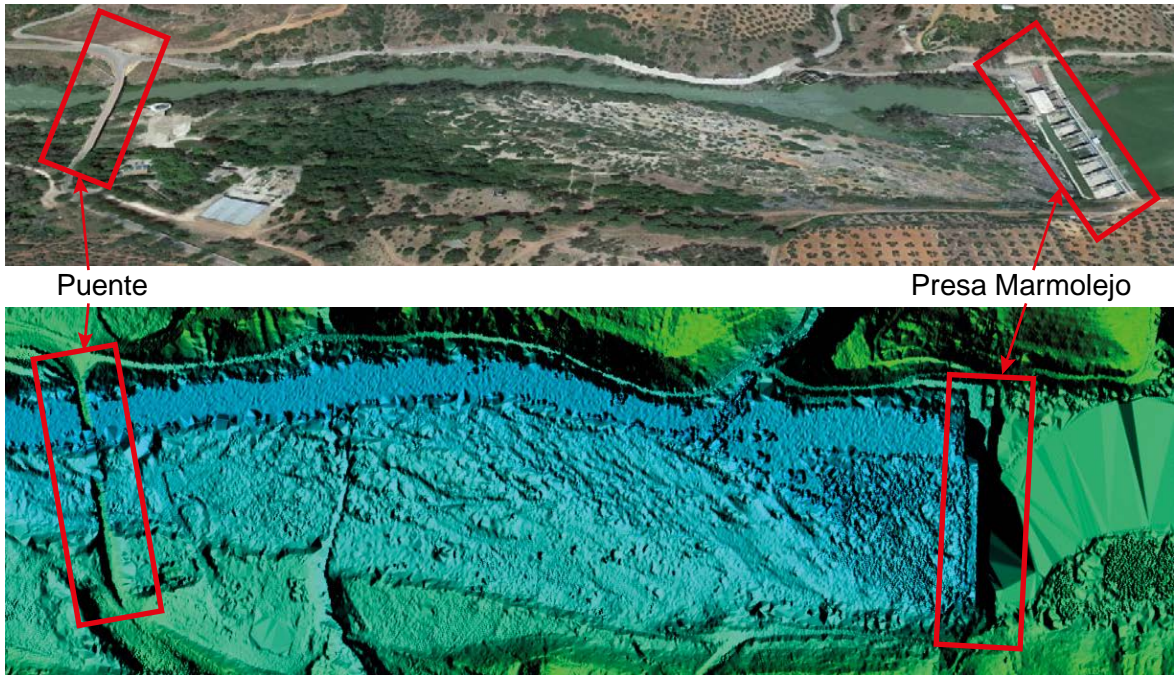


Ilustración 2.8.- Superior: Imagen de Google Earth donde se observa el puente. Inferior: Datos LiDAR de la zona, donde se aprecia que no se ha detectado correctamente el puente.

A continuación, se muestra un esquema comparativo donde se puede ver un modelo simplificado del puente real, así como la geometría que captura el MDT bruto. Se observa que el flujo a través del mismo será totalmente diferente si se modela de un modo o de otro.

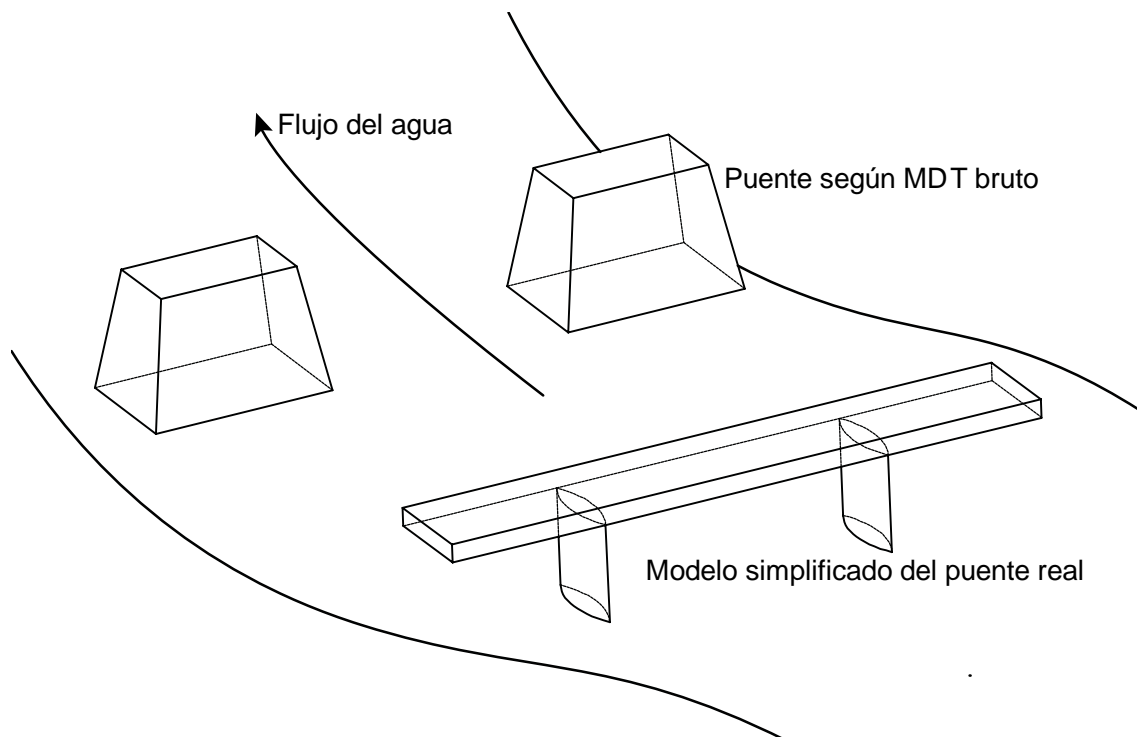


Ilustración 2.9.- Esquema de los modelos tridimensionales del puente. Izquierda: puente modelado según datos brutos del MDT. Derecha: Esquema tridimensional simplificado del puente real.

3 Descripción detallada del proceso seguido en la generación de la malla

3.1 Generación de un archivo STL con los datos del MDT

El primer paso para la generación de la malla consiste en obtener el MDT de la zona de estudio. Éste se obtiene de la página web del ministerio de fomento: <http://centrodedescargas.cnig.es/CentroDescargas/catalogo.do?Serie=LIDAR#>

Dentro de esa web, y dado que el objetivo del presente TFM es generar la malla a partir de datos LiDAR, seleccionamos para descargar los datos LiDAR. Éstos están en formato .laz, indicándose la información altimétrica de la nube de puntos LiDAR. Estos puntos conforman un modelo digital de superficie (MDS), que no es exactamente igual que un modelo digital del terreno (MDT). En un MDT, la información que se representa es la superficie de la *Tierra*, mientras que en un MDS se representa la Tierra *y todo lo que hay sobre ella*: puentes, edificios, vegetación, etc...

Buscamos por mapa > por punto > por polígono (Ilustración 3.1):

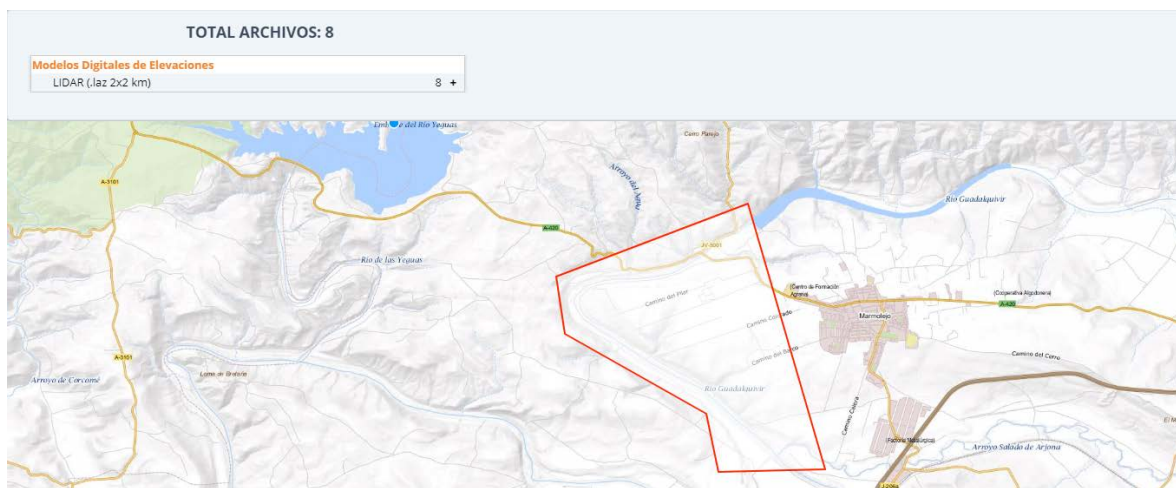


Ilustración 3.1.- Polígono que incluye la zona de estudio

Y se descargan ocho archivos .laz. Una vez descargados, se abren con global mapper y se seleccionan los datos del terreno y del agua (Ilustración 3.2).

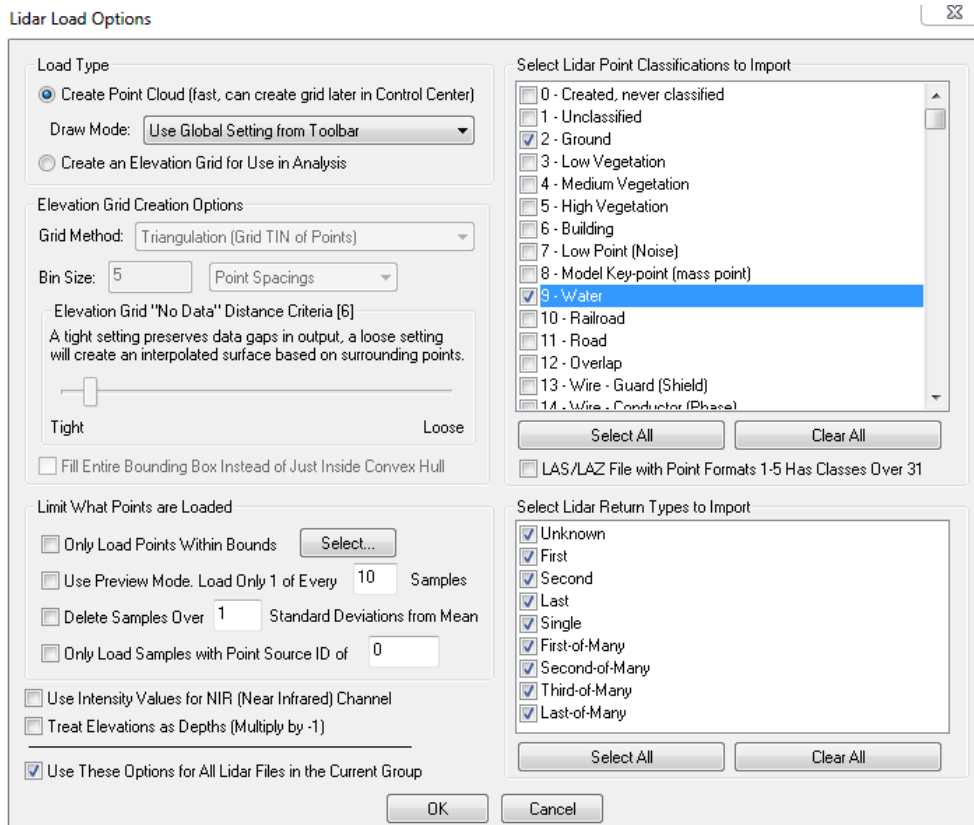


Ilustración 3.2.- Parámetros que se seleccionan a la hora de importar los datos LiDAR

Obteniéndose los puntos:

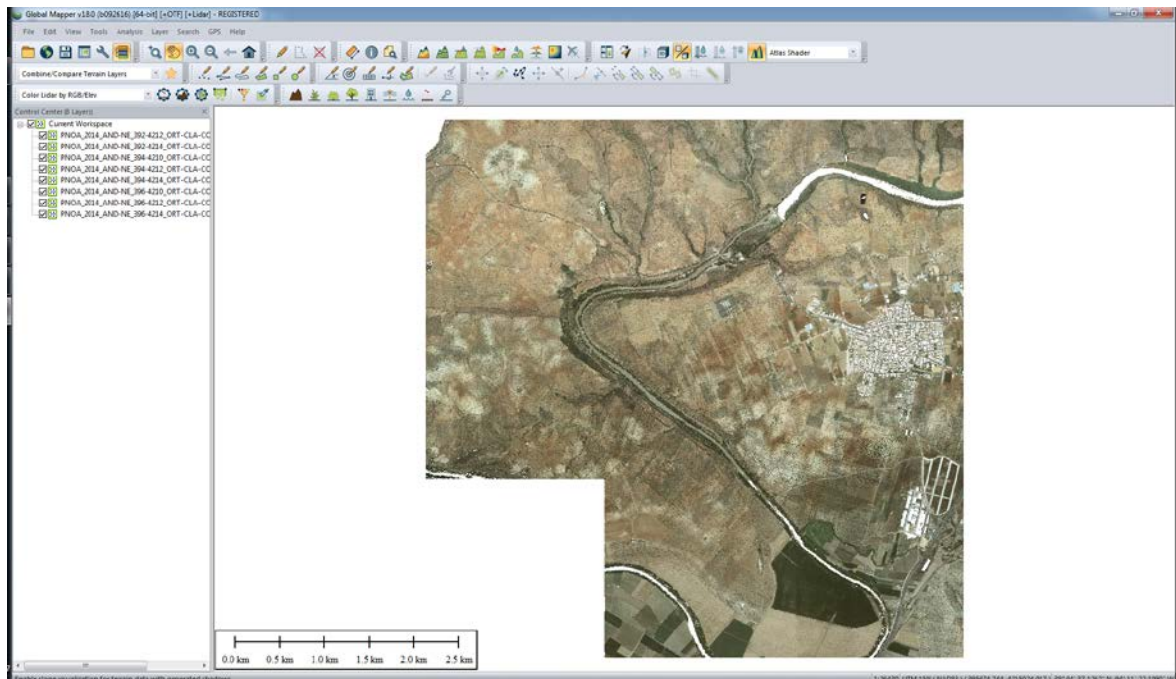


Ilustración 3.3.- Datos LiDAR importados en Global Mapper

Es importante destacar que lo que se representa no es una ortofoto, sino los puntos LiDAR coloreados. Cada punto tiene información altimétrica y color. La información altimétrica incluye la cota, el tipo de rebote del láser y otros parámetros.

El color se obtiene de la ortofoto de la misma zona. De ahí que parezca una ortofoto, pero si se amplía cualquier zona, se observa que es una nube de puntos coloreada:

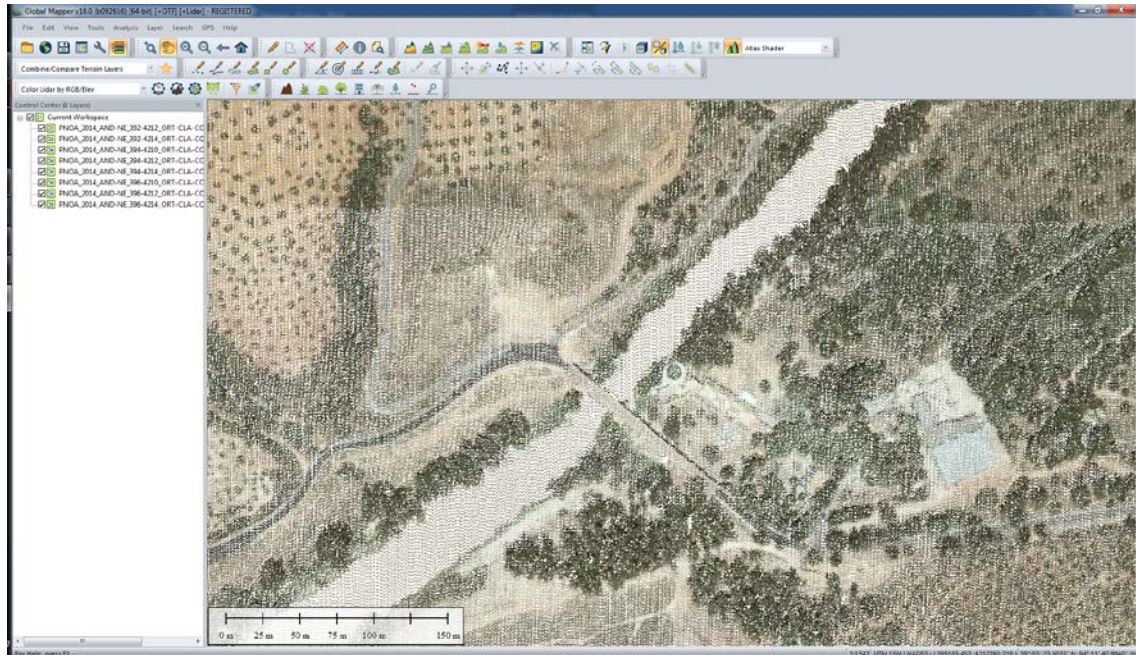


Ilustración 3.4.- Zoom de la zona para observar que no es una ortofoto, sino puntos LiDAR coloreados según los colores de la ortofoto

El siguiente paso es exportar los datos .laz a un archivo de elevaciones para poder manejarlo y editarlo en Global Mapper. Así, crearemos un archivo de elevaciones: *Analysis > Create elevation grid from 3D vector/lidar data*. Seleccionamos todas las capas y le indicamos al programa que tome como paso de malla 5 m:

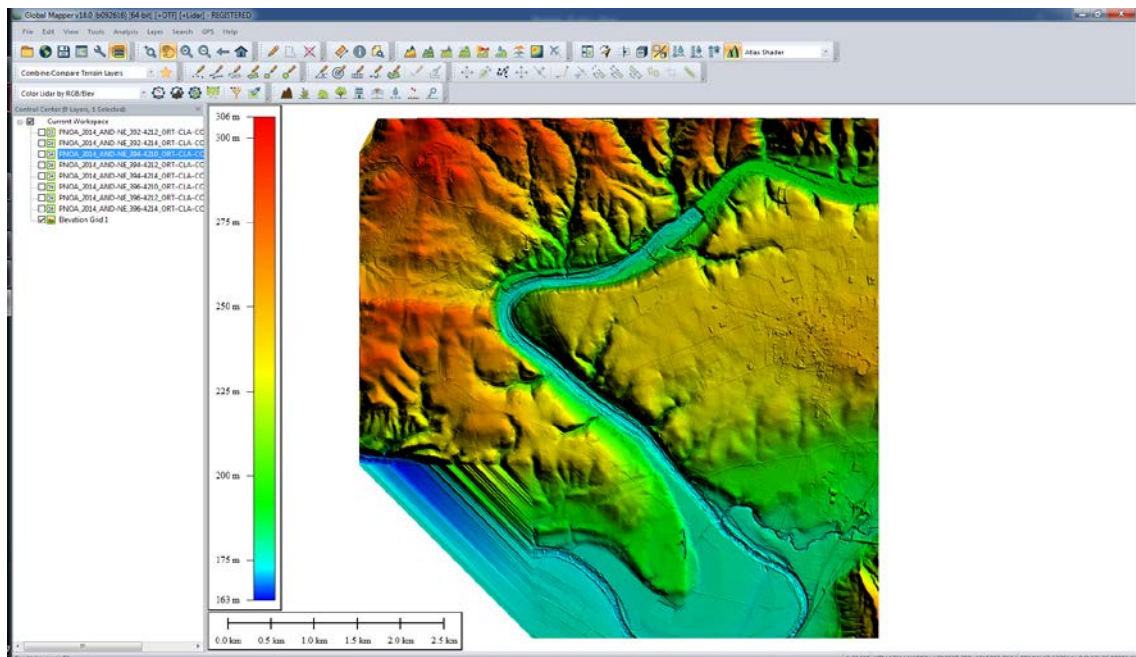


Ilustración 3.5.- MDT coloreado según la cota de los puntos

Dado que la Guía metodológica para el desarrollo del sistema nacional de cartografía de zonas inundables recomienda considerar tan solo la superficie de la tierra (datos MDT) en la determinación de zonas inundables, deberemos filtrar los datos LiDAR, eliminando, en este caso concreto, un puente que cruza el Guadalquivir aguas abajo de la presa.

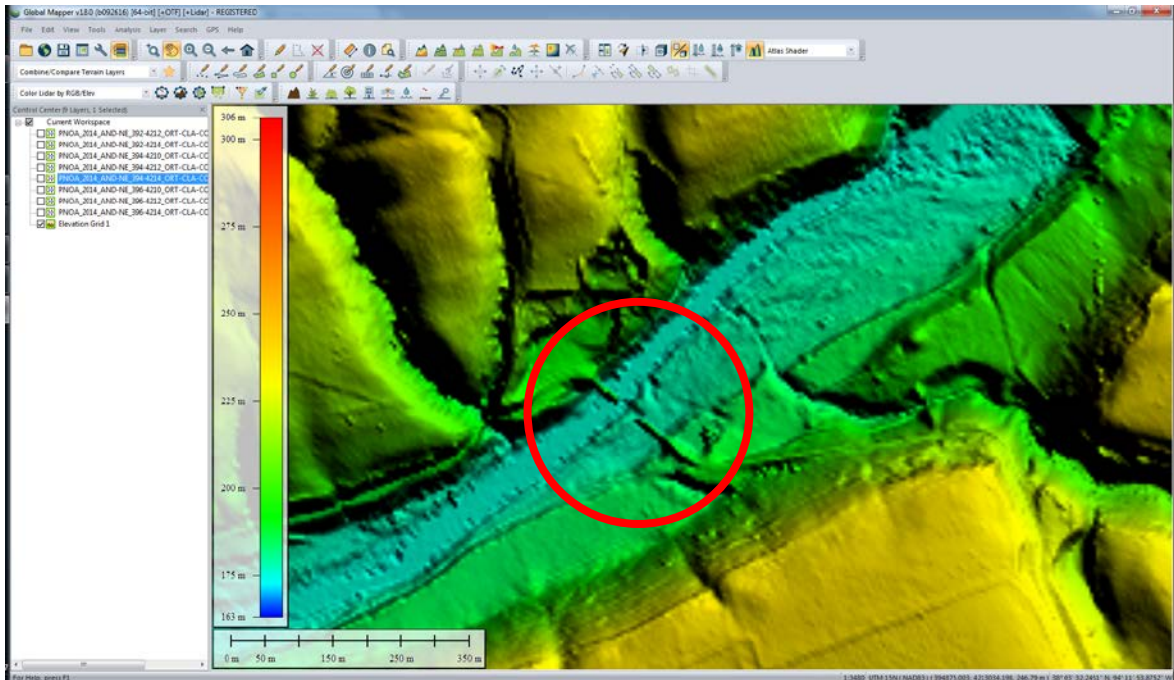


Ilustración 3.6.- Zoom del puente que es necesario eliminar

Lo que ocurre es que los datos LiDAR que proporciona el Ministerio no han detectado todo el puente como tal, sino parte como puente y parte como terreno. Lo que significa que se debe eliminar manualmente esta estructura. En teoría, si se hubiese detectado el puente adecuadamente, tan sólo deberíamos importar en el software Global Mapper los datos del terreno y el agua.

Para eliminar el puente, se procede como sigue. Primero determinamos la cota a la que debe estar cada parte que vamos a eliminar:

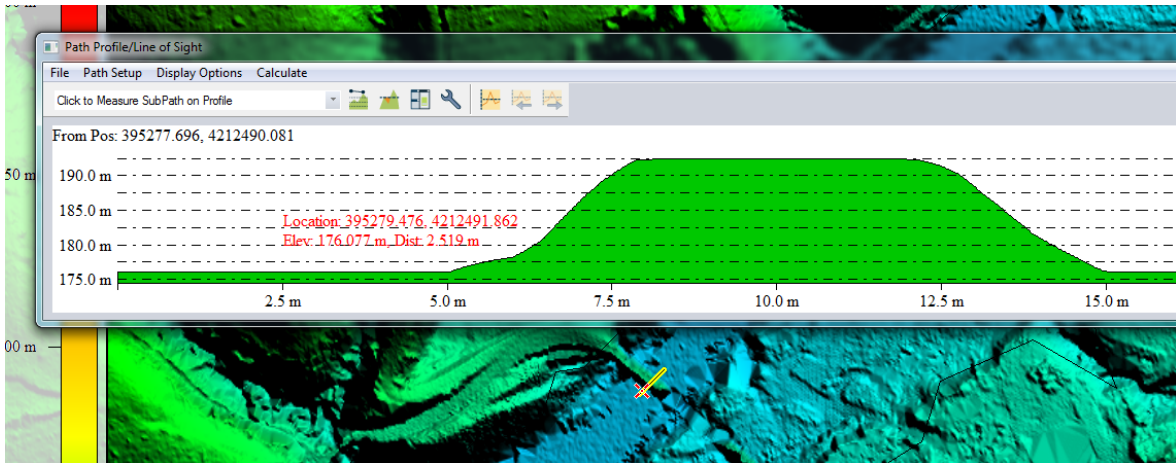


Ilustración 3.7.- Perfil del terreno de la parte derecha del puente. Cota del agua: 176,1 m

Se crea un área que ‘envuelva’ al puente para poder modificar su cota, y se le asigna el valor correspondiente, en este caso 176,1 m:

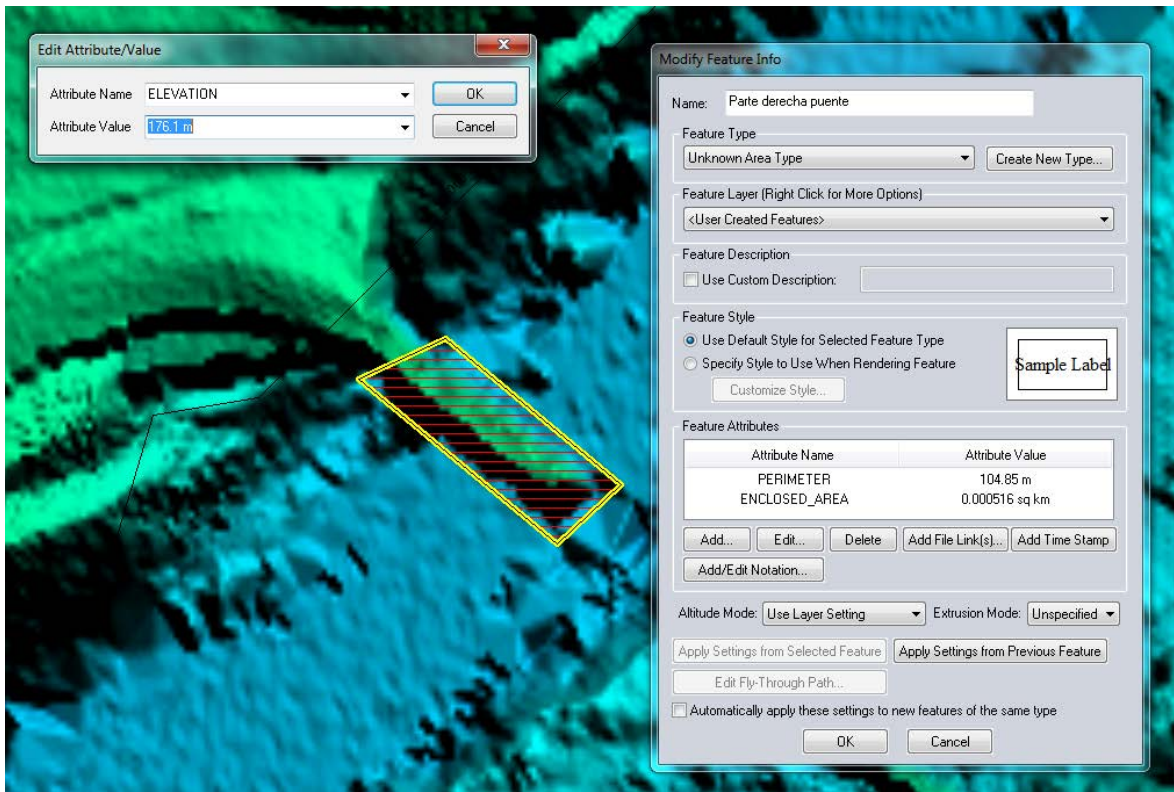


Ilustración 3.8.- Área generada para poder sustituir su valor de cota

Se realiza lo mismo para la otra parte del puente. A continuación, se crea una malla a partir de las dos áreas y la cota que se le ha asignado:

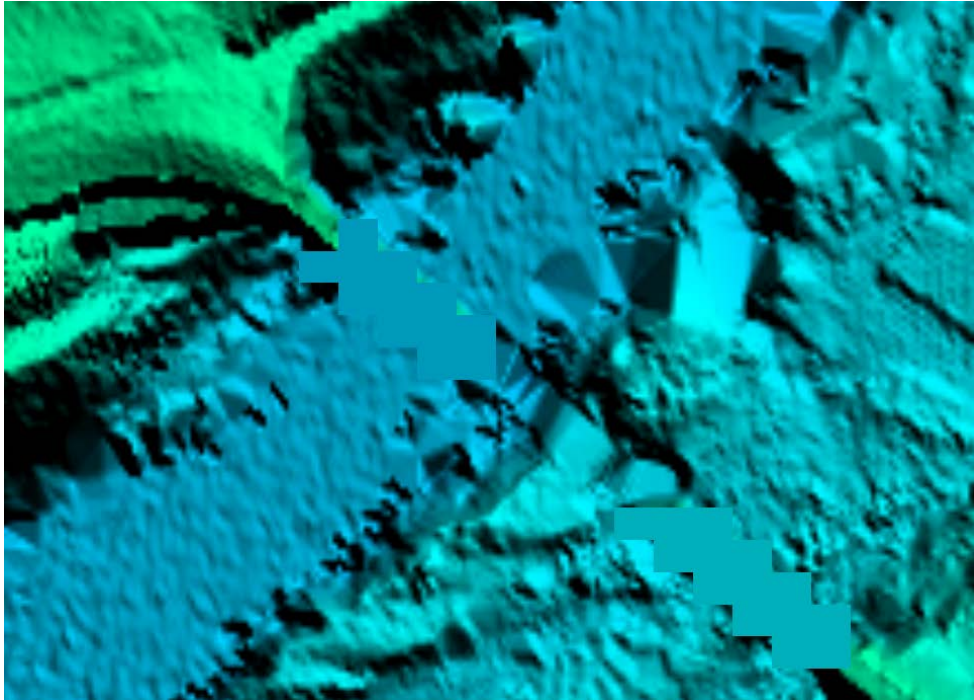


Ilustración 3.9.- MDT después de la corrección de la cota del puente

Se combina el MDT original con las dos mallas nuevas que se acaban de crear. El criterio para obtener el MDT resultante es que, donde haya solape, se tomará la menor de las dos cotas. El paso de malla vuelve a ser de 5 x 5 m.

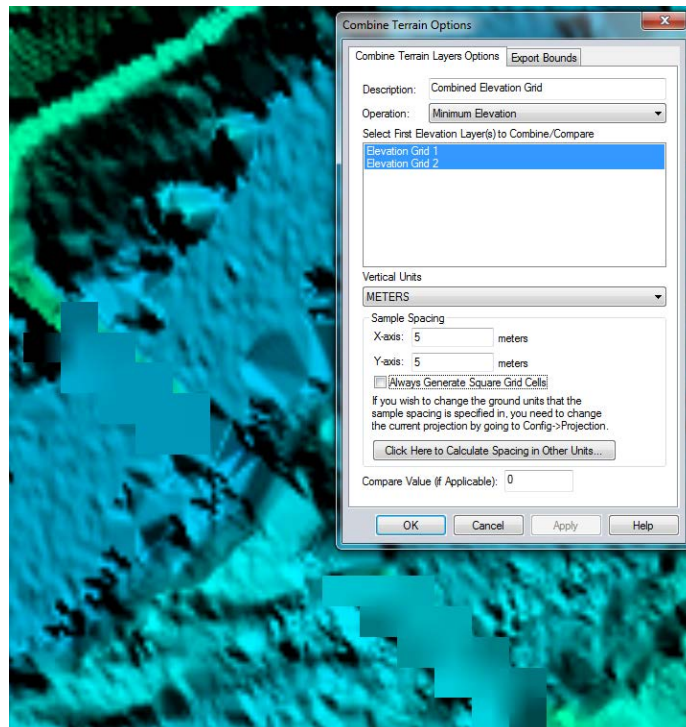


Ilustración 3.10.- Combinación de ambos MDTs

A continuación, se descarta toda la zona que no es objeto de estudio. Para ello, nos basamos en la zona inundable obtenida a partir de simulaciones previas. Estas

simulaciones han sido realizadas por el tutor del presente TFM, el Dr. Patricio Bohórquez a partir de MDTs obtenidos por estereocorrelación fotogramétrica. Los datos proceden de una simulación 2D para un caudal máximo de 2.000 m³/s. Esta limitación implica que la malla que aquí se genera no es válida para simulaciones donde el caudal sea superior a 2.000 m³/s.

La justificación de tomar un paso de malla de cinco metros se debe a dos motivos:

1. La superficie inundable para un caudal de 2.000 m³/s es del orden de 1 km² y el número máximo de celdillas que simulaciones previas recomiendan hacer está comprendido entre 500.000 y 1.500.000.
2. El nivel de precisión que se obtiene con este paso de malla se considera apropiado, dado que el objetivo es determinar la zona inundable más que la simulación específica de algún elemento concreto. Así, no se modela el puente ni los árboles, pues el tamaño de estos es del orden del paso de malla.

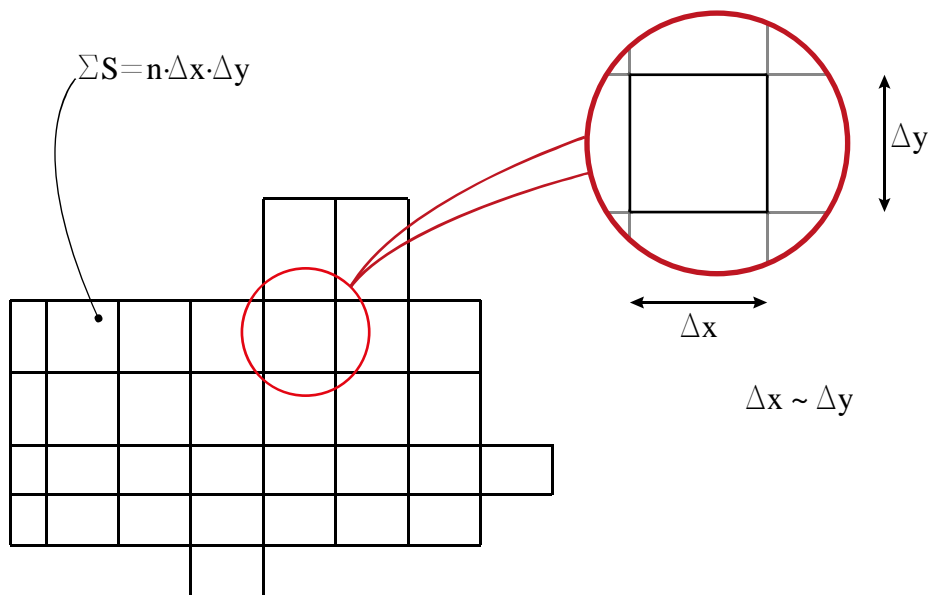


Ilustración 3.11.- Esquema de malla estructurada. Determinación del paso de malla

De este modo, si el fijamos el número de elementos en la vertical en cuarenta y el número total de celdillas de la malla en 1.000.000; podemos decir que cada 'capa' horizontal de la malla tendrá 25.000 elementos. Así, podemos escribir:

$$\Sigma S = n \cdot \frac{1}{2} \cdot \Delta x \cdot \Delta y = 1 \text{ km}^2 = 1.000.000 \text{ m}^2 \quad (3)$$

$$\Delta \sim \Delta y = \sqrt{\frac{\Sigma S}{n}} = \sqrt{\frac{1.000.000}{25.000}} \approx 6,33 \text{ m} \quad (4)$$

Por último, exportamos un archivo .stl con el MDT generado a partir de los datos LiDAR (formato .laz) con la eliminación del puente situado justo aguas abajo de la presa.

3.2 Generación de la malla a partir del MDT en formato STL

Se comienza por importar el archivo .stl en Pointwise: *File > Import > Database* (*Shell Split angle = 0*) (Ilustración 3.12)

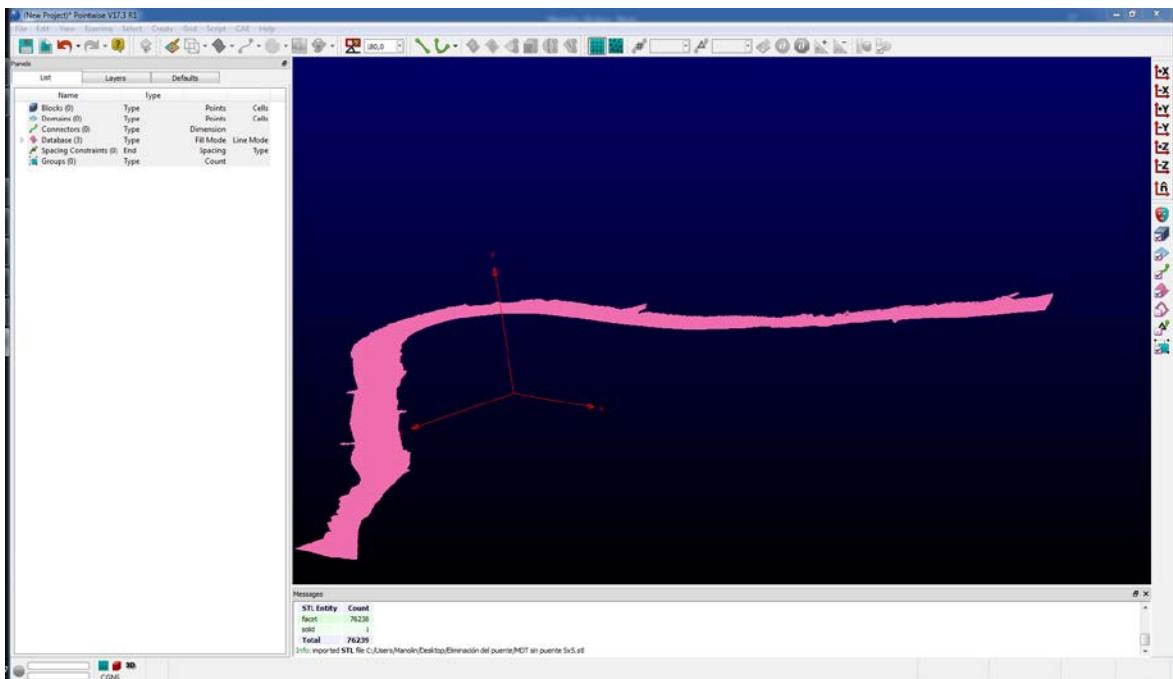


Ilustración 3.12.- MDT en formato STL importado en Pointwise

Al ser la entrada (*inlet*) un poco irregular, se corta para poder obtener un plano de entrada adecuado:

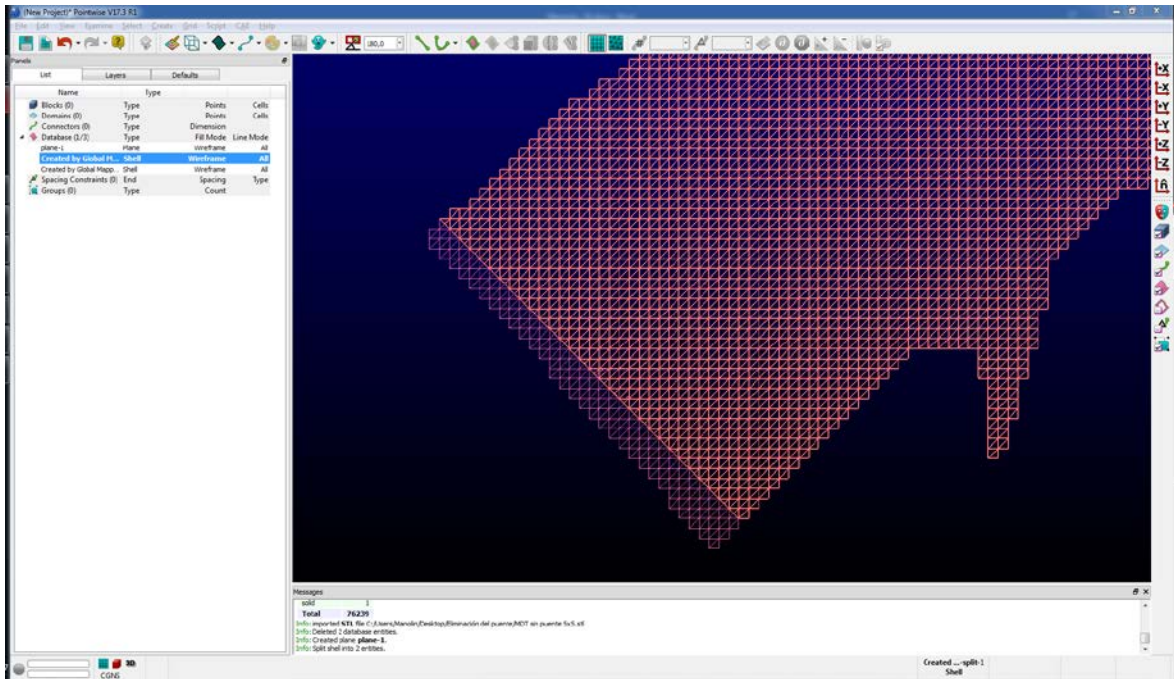
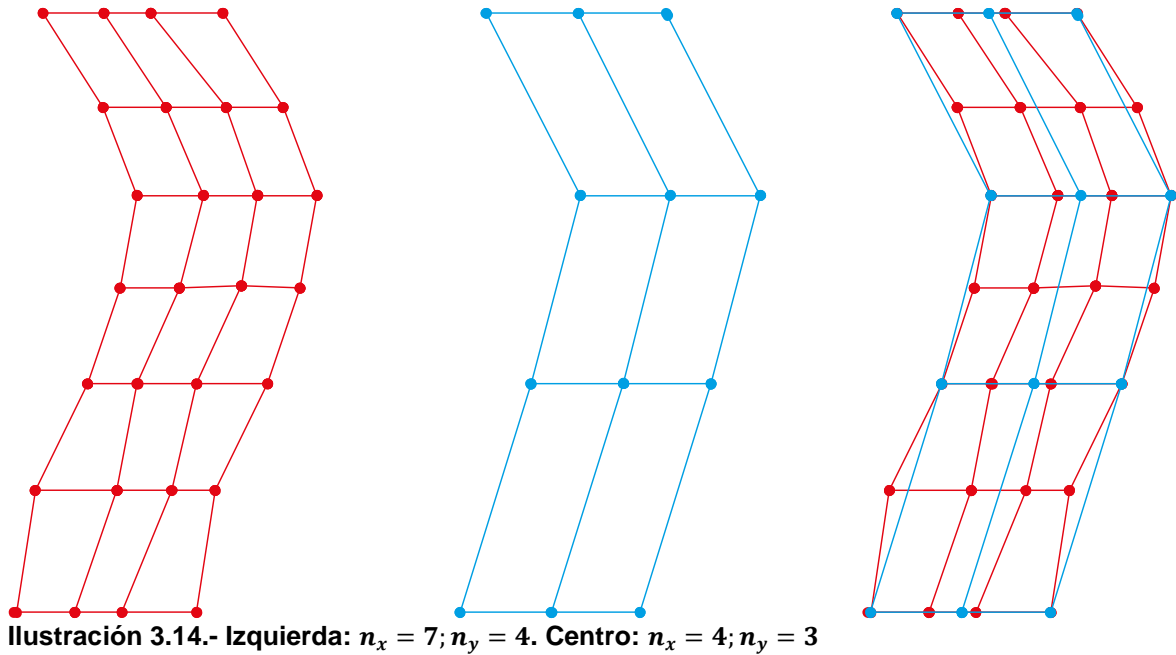


Ilustración 3.13.- Plano que divide el MDT en dos partes

Seleccionamos el *Database*, *Edit > Split > Point and Normal*, y marcamos un punto y una dirección del vector (en este caso, la dirección es $[-1 \ 0 \ 1]$) y hacemos click en *OK*. A continuación, descartamos la parte que no es de nuestro interés.

Lo siguiente será crear los *connectors* a partir del MDT. Estos *connectors* son las líneas que conforman el perímetro. Se selecciona el *Database > Connectors on database entities*. El programa crea dos *connectors*, que debemos unir: los seleccionamos: *Edit > Join*. Una vez unidos, se dividen en cuatro: dos márgenes (izquierda y derecha), entrada y salida.

A continuación, debemos indicar la *dimensión* de estos cuatro conectores. Esto es, el número de puntos que conforman las líneas del perímetro. En la Ilustración 3.14 se observan dos configuraciones distintas para los mismos datos iniciales. A la izquierda (rojo), la dimensión de los *connectors* de entrada y salida es de 3 ($n_y = 3$) y la dimensión de los *connectors* de las márgenes del río es de 7. En el caso del centro (azul) las dimensiones son 4 y 2 respectivamente.



Para poder tomar una decisión en cuanto a la dimensión de los *connectors*, es necesario realizar dos hipótesis:

1. El número de elementos en la vertical será igual a 40
2. La relación de aspecto la limitaremos a 5

Además, se conoce la longitud de cada uno de los *connectors* que componen el perímetro:

Tabla 2.- Longitud de los *connectors*

Elemento	Longitud (m)
Margen Izquierda	5.800
Margen Derecha	5.800
Entrada	220
Salida	370

Con todo lo anterior, se puede escribir:

- El producto de las dimensiones de los *connectors* menos uno será igual al número de hexaedros que habría en una malla de estructurada:

$$(n_x - 1) \cdot (n_y - 1) \cdot \underbrace{(n_z - 1)}_{n_z=40} = 10^6 \tag{5}$$

- La longitud de cada celdilla (l_i) multiplicada por la dimensión del *connector* en esa dirección ($n_i - 1$) es igual a la longitud total del *connector*.

$$(n_x - 1) \cdot l_x = L_x = 4.800 \text{ m} \quad (6)$$

$$(n_y - 1) \cdot l_y = L_y = 300 \text{ m} \quad (7)$$

Donde se ha tomado el valor medio de la anchura de la sección de entrada y la sección de salida

- La relación de aspecto (*aspect ratio*) de las celdillas se fija en 5. El motivo es que diversos autores recomiendan que no sea superior a 10, y dado que la forma de la celdilla debe adaptarse a las condiciones, fijándolo en 5 todavía se permite una cierta deformación del elemento:

$$\frac{l_x}{l_y} = 5 \quad (8)$$

Las ecuaciones (4.) a (7.) forman un sistema de cuatro ecuaciones con cuatro incógnitas, cuyas soluciones son:

- $l_x = 18 \text{ m}$
- $l_y = 3,7 \text{ m}$
- $n_x = 315 \text{ elementos}$
- $n_y = 82 \text{ elementos}$

Donde tan sólo es de interés la dimensión de cada *connector* (n_i). A partir de estos resultados, se continúa con la generación de la malla. Se seleccionan las dos márgenes y se les asigna la dimensión de 315 elementos, realizándose lo mismo para la entrada y salida, con dimensión de 82 elementos. Así, se tiene una malla como la de la Ilustración 3.15

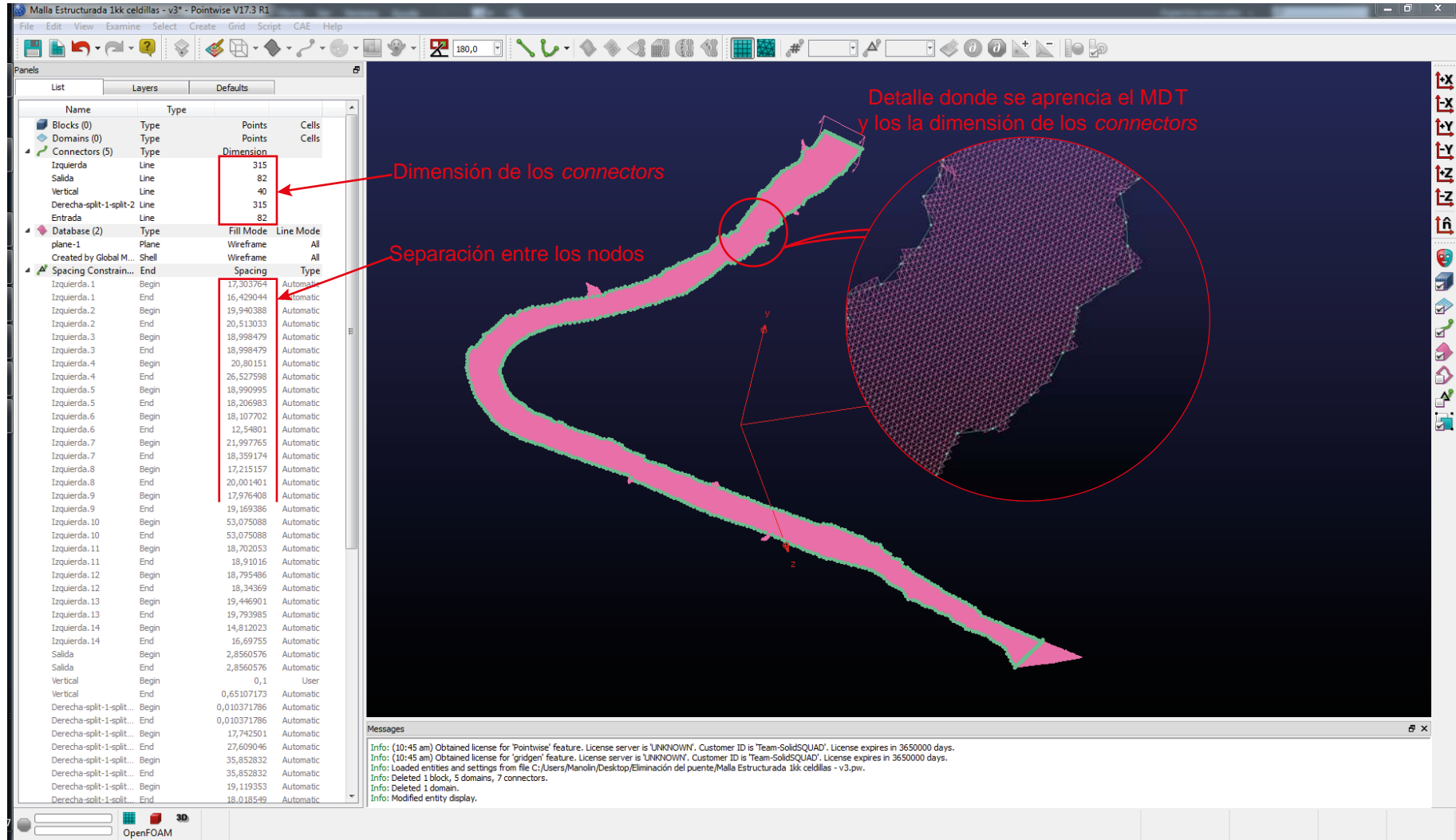


Ilustración 3.15.- MDT (rosa) y connectors (azul). Características de los elementos en PointWise

Lo siguiente es crear un *connector* vertical en el que se determine la altura de las celdillas. Para ello se procede como sigue. Se fija la longitud total del *connector*, que en base a simulaciones previas se puede tomar el valor de 10 m. Para crear un *connector* vertical, se selecciona *Create > 2 point Curves...* y se toma cualquier punto de la malla (para mayor comodidad tomar un punto del perímetro). Se marca el recuadro *Advanced* y se escribe en *XYZ Offset* el vector 0 10 0. Así se crea un *connector* vertical de longitud 10 m.

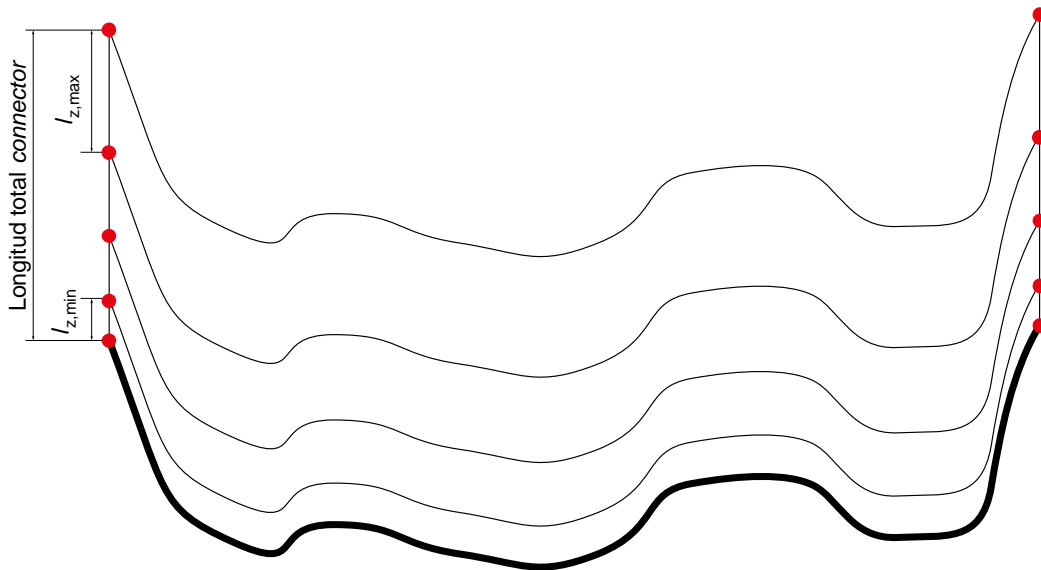


Ilustración 3.16.- Esquema de la altura de las celdillas en función de la distancia al cauce

Se le asigna la dimensión de 40. Al asignar la dimensión los puntos en los que se divide el segmento (nodos) se disponen equiespaciados. Puesto que se precisa una malla más refinada en la parte próxima al cauce, es preciso especificarlo. Para ello, en *Spacing constraints* se marca el *connector* vertical y se fija el paso de malla. Por defecto se toma 0,256 m (0,2564 m·39 = 10 m) pero lo modificamos y lo fijamos en 0,1 m. El resultado se observa en la Ilustración 3.17:

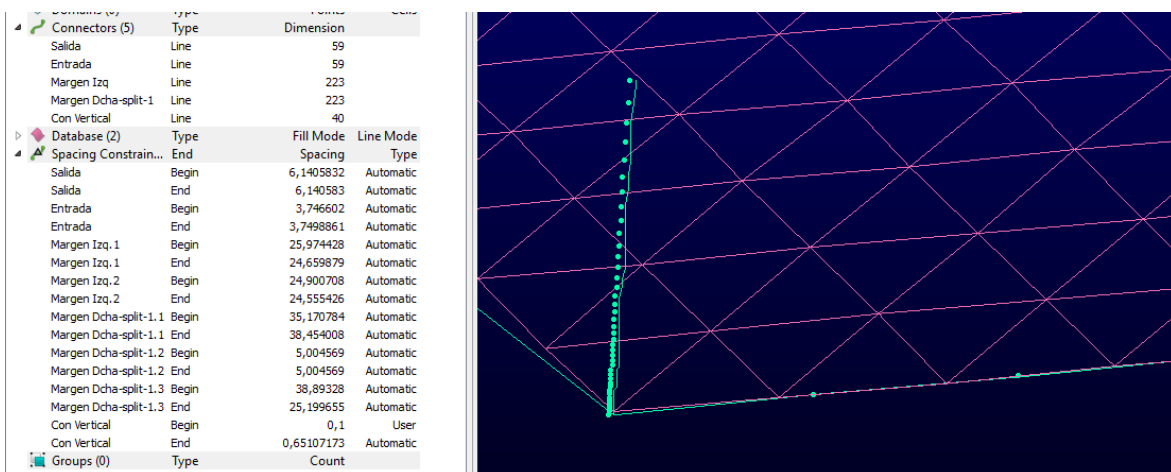


Ilustración 3.17.- Connector sobre el que se extruye el MDT

En este momento se genera el *domain* que es la malla bidimensional estructurada a partir del perímetro del MDT. Lo que se persigue realizar se esquematiza en la Ilustración 3.18:

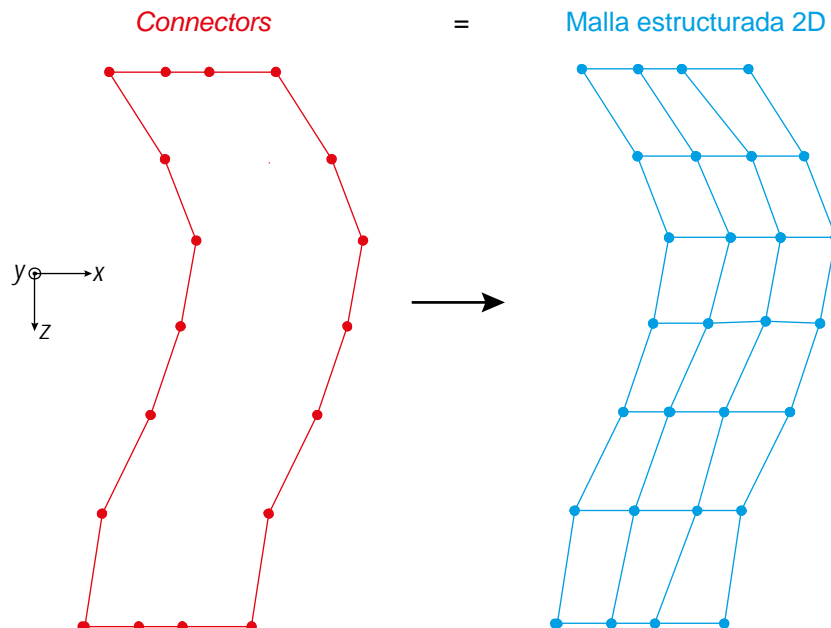


Ilustración 3.18.- Esquema del proceso seguido para la generación del domain estructurado que se proyectará sobre el MDT

Es importante destacar que ni la malla se encuentra en un plano, ni sobre el MDT. Tan sólo se emplean los puntos del perímetro del MDT para generar la malla. Para ello, se procede como sigue:

1. Se seleccionan los *connectors* de las márgenes del río y se dividen en tantos tramos (Ilustración 3.19) como sean necesarios para evitar que ocurra lo que se muestra en la Ilustración 3.20. En esta ilustración se aprecia lo que ocurre si no se especifica cómo deben situarse los nodos de los *connectors*. Al tener a cada lado *connectors* de la misma dimensión, en nodo *i*-ésimo de un lado se conecta con su homólogo del otro lado. Al unirse no se tiene en cuenta que la línea que los una sea aproximadamente perpendicular a la dirección del flujo. Se observa, de un modo cualitativo, que la calidad de la malla mejora sustancialmente al indicar el número de nodos y su distribución en zonas altamente irregulares (meandro). El parámetro indicador para esta situación es el *skewness*, del que se hablará en el apartado correspondiente. Este parámetro está asociado a la *oblicuidad* de las celdillas: cuanto más deformadas, peor es la malla.

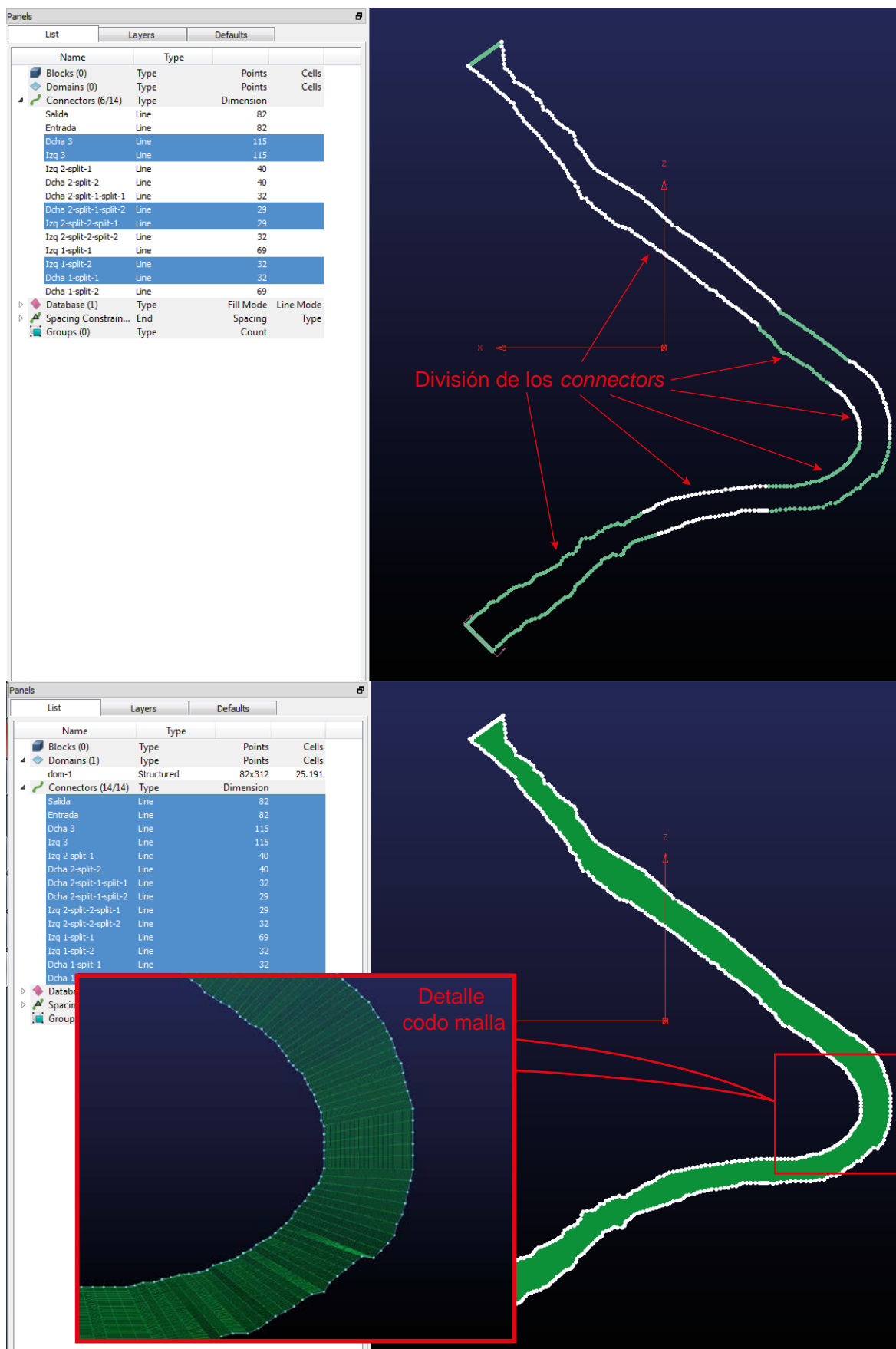


Ilustración 3.19.- Superior: División de las márgenes en conectores de igual dimensión. Inferior: Malla 2D estructurada y detalle de la misma en el codo

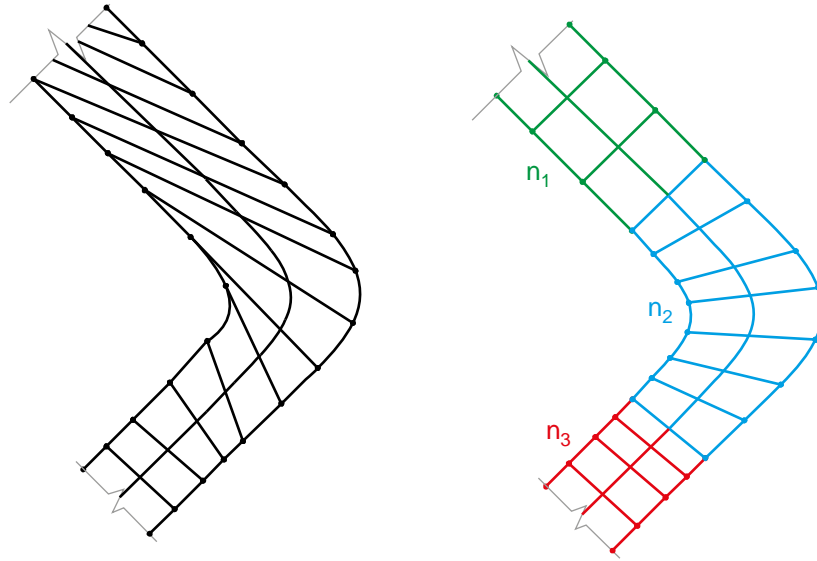


Ilustración 3.20.- Izquierda: disposición incorrecta de los nodos de los conectores de las márgenes del río. Derecha: disposición correcta.

2. Una vez se han distribuido los nodos sobre el perímetro de un modo adecuado, se selecciona todo el perímetro (debe estar totalmente cerrado) y se hace click en *Assemble Domains*. Si todo está correcto, se creará la malla estructurada que se muestra en la parte inferior de la Ilustración 3.19.
3. El siguiente paso es suavizar la malla. Este punto es muy importante, pues de un modo muy sencillo mejora sustancialmente la calidad de la malla. En este paso la malla se va a deformar suavizándose automáticamente. Para ello, basta con seleccionar el *domain* y hacer click en *Start Solve*. Cuando ya se vea que la malla no se modifica, se hace click en *Stop Solve* (Ilustración 3.21).

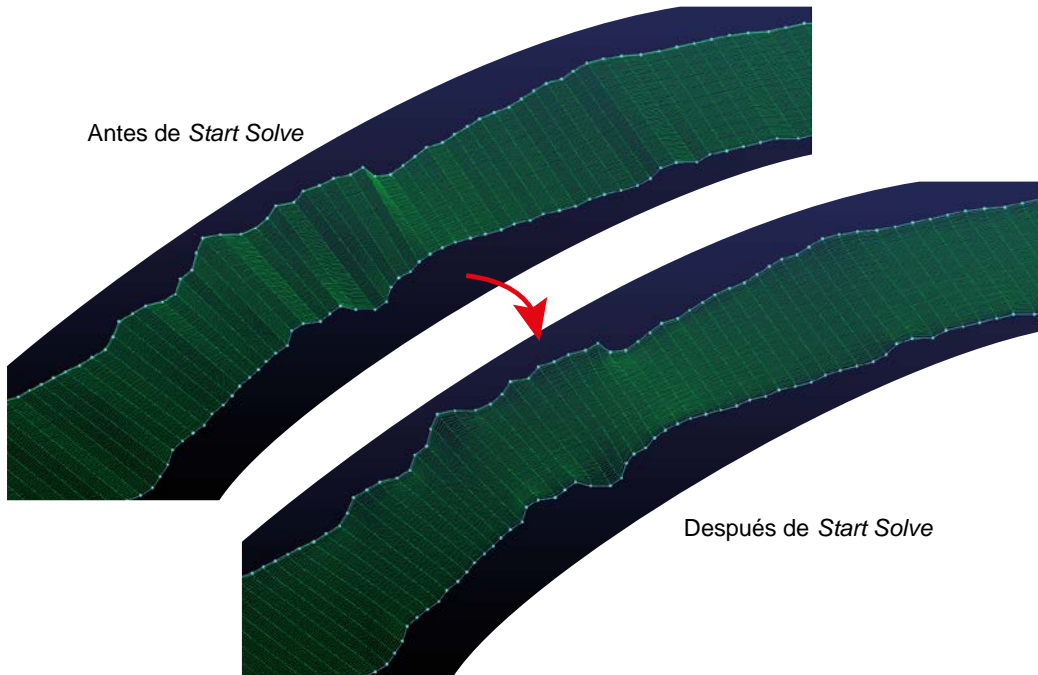


Ilustración 3.21.- Comparativa del efecto del suavizado automático de la malla

- Una vez se ha generado el *domain* en 2D que conforma la malla del perímetro, es necesario asignarle cota a los puntos interiores. Nótese que no ha sido necesario emplear el MDT directamente para la generación de la malla, tan sólo el perímetro, por lo que los puntos interiores de la malla son interpolaciones lineales de los puntos del perímetro. Así, el siguiente paso es seleccionar la malla actual y *proyectarla* sobre el MDT. Para ello, se selecciona el *domain*, *Edit > Project...* Y se marca como indica la Ilustración 3.22. Por último, se hace click en *Project*. El resultado antes y después se puede ver en la Ilustración 3.23

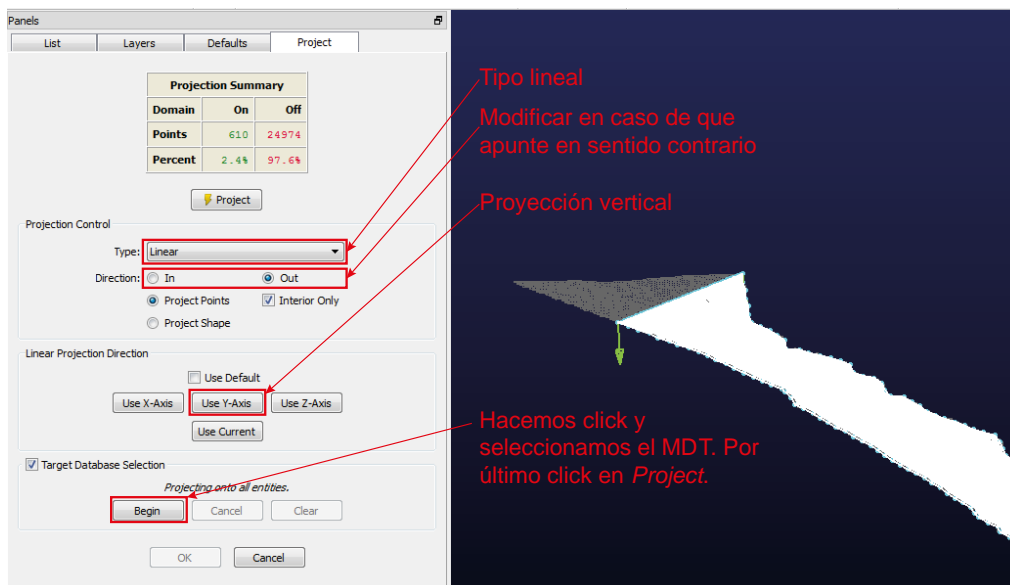


Ilustración 3.22.- Pasos seguidos para proyectar el domain sobre el MDT

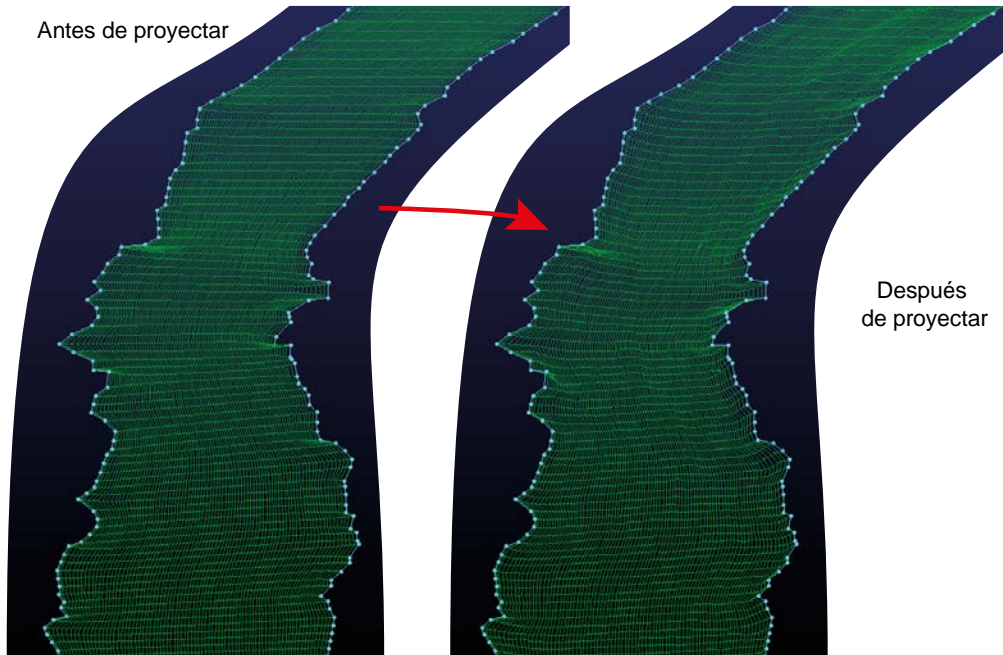


Ilustración 3.23.- Resultado de la proyección del domain sobre el MDT

Ya sí se tiene una malla bidimensional estructurada, alineada con la dirección del flujo y del tamaño adecuado que representa fielmente el MDT.

5. El siguiente paso es *extruir* la malla 2D a través del *connector* vertical que se ha creado anteriormente:

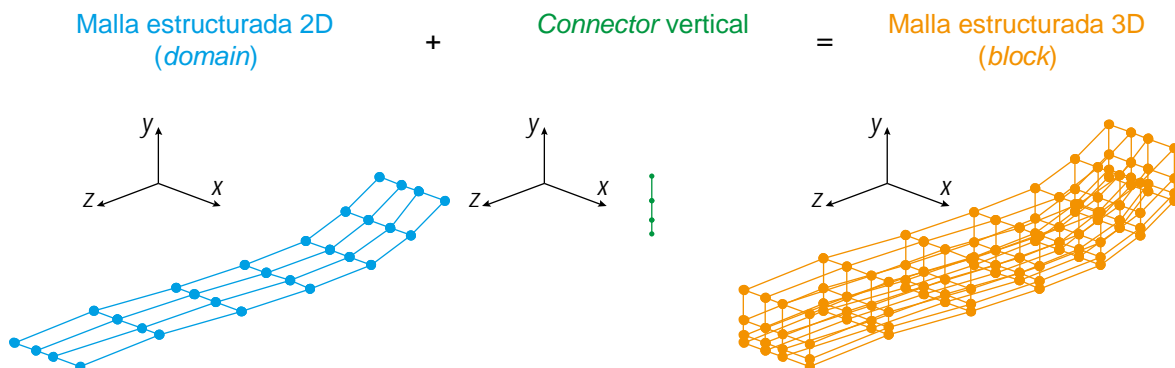


Ilustración 3.24.- Esquema general seguido para la generación del block que conformará la malla

Para ello, se selecciona la malla 2D, *Create > Extrude > Path...* se selecciona el *connector* vertical y se hace click en *Run*. Si la malla 2D está bien realizada, la extrusión no debería dar ningún problema.

6. El último paso consiste en asignarle las propiedades para realizar la simulación. Así, se selecciona *CAE > Select solver... OpenFOAM*. Lo siguiente será asignar las condiciones de contorno: *CAE > Set Boundary Conditions...* y se definen cinco tipos de condiciones de contorno, a saber:

- Entrada
- Salida
- Cauce
- Aire
- Laterales

De las seis caras (*domains*) que conforman la malla, dos serán los laterales, mientras que el resto de categorías tendrán solo una cara.

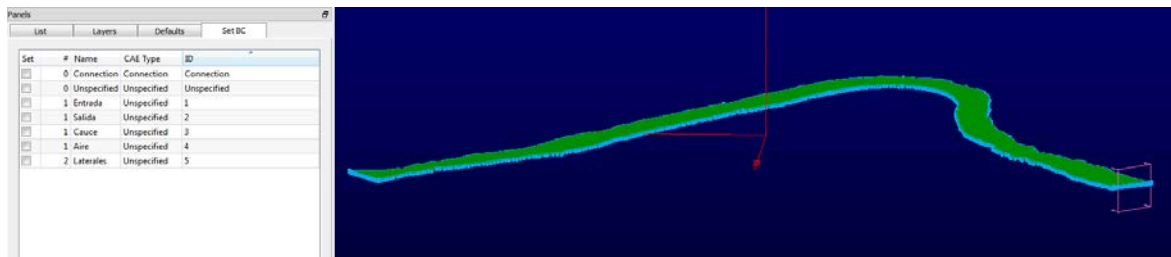


Ilustración 3.25.- Resultado final de la malla generada

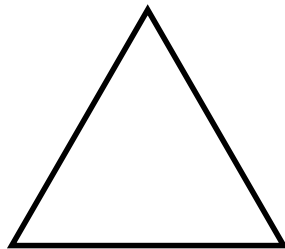
4 Parámetros de calidad de malla

La mejor manera de determinar la calidad de la malla es observando diversos parámetros como el *máximum skewness* aunque es necesario también realizar una inspección visual.

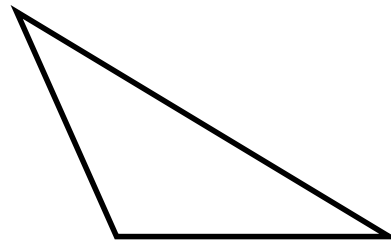
Una buena malla garantiza los mejores resultados para el análisis, minimiza las necesidades de otros análisis y mejora las capacidades predictivas. La malla debe ser suficientemente fina para capturar los fenómenos de interés. La resolución de la malla depende de la malla del contorno, a partir de la cual se genera el interior de la misma. En mallas de calidad, la variación del tamaño de una cara o celdilla a la contigua es mínima. Una gran variación del tamaño entre caras o celdillas contiguas irá en detrimento de la calidad de la malla debido a que los métodos numéricos que se emplean para resolver las ecuaciones en derivadas parciales se basan en la suposición de que la variación del tamaño entre elementos es suave.

A continuación, se describen los principales indicadores de calidad de malla, según la guía de Ansys Fluent:

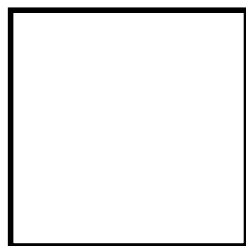
4.1 Skewness



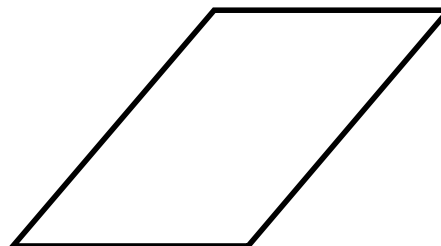
Triángulo equilátero
(*equilateral triangle*)



Triángulo muy oblicuo
(*highly skewed triangle*)



Cuadrilátero rectángulo
(*equiangular quad*)



Cuadrilátero muy oblicuo
(*highly skewed quad*)

Ilustración 4.1.- Ejemplo gráfico del indicador skewness para diversas geometrías 2D

El *skewness* (oblicuidad en español) es probablemente el indicador de calidad de malla más importante. Determina cómo de próximo a una celdilla ideal es cada una de las celdas (caras en mallas 2D). Se entiende como celdilla ideal aquella que es equilátera, (todos los lados iguales entre sí). La siguiente tabla muestra el valor de *skewness* y la correspondiente calidad de celda:

Tabla 3.- Calidad de la celdilla en función del parámetro Skewness

Skewness	Calidad de la celdilla
1,00	Degenerada
0,90 ÷ 1,00	Mala (<i>sliver</i> = astilla)
0,75 ÷ 0,90	Pobre
0,50 ÷ 0,75	Aceptable
0,25 ÷ 0,50	Buena
0,00 ÷ 0,25	Excelente
0,00	Equilátera

De acuerdo con la definición de *skewness*, un valor de 0 indica que la celda es equilátera (la mejor) y un valor de 1 indica que está completamente degenerada. Las celdas degeneradas (astillas, *sliver* en inglés) se caracterizan porque sus nodos son casi complanarios. Celdas con valor de *skewness* superior a la unidad son inválidas.

Las caras y celdas muy oblicuas deben evitarse debido a que tienden a producir resultados mucho menos precisos que cuando las celdas son relativamente equiláteras. Existen dos métodos para medir la oblicuidad de una celda:

- Basados en el volumen equilátero (únicamente aplicables a tetraedros)
- Basados en la desviación de los ángulos. Este último se aplica a celdas y caras, como pueden ser pirámides y prismas.

Se define **oblicuidad basada en el volumen equilátero** (*Equilateral-Volume-Based Skewness*) según la expresión:

$$\text{Skewness} = \frac{\text{Optimal Cell Size} - \text{Cell Size}}{\text{Optimal Cell Size}} \quad (9)$$

Donde el tamaño óptimo de celda es el tamaño de una celda equilátera con el mismo circunrado (Ilustración 4.2).

En mallas tridimensionales, la mayoría de las celdas deberían ser buenas (*skewness* < 0,50) y normalmente habrá una fracción de celdillas que serán

aceptables ($0,50 < skewness < 0,75$). La presencia de celdillas pobres ($0,75 < skewness < 0,90$) indica normalmente una disposición incorrecta de los nodos del contorno. En estos casos, lo apropiado es mejorar el contorno de la malla lo máximo posible, debido a que la calidad de la malla general no puede ser superior que la calidad de la malla del contorno.

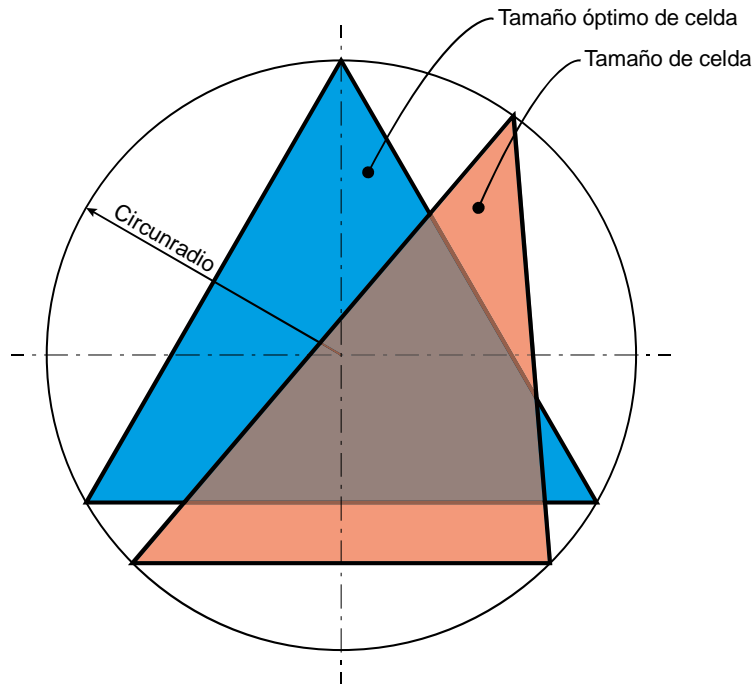


Ilustración 4.2.- Definición gráfica de skewness

Se define **oblicuidad normalizada equiangular** (*Normalized Equiangular Skewness*) según la expresión:

$$Skewness = \max \left[\frac{\theta_{max} - \theta_e}{180^\circ - \theta_e}; \frac{\theta_e - \theta_{min}}{\theta_e} \right] \quad (10)$$

Donde:

- θ_{max} es el ángulo mayor de la cara
- θ_{min} es el ángulo menor de la cara
- θ_e es el ángulo óptimo de la cara (60° para triángulos, 90° para cuadriláteros, etc...)

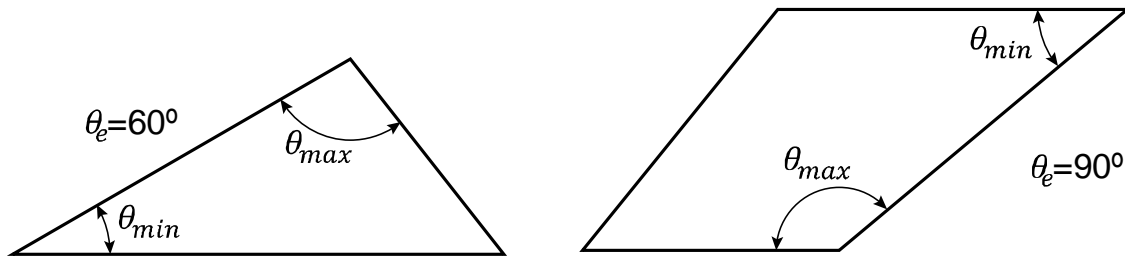


Ilustración 4.3.- Normalized equiangular skewness

La oblicuidad (*skewness*) de una celda será la máxima calculada para cada cara. Por ejemplo, una pirámide ideal (*skewness* = 0) será aquella constituida por cuatro caras triangulares que sean equiláteras (60°) y la base cuadrada (90°).

4.2 Size change

El parámetro *size change* se define como la ratio entre el área (o volumen) de una cara (o celda) con relación al área (o volumen) de cada cara (o celda) contigua. Este parámetro se calcula para cada cara (o celda) del dominio computacional.

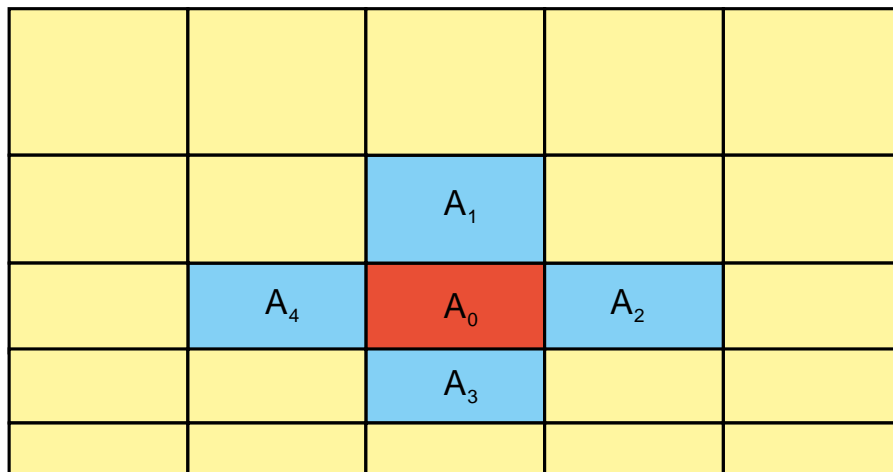


Ilustración 4.4.- Definición gráfica del parámetro size change para la cara A₀.

$$\text{Size Change}_{\text{cara}} = \frac{\max\{A_i\}}{A_0} ; \quad i = 1,2,3,4 \tag{11}$$

4.3 Aspect ratio

El parámetro *aspect ratio* se define como la ratio entre la máxima longitud del elemento con respecto a la mínima longitud del mismo elemento. Este parámetro se emplea en elementos triangulares, tetraédricos y hexaédricos; y se define de manera

diferente según sea el elemento. El *aspect ratio* se puede utilizar para determinar la proximidad de una celda a ser ideal.

- Para caras (o celdas) equiláteras (como triángulos equiláteros, cuadrados, etc...) el *aspect ratio* es la unidad
- Para caras (o celdas) regulares, el *aspect ratio* será superior a uno, según la variación de longitud de las aristas
- Para caras tetraédricas y celdas piramidales, suele ser mejor centrar los esfuerzos en mejorar el *skewness*, ya que el *aspect ratio* mejorará también.
- Para prismas, es importante chequear el *aspect ratio* y/o el *change size* además del *skewness*, ya que es posible tener un cambio brusco del tamaño de celda con un *skewness* bajo, así como es posible tener un tamaño más o menos uniforme (*aspect ratio* ~ 1) y un cambio brusco del tamaño de celda (*aspect ratio* >> 1)

Es importante destacar que un *aspect ratio* elevado no necesariamente significa que la malla sea mala, siempre que deformar tanto los elementos tenga una razón justificada. Habitualmente se generarán mallas con zonas localizadas de elementos muy esbeltos (*aspect ratio* elevada, *skewness* moderada) en zonas de especial interés, habitualmente para capturar la capa límite.

4.4 Ortho Skew

El parámetro *ortho skew* para una **celda** se calcula utilizando el vector normal de cada cara, \vec{A}_i ; el vector que une el centroide de una celdilla con el centroide de la celdilla vecina, \vec{c}_i y el vector que une los centroides de la celdilla y sus caras, \vec{f}_i . La Ilustración 4.5 muestra los vectores utilizados para determinar el parámetro *ortho skew* de una celda.

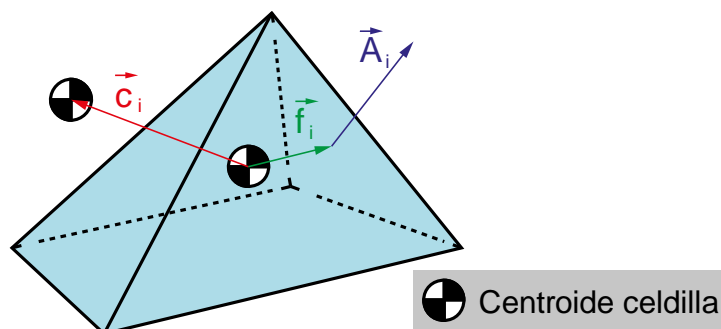


Ilustración 4.5.- Celdilla y vectores necesarios para la definición del Ortho skew de una celda.

Para cada cara i , se calcula el valor $(1 - \text{coseno del ángulo entre el vector normal a la cara, } \vec{A}_i, \text{ y el correspondiente vector que une el centroide de la celda con el de la cara, } \vec{f}_i)$:

$$\text{OrthoSkew}_1 = 1 - \frac{\vec{A}_i \cdot \vec{f}_i}{|\vec{A}_i| \cdot |\vec{f}_i|} \quad (12)$$

Después, para cada cara i , se calcula el valor $(1 - \text{coseno del ángulo entre el vector normal a la cara, } \vec{A}_i, \text{ y el correspondiente vector que une el centroide de la celda con el centroide de la celdilla con la que comparte cara, } \vec{c}_i)$:

$$\text{OrthoSkew}_2 = 1 - \frac{\vec{A}_i \cdot \vec{c}_i}{|\vec{A}_i| \cdot |\vec{c}_i|} \quad (13)$$

El parámetro *ortho skew* para la celda depende del tipo de celda:

- Para celdas piramidales, el *ortho skew* es el máximo de los dos posibles calculados anteriormente y el *cell skewness*.
- Para el resto de celdas, el *ortho skew* es el máximo de los dos posibles calculados anteriormente:

$$\text{OrthoSkew}_{\text{piramidal}} = \max\{\text{OrtoSkew}_1; \text{OrtoSkew}_2; \text{Skewness}\} \quad (14)$$

$$\text{OrthoSkew}_{\text{no-piramidal}} = \max\{\text{OrtoSkew}_1; \text{OrtoSkew}_2\} \quad (15)$$

Cuando una cara de la celdilla pertenece al contorno, el vector \vec{c}_i a través de esa cara se ignora.

El parámetro *ortho skew* para una **cara** se calcula de un modo similar utilizando el vector normal, \vec{A}_i , y el vector que une los centroides de la cara y la arista, \vec{e}_i (Ilustración 4.6):

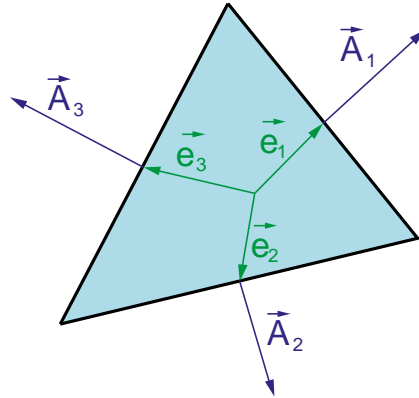


Ilustración 4.6.- Cara y vectores necesarios para la definición del Ortho skew de una cara

El *ortho skew* de una cara es el máximo de los valores (1 - coseno del ángulo entre el vector normal a la arista, \vec{A}_i , y el correspondiente vector que une el centroide de la cara con el de la arista, \vec{e}_i):

$$\text{OrthoSkew}_{\text{arista},i} = 1 - \frac{\vec{A}_i \cdot \vec{e}_i}{|\vec{A}_i| \cdot |\vec{e}_i|} \quad (16)$$

$$\text{OrthoSkew}_{\text{cara}} = \max\{\text{OrthoSkew}_{\text{arista},i}\} \quad (17)$$

El parámetro *Orthogonal Quality*, que es el que se emplea en ANSYS Fluent, es equivalente al parámetro *Orthoskew*, tan sólo con la escala invertida:

$$\text{Orthogonal Quality} = 1 - \text{OrthoSkew} \quad (18)$$

De este modo, las celdillas de una malla ideal tendrán un OrthoSkew de 0, y un *Orthogonal Quality* igual a la unidad.

4.5 Ejemplo de aplicación

Para destacar la influencia que tienen los indicadores descritos anteriormente en el resultado final de la simulación, teniendo en cuenta tanto el coste computacional, como la precisión y la capacidad de capturar el fenómeno de interés, se va a exponer un ejemplo que está extensamente estudiado y se conoce el resultado a través de ecuaciones analíticas. Supóngase que se quiere analizar el flujo a través de un obstáculo en un canal de sección regular como se muestra en la Ilustración 4.7:

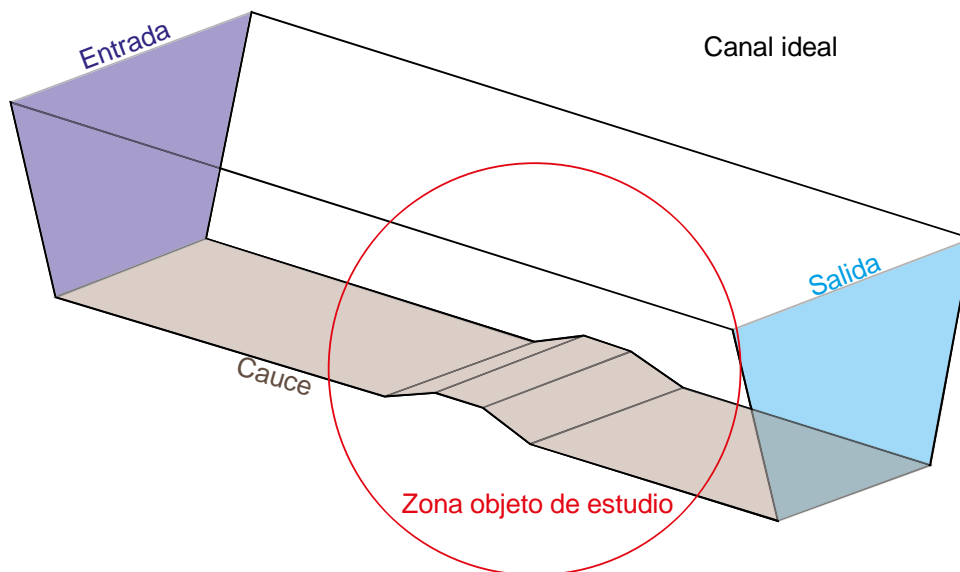


Ilustración 4.7.- Canal de sección trapezoidal (ideal) con un obstáculo también trapezoidal.

La distribución de velocidades es conocida, siendo aproximadamente la que se muestra en la Ilustración 4.8.

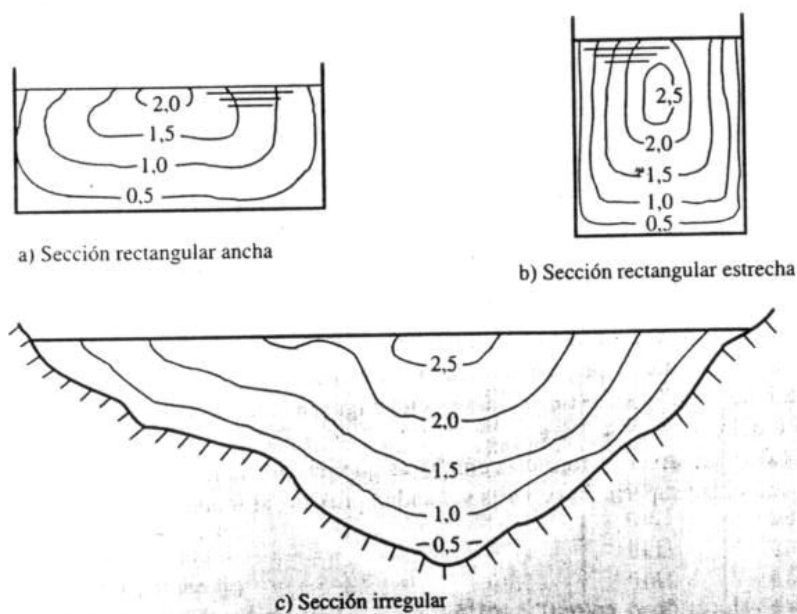


Ilustración 4.8.- Distribución de la velocidad en una sección transversal de un canal. Fuente: Lázaro López.

Se plantea ahora la generación *optimizada* de la malla. Como todas las mallas, se pueden dividir fundamentalmente en dos categorías: estructuradas y no estructuradas, aunque se podría considerar una tercera categoría: híbridas.

A continuación, se van a generar dos mallas, una estructurada y otra no estructurada, y se van a comparar ambas mallas, empleando tanto los indicadores descritos anteriormente como en análisis visual de la malla y el conocimiento que se tiene del fenómeno estudiado.

Los fenómenos de interés en esta simulación son fundamentalmente dos: la distribución de velocidades del agua en cualquier sección del canal y el perfil del agua al pasar por el obstáculo. Teniendo en cuenta la distribución de velocidades que se observa en la Ilustración 4.8, se ha generado la siguiente malla estructurada (Ilustración 4.9):

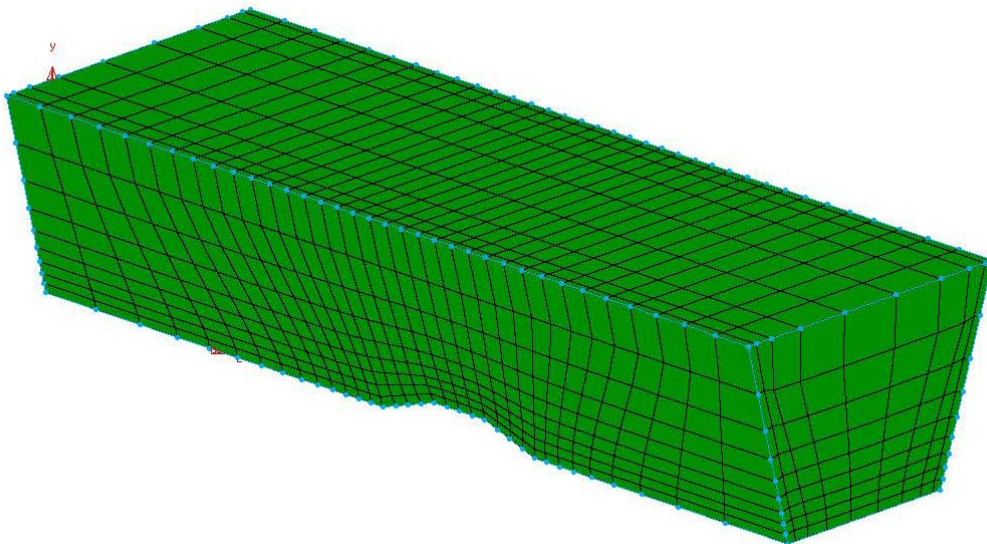


Ilustración 4.9.- Malla estructurada de la geometría del canal descrito (2.835 celdillas)

Se ha procurado refinar la malla en las proximidades del contorno para capturar el gradiente de velocidad que tiene lugar, así como refinar la malla en la zona del obstáculo. Se observa que los vectores normales de las caras de las celdillas, a priori, están orientados en la dirección del flujo. Además, el espesor de las celdillas es menor donde es mayor el gradiente de velocidades. Se concluye, por tanto, a falta de que sea validado con una simulación, que la malla computacional estructurada es adecuada para analizar el fenómeno.

Por otro lado, y siendo **la misma geometría**, se ha realizado una malla no estructurada, generada automáticamente mediante el software *Pointwise* (Ilustración 4.10). La dimensión de los *connectors* es la misma en ambos casos.

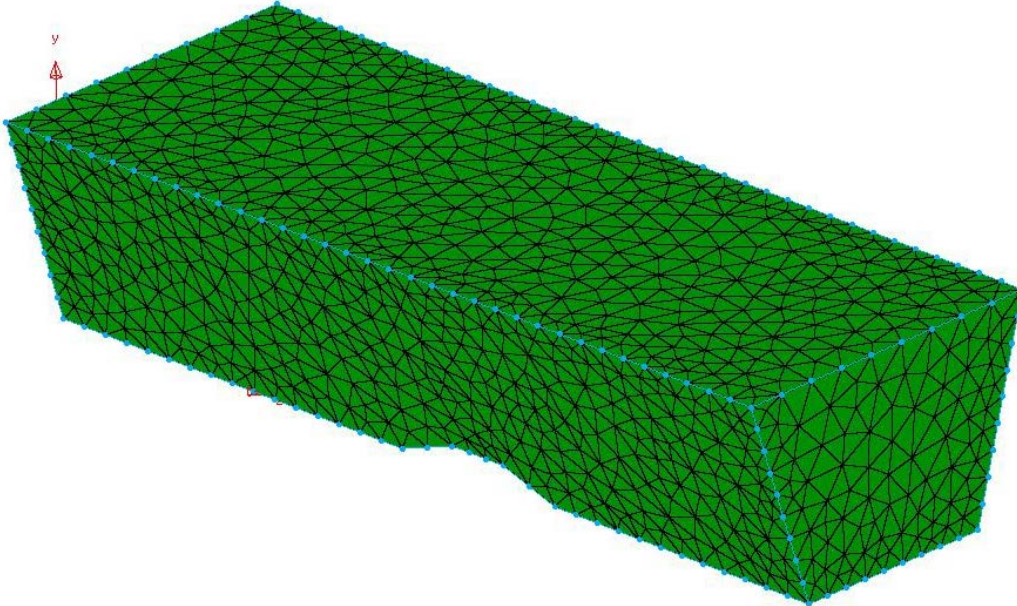


Ilustración 4.10.- Malla no-estructurada de la geometría del canal descrito (17.922 celdillas)

En esta última malla se observa todo lo contrario que en la primera: la mayoría de los vectores normales de las caras no están alineados con la dirección principal del flujo y, aunque la malla está refinada en algunas zonas, la capa límite no va a ser capturada. Además, el número de celdillas de la malla no estructurada es del orden de 6 o 7 veces superior al de la malla estructurada. Esto ocasiona un mayor coste computacional por, al menos, tres motivos:

1. Un número superior de celdillas implica, por norma general, un coste computacional superior, siendo esta proporción no lineal. Eso quiere decir que duplicar el número de celdillas aumenta en un factor superior a dos el coste computacional.
2. Por la propia forma de almacenar la malla en la computadora, al estar definida implícitamente la conectividad de las celdillas de la malla estructurada, la memoria requerida es muy inferior, y menos información debe ser transmitida. Actualmente el cuello de botella puede ser la comunicación entre procesadores, y no la propia potencia de cálculo de los mismos, por lo que las mallas estructuradas son muy superiores en ese sentido a las no-estructuradas

- Como se ha mencionado anteriormente, al no estar alineados los vectores normales de las caras con el vector velocidad del flujo que atraviesa dicha cara, se requieren más iteraciones para que la solución converja, encareciendo computacionalmente la simulación.

Dicho esto, se van a analizar, empleando los indicadores descritos anteriormente, ambas mallas mediante *Pointwise*:

Tabla 4.- Cuadro resumen con los principales indicadores de calidad de malla para el canal analizado. Entre paréntesis, valores promedio; formato regular valores máximos.

		Estructurada	No estructurada
Caras (faces)	Area ratio	7,30 (2,13)	2,43 (1,36)
	Skewness equiangle	0,38 (0,07)	0,58 (0,22)
Celdas (cells)	Volumen ratio	2,59 (1,66)	3,96 (1,54)
	Aspect ratio	7,29 (2,57)	3,04 (1,25)
	Skewness equiangle	0,46 (0,19)	0,72 (0,37)
	Skewness centroid	0,22 (0,05)	0,51 (0,12)

De esta tabla se desprenden unas conclusiones muy interesantes. Por un lado, no se observan grandes diferencias entre unos parámetros y otros. Aunque se podría pensar que la malla estructurada es mejor, dado que sus valores de *skewness* asociados son menores, las relaciones de aspecto, tanto de áreas como de volumen son muy superiores.

Por otro lado, se remarca lo que ya se sabía: el proceso de mallado sin tener información del fenómeno que se está estudiando genera mallas que no están optimizadas. Se puede consultar gran cantidad de bibliografía sobre este tema comparando análisis experimentales con resultados de FVM.

Se debe tener en cuenta que, en este ejemplo, los parámetros de la malla estructurada no se han disparado porque la geometría es altamente regular. No ocurre lo mismo con el cauce objeto de estudio del presente TFM. En este caso la geometría es muy irregular, como ya se ha mencionado en múltiples ocasiones, y los parámetros de calidad de malla asociados empeoran mucho.

5 Análisis de calidad de malla y comparativa con estudios previos

Este capítulo tiene por objetivo comprar las distintas mallas que son objeto de análisis. Las mallas que se van a comprar son tres, a saber:

1. Malla generada por Aranda (2016)
2. Malla realizada según el proceso descrito en *Descripción detallada del proceso seguido en la generación de la malla*. (Malla nº1)
3. Malla optimizada a partir de la anterior y propuesta definitiva del presente TFM. (Malla nº2)

Se van a realizar comparativas empleando tres softwares distintos:

1. Software propio de OpenFOAM: CheckMesh
2. Software de preprocesado de datos de ANSYS Fluent, y

El propio programa que se ha empleado para generar la malla del presente TFM: Pointwise

Posteriormente, y teniendo en cuenta el ejemplo expuesto anteriormente en el que se ponía de manifiesto que no es posible realizar una comparativa de mallas basándose únicamente en indicadores de calidad de malla, sino que es necesario conocer el fenómeno que se está estudiando, así como realizar una inspección visual de la malla; se realiza una comparativa en la que se justifica, de una manera razonada, la idoneidad de la malla propuesta en este trabajo.

5.1 Comparativa de las mallas empleando CheckMesh

CheckMesh es el software que emplea OpenFOAM para la comprobación de mallas computacionales. Se puede emplear tanto para mallas estructuradas como no-estructuradas. Fundamentalmente analiza los siguientes indicadores de calidad de malla:

- Aspect ratio
- Cell non-orthogonality (Ortho-skew)
- Skewness

Los cuales han sido definidos previamente. CheckMesh exporta un fichero de texto plano donde se muestran los parámetros de la malla, e indica si son apropiados o no para realizar una simulación por FVM empleando el software OpenFOAM. A continuación, se muestran los archivos generados a partir de las mallas:

- Malla generada por Aranda (2016)
- Malla propuesta en el presente TFM. (Malla nº2)

```
/*-----*\
| ===== |
| \\ / F i e l d | OpenFOAM: The Open Source CFD Toolbox |
| \\ / O p e r a t i o n | Version: 5.0 |
| \\ / A n d | Web: www.OpenFOAM.org |
| \\ / M a n i p u l a t i o n | |
\*-----*/

Create time

Create polyMesh for time = 0

Time = 0

Mesh stats
  points:          102613
  faces:           249435
  internal faces:  214291
  cells:           75244
  faces per cell:  6.16296316
  boundary patches: 8
  point zones:    0
  face zones:     0
  cell zones:     0

Overall number of cells of each type:
  hexahedra:      57567
  prisms:         3724
```

```
wedges:      0
pyramids:    0
tet wedges:  383
tetrahedra:  9
polyhedra:   13561

Breakdown of polyhedra by number of faces:
```

faces	number of cells
4	3336
5	2357
6	1792
7	260
8	358
9	3405
10	16
12	1282
15	736
18	19

Checking topology...

```
Boundary definition OK.
Cell to face addressing OK.
Point usage OK.
Upper triangular ordering OK.
Face vertices OK.
Number of regions: 1 (OK).
```

Checking patch topology for multiply connected surfaces...

Patch	Faces	Points	Surface topology
ffminx	0	0	ok (empty)
entrada	226	281	ok (non-closed singly connected)
salida	329	419	ok (non-closed singly connected)
ffmaxy	0	0	ok (empty)
ffminz	0	0	ok (empty)
atmosfera	8569	9114	ok (non-closed singly connected)

```

mallaPaperCanon5mdt2011      25499      29061  ok (non-closed singly connected)
mallaPaperCanon5mdt2011_def      521        757  ok (non-closed singly connected)

Checking geometry...

Overall domain bounding box (393659.2916 4211500 174.4115389) (395040
4212637.714 205)

Mesh has 3 geometric (non-empty/wedge) directions (1 1 1)

Mesh has 3 solution (non-empty) directions (1 1 1)

Boundary openness (-2.422806152e-17 -1.099010662e-17 -2.108842194e-15) OK.

Max cell openness = 6.95997997e-16 OK.

Max aspect ratio = 48.73443452 OK.

Minimum face area = 0.3462735243. Maximum face area = 134.8935984. Face area
magnitudes OK.

Min volume = 1.5945719. Max volume = 718.223415. Total volume = 13708145.38.
Cell volumes OK.

Mesh non-orthogonality Max: 64.76381938 average: 19.07246789

Non-orthogonality check OK.

Face pyramids OK.

***Max skewness = 4.387119275, 2 highly skew faces detected which may impair the
quality of the results

<<Writing 2 skew faces to set skewFaces

Coupled point location match (average 0) OK.

Failed 1 mesh checks.

End

```

Reporte de Checkmesh de la malla de Aranda

```

/*-----*\
| ===== |
|  \ \      /  F i e l d      | OpenFOAM: The Open Source CFD Toolbox |
|  \ \      /  O p e r a t i o n | Version: 5.0 |
|   \ \ /   /  A n d           | Web:      www.OpenFOAM.org |
|    \ \ /   /  M a n i p u l a t i o n | |
\*-----*/

```

```
Create time

Create polyMesh for time = 0

Time = 0

Mesh stats

  points:          1019520
  faces:           2966070
  internal faces:  2875272
  cells:           973557
  faces per cell:  6
  boundary patches: 5
  point zones:    0
  face zones:     0
  cell zones:     0

Overall number of cells of each type:

  hexahedra:      973557
  prisms:         0
  wedges:         0
  pyramids:       0
  tet wedges:     0
  tetrahedra:    0
  polyhedra:      0

Checking topology...

  Boundary definition OK.
  Cell to face addressing OK.
  Point usage OK.
  Upper triangular ordering OK.
  Face vertices OK.
  Number of regions: 1 (OK).
```

```

Checking patch topology for multiply connected surfaces...

Patch          Faces    Points  Surface topology
Aire           24963   25488   ok (non-closed singly connected)
Cauce         24963   25488   ok (non-closed singly connected)
Entrada       2067    2160    ok (non-closed singly connected)
Laterales     36738   37760   ok (non-closed singly connected)
Salida        2067    2160    ok (non-closed singly connected)

Checking geometry...

Overall domain bounding box (393605 171.885 -4.21298e+06) (395951 220.022 -
4.20988e+06)

Mesh has 3 geometric (non-empty/wedge) directions (1 1 1)
Mesh has 3 solution (non-empty) directions (1 1 1)
Boundary openness (4.06267e-16 8.4993e-16 5.91707e-17) OK.
Max cell openness = 1.57094e-15 OK.
Max aspect ratio = 159.9 OK.

Minimum face area = 0.116841. Maximum face area = 123.574. Face area
magnitudes OK.

Min volume = 1.34424. Max volume = 79.08. Total volume = 9.79763e+06. Cell
volumes OK.

Mesh non-orthogonality Max: 64.3043 average: 9.63234
Non-orthogonality check OK.
Face pyramids OK.
Max skewness = 2.94103 OK.
Coupled point location match (average 0) OK.

Mesh OK.
End

```

Reporte de Checkmesh de la malla propuesta en el presente TFM

Donde los parámetros más relevantes se recogen en la Tabla 5:

Tabla 5.- Comparativa de parámetros de calidad de malla según CheckMesh

	Aranda	Malla n ^o 2
Max. Aspect ratio	48,7	159,9
Max. Non-orthogonality	64,8	64,3
Avg. Non-orthogonality	19,1	9,6
Max. Skewness	4,39	2,94

5.2 Comparativa de las mallas empleando ANSYS Fluent

ANSYS es uno de los softwares de simulación más empleados, tanto para cálculos estructurales mediante elementos finitos (FEM) como para cálculos de mecánica de fluidos o transferencia de calor, mediante volúmenes finitos (FVM). Consta fundamentalmente de tres partes bien diferenciadas:

- Pre-procesador: en esta etapa se crea o se importa la malla, se establecen las condiciones de contorno y se importan o se crean las propiedades de los materiales que intervienen en la simulación. Esta etapa es la que es de interés en el presente trabajo, pues permite analizar las mallas generadas
- Procesador: implementa multitud de códigos para la resolución de los sistemas de ecuaciones en derivadas parciales que aparecen en los problemas que se simulan
- Post-procesador: permite visualizar los resultados y exportarlos a diversos formatos

Se emplea ANSYS Fluent para comparar mallas puesto que la malla que generó Aranda se almacenó en formato .msh. Este formato no puede ser leído por PointWise y, por tanto, se ha considerado ANSYS Fluent una buena alternativa para realizar la comparativa. A continuación, se muestran isocontornos de:

- Orthogonal Quality
- Cell Equiangle Skew

Los cuales han sido definidos previamente. Tal y como se ha mencionado en anteriores ocasiones, esta comparativa por sí misma no es de gran utilidad. Lo que sí permite es comparar cuantitativamente algunos parámetros de las mallas.

La Ilustración 5.1 muestra el *Equiangle Skew* para las celdillas de la superficie de contorno del cauce para las tres mallas que se analizan. Recuérdese que el valor óptimo es cero. Según la Tabla 3 los valores inferiores a 0,75 son aceptables, lo que significa que todas las mallas son adecuadas. Se observa una mejora notable de la malla n^o2 con respecto a la malla n^o1.

La Ilustración 5.2 muestra el *Ortho skew* para las celdillas de la superficie de contorno del cauce para las tres mallas que se analizan. Recuérdese que el valor óptimo es la unidad, pues indica que los tres vectores descritos $\{ \vec{A}_i, \vec{f}_i, \vec{c}_i \}$ están alineados. Se observa que hay muy pocas celdillas en las que este valor no sea adecuado.

Se podría continuar describiendo parámetros de las mallas del mismo modo que se ha procedido con estos dos: *orthogonal quality* y *equiangle skew*; pero no serían un verdadero reflejo de la calidad de la malla. Esto es debido fundamentalmente a que se están comparando mallas que no son fácilmente comparables entre sí. Por un lado, se tiene la malla que se realizó en el TFG de Aranda es no-estructurada. Esto significa que la mayoría de los parámetros de calidad de malla, por el propio proceso de mallado automático, serán, aparentemente, correctos. Empleando, por ejemplo, la triangulación de Delaunay para la generación de la malla, es prácticamente imposible que se obtengan celdillas oblicuas. Además, por el mismo motivo, la relación de aspecto (*aspect ratio*) será siempre próxima a la unidad excepto, quizás, en algunas zonas altamente irregulares.

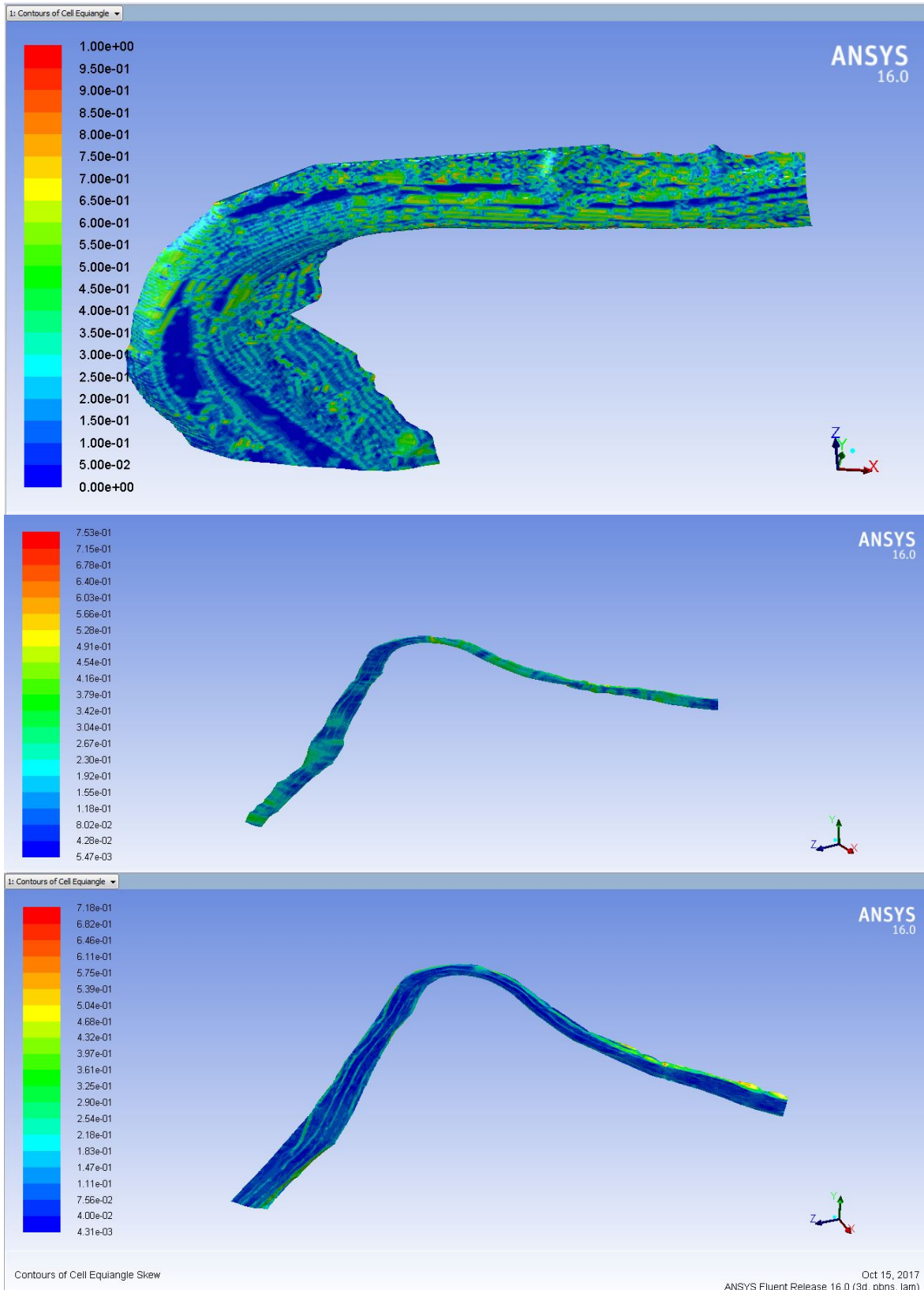


Ilustración 5.1.-Equiangle Skew para las tres mallas descritas anteriormente. Superior: Aranda Ochando, TFG (2016). Intermedia: Malla nº1. Inferior: Malla nº2.

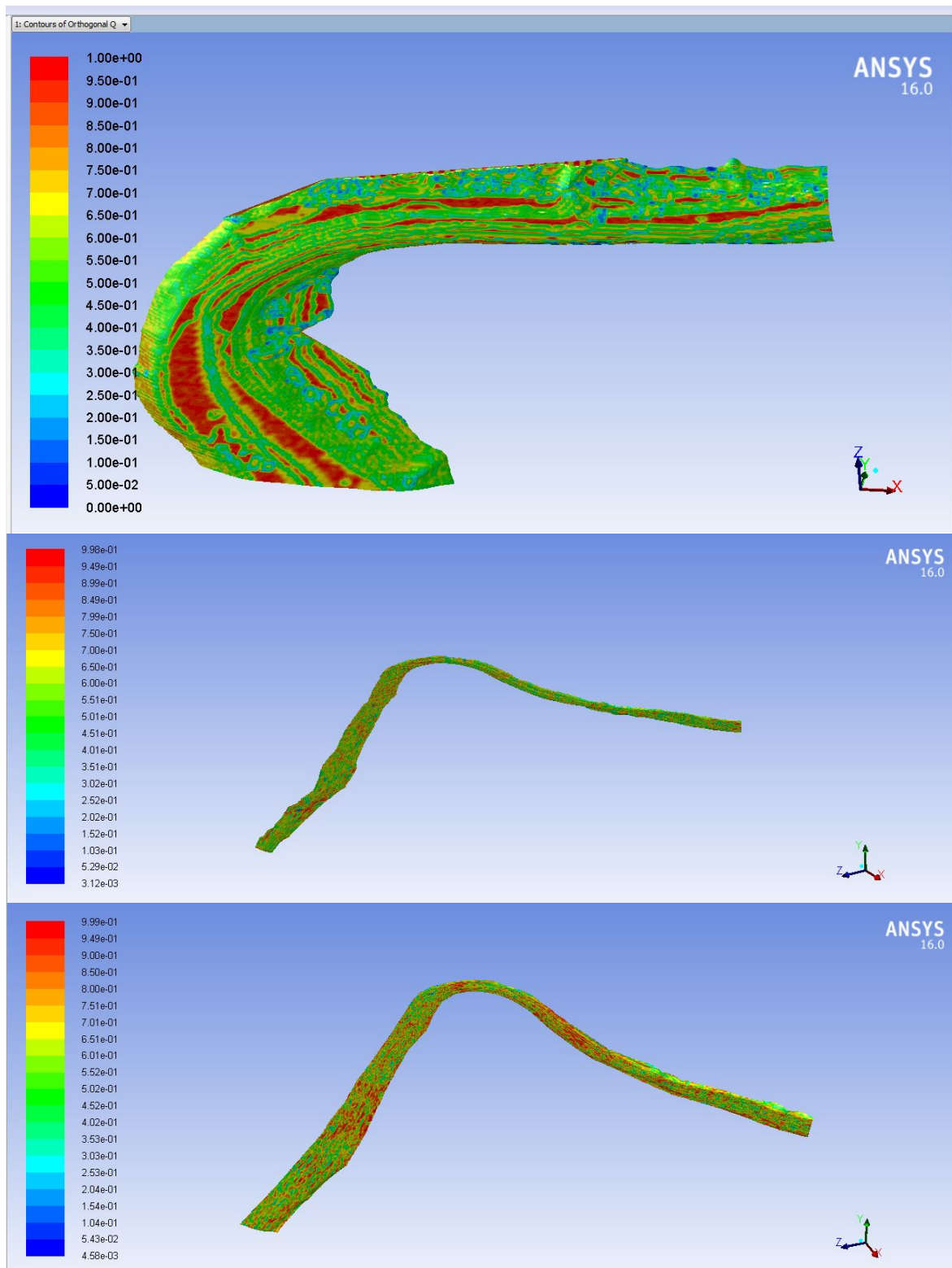


Ilustración 5.2.- Orthogonal Quality (Orto Skew) para las tres mallas descritas anteriormente. Superior: Aranda Ochando (2016). Intermedia: Malla nº1. Inferior: Malla nº2.

5.3 Comparativa de las mallas empleando PointWise

Tal y como se ha mencionado anteriormente, la malla generada por Aranda se encuentra en formato .msh y no es interpretable por PointWise. Se ha realizado un apartado comparando, empleando PointWise, las dos mallas generadas en el presente TFM. Es importante destacar que, debido a que la naturaleza de ambas mallas es similar: misma geometría, mismo problema a analizar, misma tipología de malla, etc... esta comparativa sí es de gran utilidad, pues nos indica qué malla es mejor, así como cuáles son los puntos débiles de cada una de ellas.

Se ha empleado PointWise por diversos motivos. El inmediato es que, al haber sido generada la malla mediante este programa, la importación/exportación de ficheros es trivial. Además, al ser un programa exclusivo para el mallado, presenta muchas más opciones que otros softwares que incluyen más funciones de procesado y postprocesado, y menos de preprocesado. Por último, indicar que, dado que el autor del presente TFM ha adquirido destreza con este software durante la elaboración del TFM, la comparativa puede ser mucho más completa y sencilla de realizar.

Dicho esto, se comparan las dos mallas generadas en el presente TFM. La primera, malla nº1, es la malla generada según la descripción del apartado correspondiente. La segunda malla es una versión optimizada de la anterior (malla nº2).

El proceso de *optimización* de la malla se ha realizado a posteriori de analizar los parámetros de la primera malla. Por un lado, se observó una relación de aspecto de las celdillas excesiva, aunque por sí mismo este fenómeno no supone un gran problema de cara a la simulación. Lo que realmente provocó que se volviese a generar una malla fue el hecho de que, debido a la altísima irregularidad del perímetro inundado de la simulación anterior, la malla estructurada se distorsionaba demasiado intentando adaptarse a la geometría irregular (Ilustración 5.3 e Ilustración 5.4)

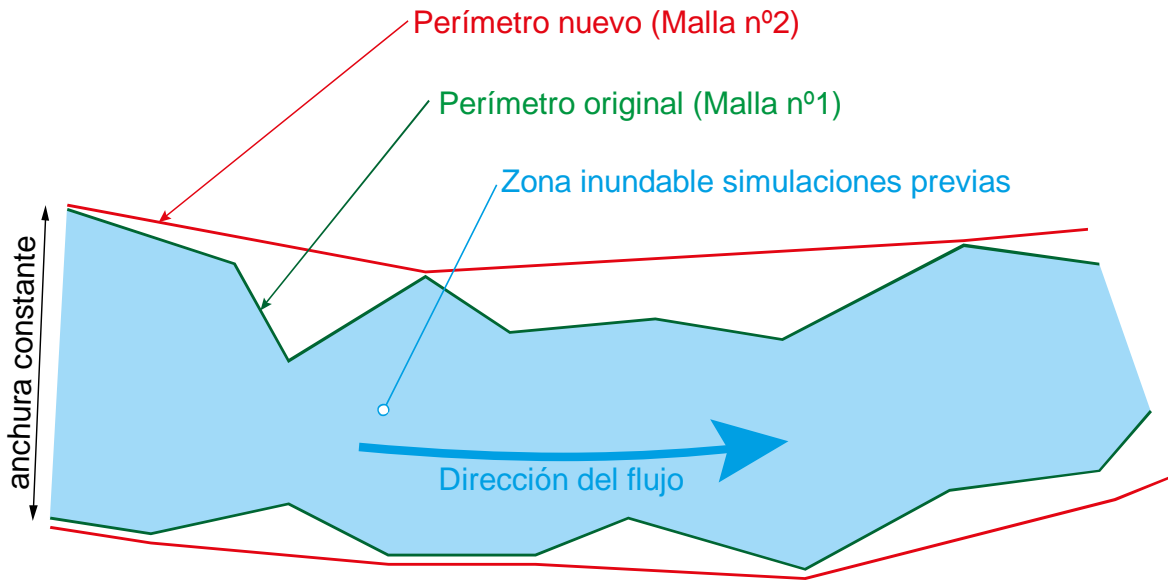


Ilustración 5.3.- Verde: perímetro inundado obtenido a partir de simulaciones previas y de las evidencias de campo. Rojo: perímetro modificado a partir del anterior, buscando regularizar la sección del cauce.

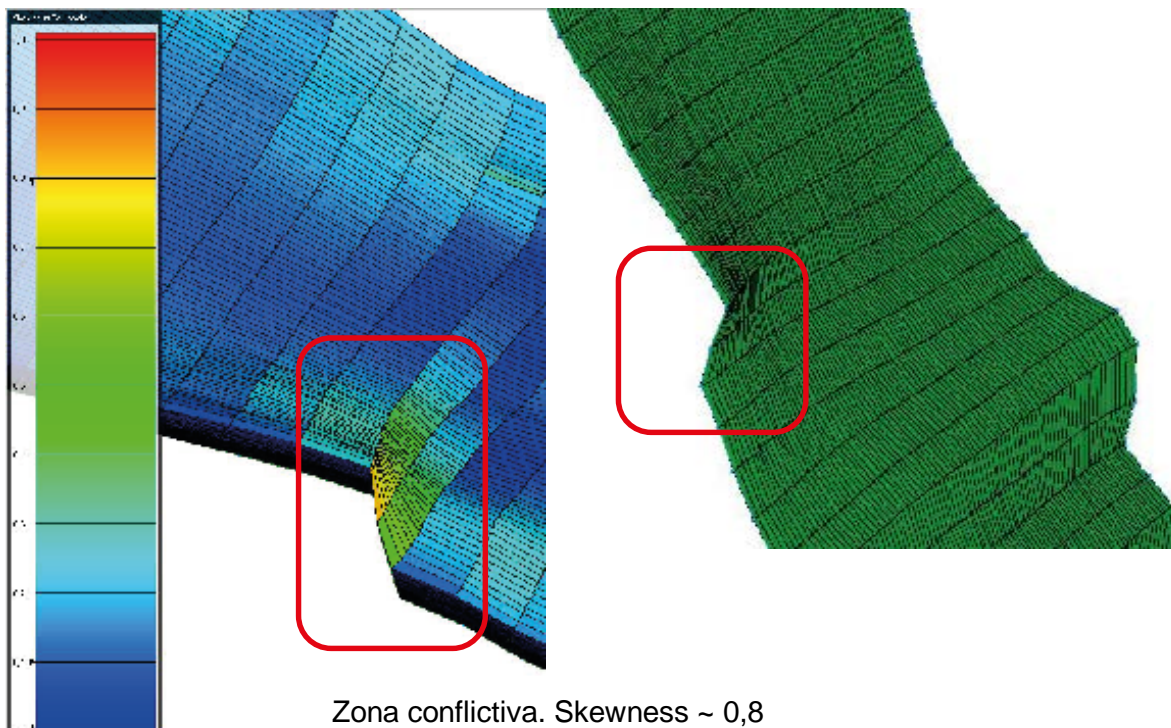


Ilustración 5.4.- Una de las zonas conflictivas debido a la irregularidad del perímetro

Debido a que la relación de aspecto era algo excesiva, se procedió a reducirla de 5 (parámetro empleado para la malla nº1) a 3. Así, la ecuación (7.) se reescribe:

$$\frac{l_x}{l_y} = 3 \quad (19)$$

De este modo, las ecuaciones (4.), (5.), (6.) y (19.) forman un sistema de cuatro ecuaciones con cuatro incógnitas, cuyas soluciones de interés son:

- $n_x = 481$ elementos (frente a los 315 de la malla nº1)
- $n_y = 54$ elementos (frente a los 82 de la malla nº1)

Además, se descompuso en un número superior de *sub-connectors* el perímetro para garantizar la alineación de las celdillas con la dirección del flujo (Ilustración 3.19).

Con todo ello, se generó otra malla, siguiendo el mismo procedimiento que el descrito en el apartado correspondiente de este documento. La Ilustración 5.5 y la Ilustración 5.6 muestran una mejora notable de la calidad de la malla, donde se muestra un histograma del parámetro equiangle skewness de las caras y de las celdillas, respectivamente.

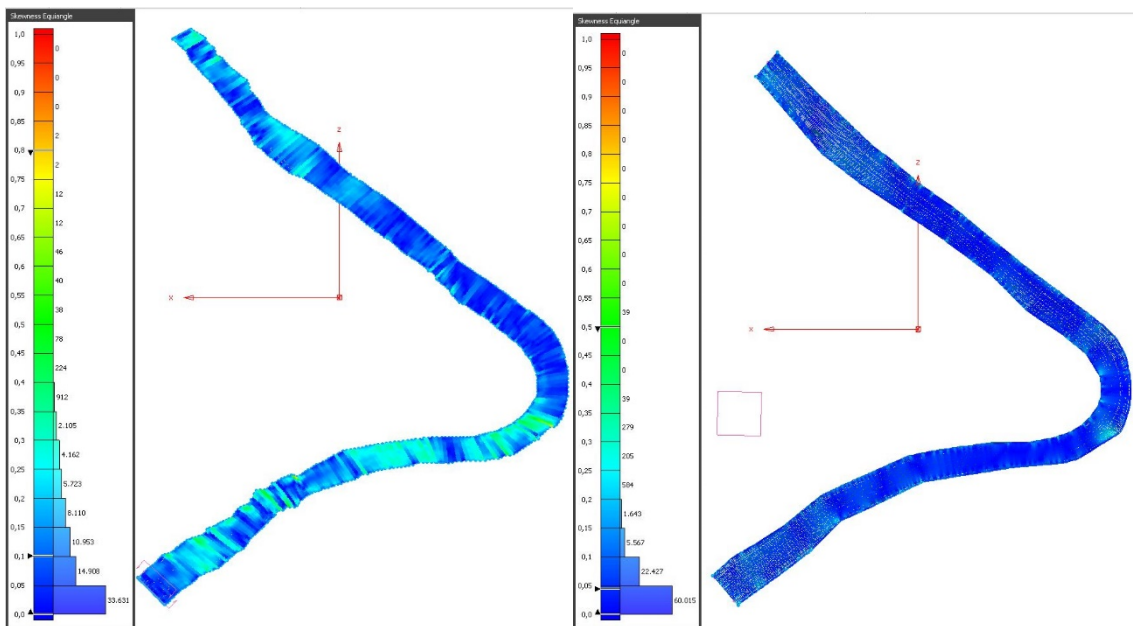


Ilustración 5.5.- Histograma y diagrama de colores del parámetro equiangle skewness de las caras de las celdillas. Izquierda: malla nº1. Derecha: malla nº2

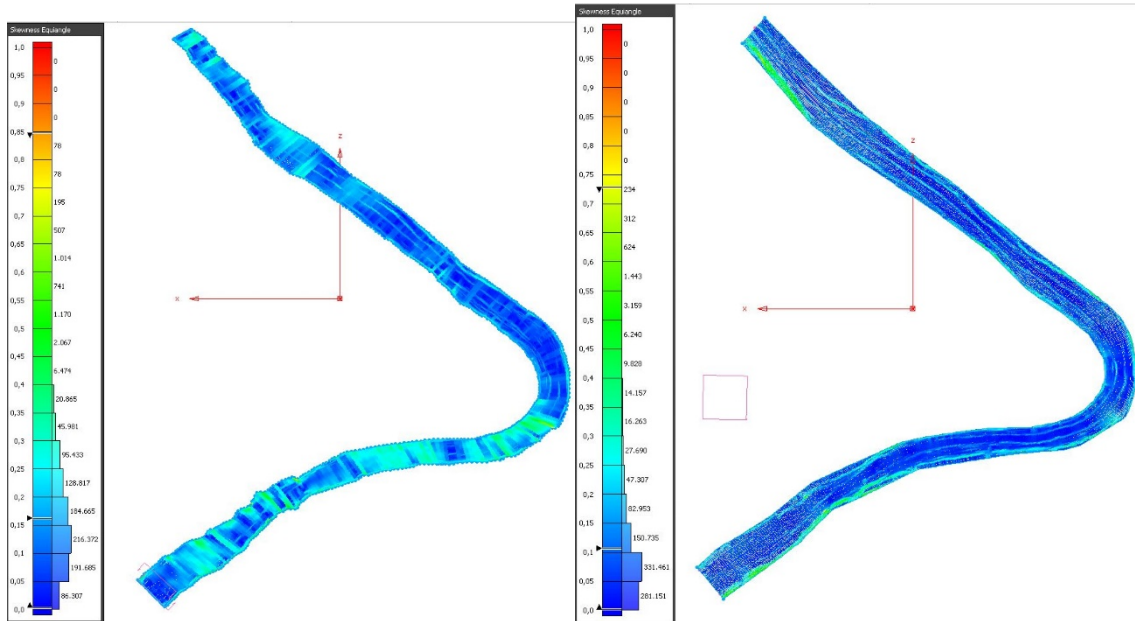


Ilustración 5.6.- Histograma y diagrama de colores del parámetro equiangle skewness de las celdas. Izquierda: malla n°1. Derecha: malla n°2

Se observa que la malla ha mejorado sustancialmente, fundamentalmente por los dos motivos comentados anteriormente. El primero, el perímetro es bastante más regular, lo que permite que la malla estructurada no tenga que deformarse tanto para adaptarse a él. El segundo, y quizás el más sutil, es debido a que se ha descompuesto en un número muy superior de *sub-connectors* el perímetro, persiguiendo que el nodo *i*-ésimo de las márgenes izquierda y derecha estén unidos mediante un *connector* perpendicular a la dirección principal del flujo (véase Ilustración 3.20).

Los histogramas muestran que, efectivamente, tanto los valores máximos como los promedio han disminuido. Se observa, además, que la varianza de estos parámetros se ha reducido notablemente.

A continuación, se muestra una tabla resumen, análoga a la Tabla 4, comparando las dos mallas generadas en el presente TFM:

Tabla 6.- Cuadro resumen con los principales indicadores de calidad de malla para las mallas n°1 y 2. Entre paréntesis, valores promedio; formato regular valores máximos.

		Malla n°1	Malla n°2
Caras (faces)	Area ratio	234 (2,21)	160 (2,37)
	Skewness equiangle	0,80 (0,10)	0,50 (0,044)
	Volumen ratio	3,12 (1,13)	4,28 (1,39)
Celdas (cells)	Aspect ratio	275 (97,7)	248 (61,9)
	Skewness equiangle	0,84 (0,16)	0,73 (0,10)
	Skewness centroid	0,74 (0,045)	0,57 (0,022)

5.4 Comparativa de las mallas mediante inspección visual

Es importante realizar una inspección visual de la malla debido a que los malladores automáticos no tienen en cuenta el fenómeno objeto de estudio. Se basan en que, dada una geometría y unos contornos (curvas en 2D, superficies en 3D), generan una malla según el método para el que estén programadas. Posteriormente, suelen permitir un refinado de la malla en la parte de interés, subdividiendo las celdillas de la zona en cuestión.

Esto los dota de una versatilidad y rapidez enorme, además de que, debido a la naturaleza de los métodos de mallado que implementan, los indicadores de calidad de malla no se suelen disparar, haciendo que generalmente las simulaciones converjan. Lo que ocurre es que los malladores automáticos generan en la práctica totalidad de los casos mallas no estructuradas que pueden ser optimizadas, habitualmente mediante un posprocesado de la malla: refinamiento de celdas en zonas con gradiente de propiedades fluidas importante, haciendo más vasta la malla en zonas donde, o bien no es de especial interés el conocimiento de las propiedades fluidas en dicha zona, o bien el gradiente de éstas es pequeño.

Con todo lo anterior, se puede concluir que, si se conoce a priori una solución aproximada de cómo va a ser el flujo en la zona de estudio, una inspección visual es determinante a la hora de estimar la calidad de la malla, así como la mejora de la misma de cara a una posterior simulación por CFD.

A continuación, se muestran dos mallas. La primera, la malla que generó Aranda y la segunda una parte de la malla que aquí se propone. Se ha recortado aguas arriba y abajo del principal codo para poder compararla con la de Aranda, pues ésta tan sólo era del codo. Se representan ambas mallas según tres vistas, a saber:

- Vista de planta
- Sección de la entrada
- Sección de la salida

Es importante indicar que las secciones de entrada y salida no son las mismas en ambas mallas. Es decir, el plano de la entrada de la malla de Aranda no es el

mismo plano que la sección de la malla aquí propuesta. Tampoco lo son los MDTs de partida. La Ilustración 5.7 y la Ilustración 5.8 muestran vistas en planta de las respectivas mallas:

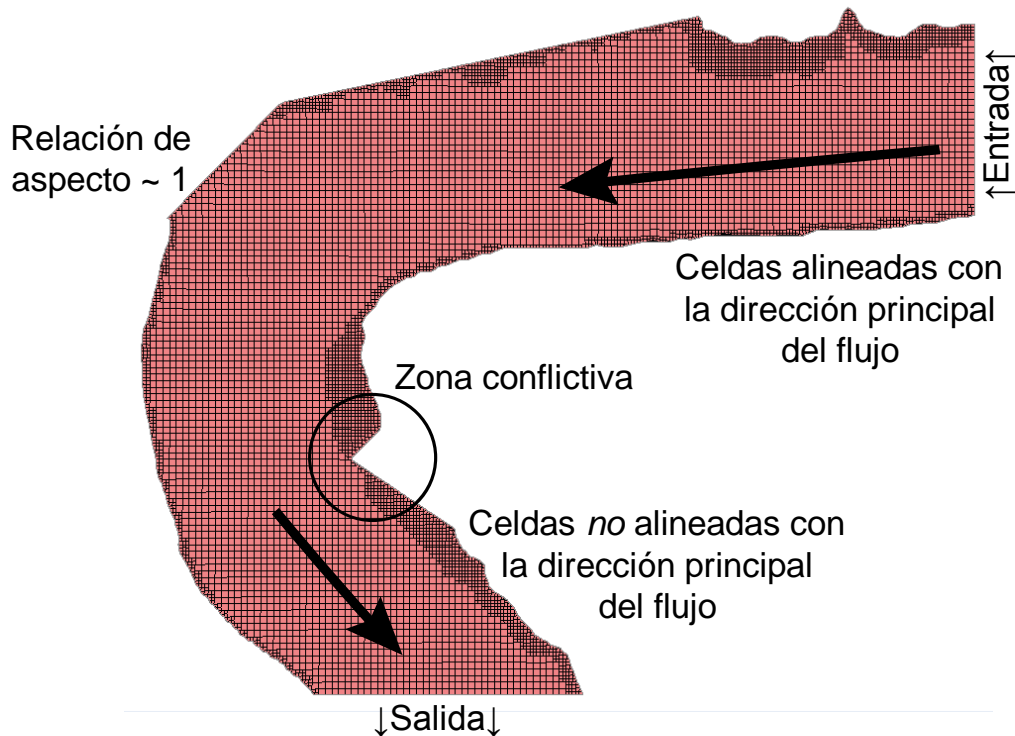


Ilustración 5.7.- Vista en planta de la malla de Aranda (2016)

Por un lado, se observa que el procedimiento para la generación de la malla de Aranda hay sido automatizado siguiendo el procedimiento de SnappyHexMesh (ver Anejo 1: Generación de mallas mediante SnappyHexMesh). Se observan los hexaedros que conforma en núcleo de la geometría y, en las zonas menos someras (las márgenes), debido a la intersección de los hexaedros con el MDT, éstos han sido subdivididos a su vez en elementos de menor tamaño, fundamentalmente tetraedros y poliedros en general.

Se aprecia perfectamente la estructura ortogonal que conforman las celdillas. Las secciones de entrada y salida son perpendiculares entre sí y están alineadas con los ejes X y Z. Por último, se observa una irregularidad importante en la parte interior de codo, señalizada en la ilustración como “zona conflictiva”. Esa región, de cara a una posterior simulación por CFD podría generar problemas de convergencia, debido a la alta irregularidad.

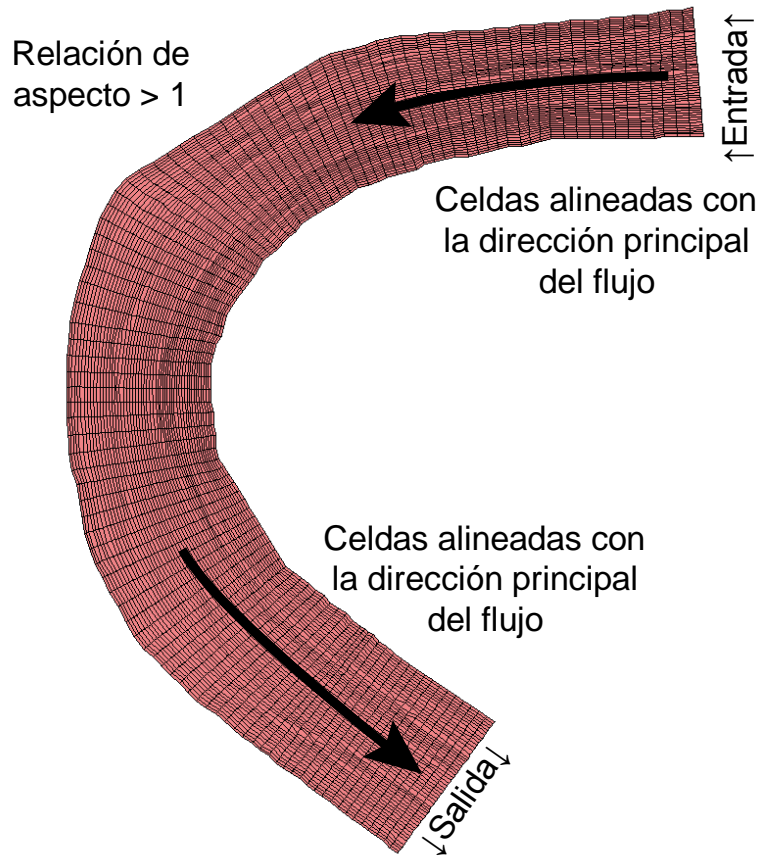


Ilustración 5.8.- Vista en planta de la malla propuesta

En cuando la malla aquí propuesta, se observa que hay alguna distorsión de la relación de aspecto, reduciéndose súbitamente en zonas concretas. Esto es debido, fundamentalmente a la descomposición de los *connectors* de las márgenes. La distribución de los puntos es manual y, por tanto, no es totalmente constante.

A continuación, se muestran las secciones de entrada y salida de ambas mallas. En este caso, la malla estructurada tiene ventajas y también inconvenientes, como se describe a continuación. Es importante destacar que la malla ha sido reescalada con fines gráficos en la coordenada vertical para poder apreciar mejor la sección, ya que, de otro modo, sería mucho más delgada. El factor de escala que se ha aplicado es de 2:

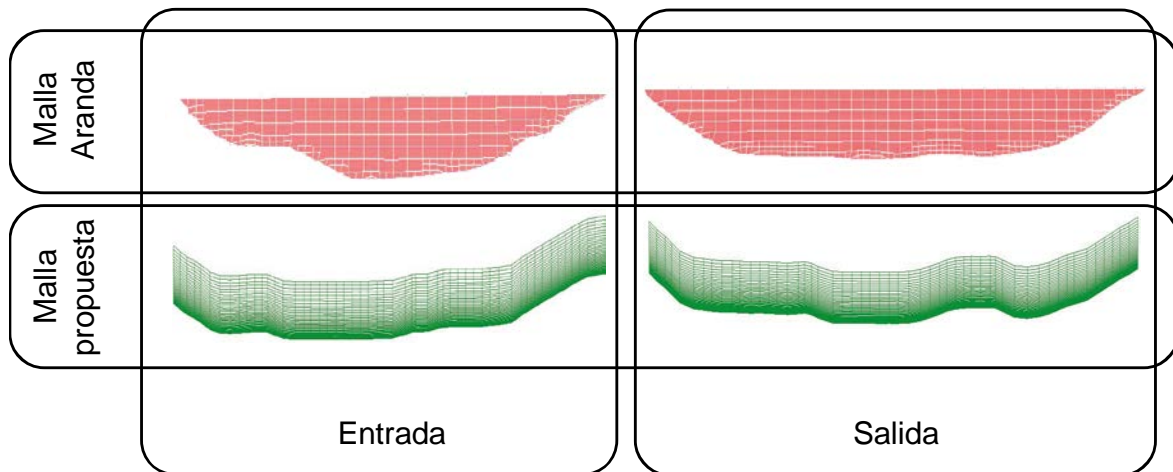


Ilustración 5.9. Vista de las secciones de entrada y salida de ambas mallas. Factor de escala vertical: 2

Se observa que la malla de Aranda es más eficiente en cuanto al número de celdillas se refiere. La malla está mucho más refinada en el perímetro y es más vasta en el centro. Además, se ha cortado por un plano horizontal, eliminando la necesidad de simular las propiedades del aire en zonas que no son objeto de estudio. Esta reducción en el número de celdillas es una importante ventaja con respecto a la malla que aquí se propone.

La malla aquí propuesta, debido a la naturaleza de las mallas estructuradas, no está cortada por un plano horizontal que elimine gran parte del aire. Esto ocasiona que, en el caso de realizar una simulación por FVM, deban determinarse el campo de presiones y velocidades del aire, con el coste computacional que ello conlleva. Por otro lado, la ventaja que tiene la malla estructurada es que permite un aumento progresivo del tamaño de celdilla, capturando el fenómeno de capa límite (Ilustración 1.7). Esto es muy interesante, ya que la potencia de la simulación 3D frente a la 2D radica en la capacidad de capturar fenómenos como la capa límite, el transporte de sedimentos y la componente vertical de la velocidad. Todos ellos fuertemente condicionados por la geometría de la malla en las zonas próximas al cauce. Por tanto, aunque multitud de celdas son redundantes y estarán llenas de aire, es importante que estén ahí, para garantizar la característica de la malla estructurada, que permite un control del mallado muy superior a las mallas no-estructuradas.

5.5 Análisis global de la calidad de la malla propuesta

En base a todo lo expuesto en los apartados anteriores del presente capítulo, se puede concluir que la malla que se expone en el presente TFM es:

- Válida para una simulación por FVM, tal y como ha indicado el software Checkmesh de OpenFOAM
- Adecuada para capturar los fenómenos que pueden ser de interés en una simulación tridimensional transitoria, como puede ser el proceso de llenado del cauce en una avenida de agua, la componente vertical de la velocidad y la capa límite en las proximidades del cauce
- Estructurada, con las ventajas desde el punto de vista del coste computacional que ello conlleva, así como la reducción del número total de celdillas debido a la propia geometría de los hexaedros

6 Funcionamiento de un código CFD

Los códigos CFD (*Computational Fluid Dynamics*) se estructuran alrededor de algoritmos numéricos que permiten abordar problemas de flujo de fluidos. Con objeto de facilitar el manejo de estos potentes códigos comerciales, se suelen incluir interfaces de usuario para la entrada y salida de datos (input-output). Por tanto, los códigos comerciales incluyen tres elementos: pre-procesador, procesador o “*solver*” y post-procesador

A continuación, se describen las funciones principales de cada uno de ellos.

6.1 Pre-procesador

El preprocesado consiste en transformar los *inputs* de un problema de flujo¹ en entradas al *solver* que éste pueda manejar. Las principales etapas son:

1. Definición de la geometría de interés: el *dominio computacional*.
2. Generación de la malla: subdivisión del dominio en un número subdominios pequeños que no se solapen: una malla de celdillas computacionales.
3. Selección de los fenómenos físicos y/o químicos que se desean modelar.
4. Definición de las propiedades de los fluidos.
5. Establecimiento de las condiciones de contorno en los límites del dominio computacional.

La solución de un problema de flujo se obtiene en los nodos dentro de cada celdilla. La precisión de la solución del código CFD depende del número de celdillas de la malla. En general, cuantas más celdas tenga la malla mejor será la precisión y mayor el coste computacional. Las mallas optimizadas normalmente son no-uniformes: refinadas en las zonas donde hay variaciones bruscas de las propiedades fluidas de interés (presión, velocidad, temperatura...) y más vastas donde no hay gran variación de las propiedades fluidas. Por ejemplo, en el análisis de un cauce del río donde el dominio computacional sea la zona inundable, la malla deberá ser mucho más fina en las proximidades del terreno: capa límite, fenómenos de turbulencia,

¹ Un problema de mecánica de fluidos es un caso particular de un problema de flujo. También lo son los problemas de transferencia de calor o masa, entre otros.

arrastre de sedimentos; mientras que en el centro de la sección no necesita ser fina puesto que las propiedades fluidas prácticamente no varían.

Gran parte del tiempo que se emplea en la industria del CFD se dedica a esta etapa de preprocesado: generación del dominio computacional y mallado del mismo.

El presente trabajo fin de máster se centra, fundamentalmente, en las primeras dos etapas del preprocesado, a saber: definición del dominio computacional y generación de la malla.

6.2 Procesador o “*solver*”

Existen tres corrientes distintas para las técnicas de solución numéricas: diferencias finitas, elementos finitos y métodos espectrales. En resumen, los métodos numéricos que implementa un *solver* siguen los siguientes tres pasos:

1. Aproximación de las variables fluidas de interés por funciones simples
2. Discretización por sustitución de las ecuaciones que gobiernan en flujo por aproximaciones y las correspondientes manipulaciones matemáticas
3. Solución de las ecuaciones algebraicas

La principal diferencia entre estas tres corrientes es la manera en que las variables fluidas son aproximadas en el proceso de discretización.

Diferencias finitas. Los métodos de diferencias describen las variables fluidas desconocidas ϕ del problema en los nodos de la malla de líneas coordenadas. Las derivadas de la función ϕ se generan mediante aproximaciones finitas de series de Taylor truncadas en los nodos de la malla y en los vecinos inmediatos. Los operadores diferenciales son sustituidos por diferencias finitas dando lugar a una ecuación algebraica para cada valor de ϕ .

Elementos finitos. Los métodos de elementos finitos emplean funciones a trozos (*piecewise functions*) lineales o cuadráticas normalmente que son válidas en los elementos para describir variaciones locales de las incógnitas ϕ . De este modo, ϕ es solución exacta de la ecuación constitutiva. Si por el contrario, las aproximaciones de ϕ se sustituyen en la ecuación, la solución no será exacta. Se definen unos *residuales*

para medir el error cometido. Posteriormente estos residuales se minimizan. Como resultado se obtiene un conjunto de ecuaciones algebraicas para los coeficientes de las funciones de aproximación. Esta teoría fue desarrollada inicialmente para el cálculo estructural.

Métodos espectrales. Los métodos espectrales aproximan las incógnitas por series de Fourier o polinomios de Chebysev. Al contrario que en los dos métodos anteriores, las aproximaciones no son locales sino *globales*. Del mismo modo, se sustituyen las incógnitas por las series truncadas. Las restricciones que permiten obtener las ecuaciones algebraicas se obtienen de unos residuales ponderados de un modo similar a como se procede en el método de los elementos finitos; o haciendo que la función aproximada coincida con la solución exacta en algunos nodos de la malla.

El método de los volúmenes finitos. El método de los volúmenes finitos fue desarrollado originalmente como una formulación especial de las diferencias finitas. A día de hoy es el método fundamental de resolución de ecuaciones en derivadas parciales en la práctica totalidad de los códigos comerciales de CFD. El algoritmo consta de los siguientes pasos:

1. Integración formal de las ecuaciones constitutivas del flujo sobre todos los volúmenes de control (celdillas computacionales) del dominio.
2. Sustitución de las aproximaciones por diferencias finitas por las ecuaciones de conservación en forma integral donde se describan los procesos de convección, difusión y fuentes/sumideros. Esta discretización convierte las ecuaciones en forma integral en un sistema de ecuaciones algebraico.
3. Solución del sistema de ecuaciones algebraicas por métodos iterativos.

Es el primer paso el que distingue el FVM del resto de métodos numéricos. Este método expresa de forma *exacta* de las ecuaciones de conservación para cada celdilla computacional. La conservación de una variable fluida ϕ , por ejemplo, una componente de la velocidad o la entalpía puede ser expresada como un balance sobre el volumen de control entre varios procesos que tienden a aumentarla o disminuirla. Así, se tiene:

$$\left[\begin{array}{c} \text{Tasa de variación} \\ \text{de la variable } \phi \\ \text{en el volumen de} \\ \text{control con respecto} \\ \text{al tiempo} \end{array} \right] = \left[\begin{array}{c} \text{Flujo neto de } \phi \\ \text{debido a fenómenos} \\ \text{convectivos en el} \\ \text{volumen} \end{array} \right] + \left[\begin{array}{c} \text{Flujo neto de } \phi \\ \text{debido a fenómenos} \\ \text{difusivos en el} \\ \text{volumen} \end{array} \right] + \left[\begin{array}{c} \text{Tasa neta} \\ \text{de creación} \\ \text{de } \phi \text{ en el} \\ \text{volumen} \end{array} \right]$$

Los códigos CFD incorporan técnicas de discretización adecuadas para el manejo de los fenómenos de transporte, convección (transporte debido al flujo del fluido) y difusión (transporte debido a variaciones de ϕ de punto a punto) así como la tasa de variación temporal. La física que hay detrás es compleja y no-lineal de modo que son necesarios métodos iterativos.

6.3 Post-procesador

Al igual que en el pre-procesado, se han hecho muchos esfuerzos en el desarrollo del post-procesado. Debido a la gran popularidad de las estaciones de trabajo, los paquetes CFD están dotados de multitud de herramientas para la correcta visualización de los problemas de flujo. Entre otras herramientas se incluyen:

- Visualización del dominio computacional y de la malla
- Visualización de magnitudes vectoriales
- Visualización del dominio con líneas de contorno
- Representación de superficies en 3D
- Seguimiento de partículas (*particle tracking*)
- Manipulación (traslación, rotación, zoom, etc.)
- Representación gráfica de la solución en vídeo

Una de las grandes ventajas de estas características es que permiten una comunicación de ideas a los no-especialistas.

7 Descripción del proceso seguido en la configuración de la simulación

En este capítulo se describe el proceso seguido en la configuración de la simulación. El software que se ha utilizado para simular el problema objeto de estudio es OpenFOAM (Open Field Operation and Manipulation) que presenta características muy interesantes:

- Actualmente, es uno de los programas de simulación CFD más potentes, siendo utilizado mundialmente para aplicaciones de mecánica de fluidos, transferencia de calor, etc... por multitud de organismos, empresas, instituciones, universidades y particulares.
- Es un software abierto y gratuito, lo que lo convierte en una opción muy atractiva frente a otros programas de simulación, como puede ser ANSYS Fluent.
- Incorpora un tutorial de flujo en canales abiertos, por lo que de cara a la configuración de la simulación no se parte de cero, sino que existe una “plantilla” validada sobre la que trabajar.
- Permite la manipulación de la propia simulación mientras esta se está desarrollando. OpenFOAM es de los pocos programas de simulación que incorporan esta característica.

A continuación, se describe someramente el proceso seguido en la configuración de la simulación para, más adelante, describirlo en detalle. El procedimiento para simular distintos caudales de inundación es idéntico, por lo que tan sólo se describe para un valor concreto, y posteriormente se muestran los resultados para los dos caudales simulados, a saber: 500 y 1.400 m³/s. La estructura general es la siguiente,

1. Se copia la carpeta del tutorial de la simulación del canal abierto: *“tutorials/multiphase/interFoam/RAS/weirOverflow”* con todo lo que se incluye en ella
2. Se modifica la malla, cambiándose los archivos por los que se han exportado de PointWise
3. Se modifican los siguientes parámetros:

- a. Modelo de turbulencia: *SpalartAllmaras*
 - b. Descomposición de la malla para procesamiento en paralelo: *método de Scotch*
 - c. Se selecciona como solver *interFoam*, los parámetros de tiempo toman los valores $t_i = 0$, $t_f = 600$ s y Δt automático, de modo que el número de Courant sea inferior a 0,4
 - d. La tolerancia de los residuales se deja por defecto
4. El parámetro *alpha.water* de cada celdilla se inicializa según se describe en el apartado correspondiente.
 5. Se aplican las condiciones de contorno de los caudales

A continuación, se describe el procedimiento de la configuración de la simulación en detalle. Lo primero que se realiza es la copia de la carpeta completa *tutorials / multiphase / interFoam / RAS / weirOverflow* del de OpenFOAM.

De un modo paralelo, se exporta la malla generada en PointWise. Es recomendable asignar las condiciones de contorno en PointWise (Ilustración 7.1), aunque se puede realizar también modificando los archivos de texto plano.

Set	#	Name	CAE Type	ID
<input type="checkbox"/>	0	Connection	Connection	Connection
<input type="checkbox"/>	0	Unspecified	Unspecified	Unspecified
<input type="checkbox"/>	1	Entrada	patch	1
<input type="checkbox"/>	1	Salida	patch	2
<input type="checkbox"/>	1	Cauce	wall	3
<input type="checkbox"/>	1	Aire	patch	4
<input type="checkbox"/>	2	Laterales	patch	5

Ilustración 7.1.- Cuadro con las definición del tipo de condición de contorno para cada *domain* de la malla.

Para exportar la malla, se selecciona el *block* y se hace click en *File > Export > CAE...* y se crea una carpeta que llamaremos *polyMesh*. Seleccionamos dicha carpeta y PointWise creará los siguientes archivos:

- *boundary*: contiene las condiciones de contorno de cada una de las superficies del contorno. En caso de no haber marcado el tipo de condición de contorno en PointWise, aquí se deberán modificar. En esta

simulación, la superficie denominada *cauce* será tipo *wall* mientras que las demás serán tipo *patch*.

- *faces*: indica cuáles son los puntos que conforman cada cara. Puesto que la malla es estructurada de hexaedros, todas las caras están definidas por los índices de los 4 puntos que conforman los vértices del cuadrilátero
- *neighbour*: indica las celdas vecinas a cada celda
- *owner*: indica las caras que definen cada celda
- *points*: contiene las coordenadas espaciales de cada uno de los nodos (*points*)

Además de una carpeta vacía llamada *sets*. A continuación, todo el contenido de la carpeta *polyMesh* que hemos copiado del tutorial de OpenFOAM lo sustituiremos por los archivos que hemos generado con PointWise. De este modo, le hemos introducido la malla a OpenFOAM.

Puesto que el eje vertical es el eje *y*, tanto en el tutorial como en PointWise, no se modifica el fichero *g* que contienen al vector aceleración de la gravedad. El fichero *transportProperties* tampoco se modifica, pues los valores por defecto de tipo de fluido (newtoniano), viscosidad cinemática (*nu*) y densidad (*rho*), así como la tensión superficial (*sigma*) se consideran apropiados para la simulación.

En cuanto al modelo de turbulencia, dentro del fichero *turbulenceProperties* se cambia el modelo de turbulencia $\kappa - \varepsilon$ por uno más sencillo, en este caso el modelo de turbulencia de Spalart-Allmaras:

```
RASModel    kEpsilon    →    RASModel    SpalartAllmaras
```

A continuación, se modifican los siguientes archivos, dentro de la carpeta *system*:

- *decomposePartDict*: el método para dividir la malla y poder procesarla en paralelo será automático, pues la malla es altamente irregular. Así, al principio del fichero debe quedar:

```
method      scotch;
```

- *controlDict*: de este fichero tan sólo se modifica el parámetro *writeInterval*, pasando de valer 2 a 25. Esta modificación, de cara a la simulación no

afecta, pero sí reduce considerablemente la cantidad de información que se almacena una vez ha finalizado la simulación.

- `fvSchemes`: este fichero tampoco se modifica
- `fvSolution`: de este archivo se modifica lo siguiente:

```
cAlpha          1          →  cAlpha          2;
nCorrectors     3;        →  nCorrectors     4;
nNonOrthogonalCorrectors 0; →  nNonOrthogonalCorrectors 1;
```

Según Bohórquez, (2010), al aumentar el parámetro *cAlpha* por encima de la unidad, se comprime la interfase agua-aire y se reduce la difusión numérica. Por otro lado, el parámetro *nCorrectors* es determinante a la hora de la convergencia de la simulación. Se ha adoptado el valor 4 debido a que la simulación no convergía con el valor 3, que venía por defecto. Además, se ha modificado *nNonOrthogonalCorrectors* adoptando el valor de la unidad.

En base a los tres parámetros anteriores, la ecuación de la presión se resuelve ocho veces en cada paso de tiempo, haciendo que la simulación converja.

Lo siguiente será modificar los ficheros de las variables fluidas dentro de la carpeta "0". La malla del tutorial, consta de 4 superficies de contorno, mientras que la simulación que se está configurando consta de cinco. La diferencia radica en que, la superficie *atmosphere* del tutorial, en el caso del cauce se subdivide en dos, aunque ambas son idénticas de cara a la simulación. Estas son: *aire* y *laterales*. Es por ello que es necesario modificar todos los ficheros dentro de la carpeta 0, cambiando el nombre de las superficies, e incluyendo bien *aire* si se ha sustituido *atmosphere* por *laterales* o al contrario: incluyendo *laterales* si se ha sustituido *atmosphere* por *aire*. Los archivos que deben modificarse atendiendo a este criterio son:

- `epsilon`
- `k`
- `nut`
- `p_rgh`
- `U`

En el caso de la velocidad, U , se modifica la condición de contorno a la salida. En el caso del tutorial, se fija este parámetro a *zeroGradient*, de modo que no se impone que $Q_{entrada}=Q_{salida}$. Obviamente, en régimen estacionario, se cumplirá esta igualdad. En la simulación que se configura en el presente TFM se va a imponer explícitamente la condición de igualdad, por lo que los parámetros serán iguales a la salida que a la entrada, con la condición de que *flowrate* será igual a *inletFlowRate* a la entrada y *outletFlowRate* a la salida. Estos parámetros se definen en el archivo *initialConditions* que se encuentra en “0/include”

Por otro lado, es necesario incluir el archivo *nuTilda*, pues es la variable que se ha introducido al considerar el modelo de turbulencia de Spalart-Allmaras. Esta ($\tilde{\nu}$) será la incógnita que se resolverá, en vez de κ y ε . En cualquier caso, el fichero *nuTilda* es idéntico a *nut*.

Dentro de la carpeta “0” hay otra carpeta llamada *include*, que contiene al fichero *initialConditions*. Este archivo se debe modificar, incluyendo el parámetro *outletFlowRate*. La siguiente tabla muestra el valor de los parámetros *inletFlowRate* y *outletFlowRate* en función del caudal de la inundación que se considere:

Tabla 7.- Valor de los parámetros *inletFlowRate* y *outletFlowRate* en función de la simulación que se considere

Caudal (m ³ /s)	<i>inletFlowRate</i>	<i>outletFlowRate</i>
500	500	-500
1.400	1400	-1400

El resto de parámetros de *initialConditions* se deja con los valores por defecto. El siguiente paso es uno de los más importantes de la simulación, y es determinante de cara a que la misma converja en un periodo de tiempo razonable. Lo que se persigue es inicializar la solución del parámetro *alpha.water* o, lo que es lo mismo, indicar el estado inicial de “llenado” del cauce para el caudal considerado de modo que no sea necesario simular el proceso de llenado, sino que se parta directamente de una condición próxima a la solución. Para ello, se emplean los datos de las simulaciones realizadas por Bohórquez (2016), que se muestran en la Ilustración 7.2:

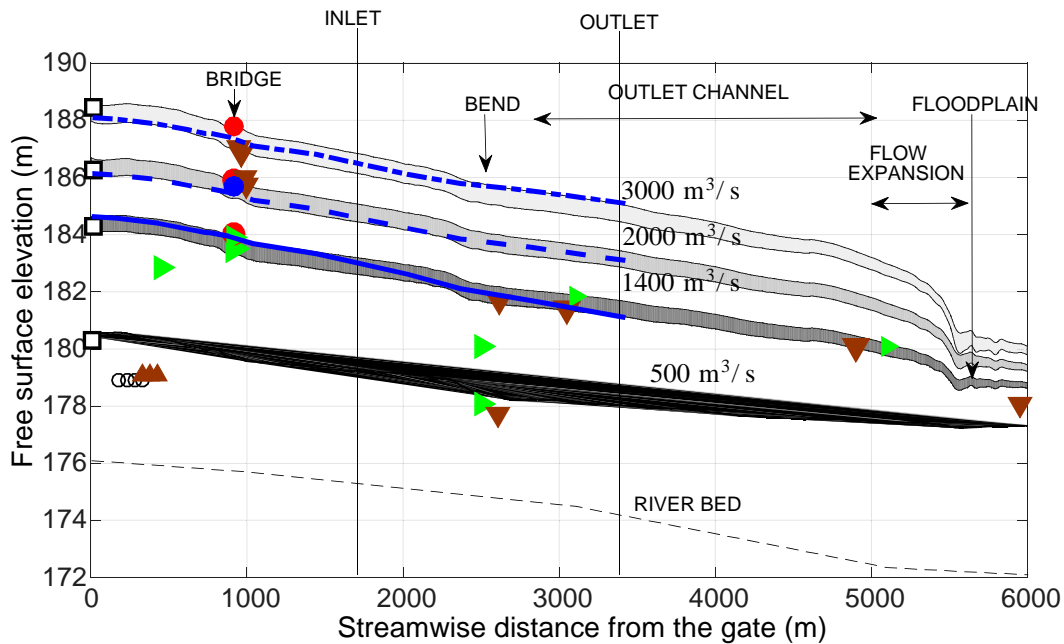


Ilustración 7.2.- Elevación de la superficie libre de la lámina de agua en los 6 km aguas abajo de la presa de Marmolejo. Fuente: Bohórquez (2016)

A continuación, se recogen en una tabla los valores numéricos correspondientes a la elevación de la superficie libre de la lámina de agua para los caudales considerados en las secciones de entrada y salida. Es importante destacar que en el presente TFM tan sólo son de interés los caudales de 500 y 1.400 m³/s:

Tabla 8.- Datos de elevación de la superficie libre de la lámina de agua a la entrada y a la salida del dominio computacional. Fuente: Bohórquez (2016)

Caudal (m ³ /s)	Entrada	Salida
500	179,1	177,9
1.400	183,0	181,6
2.000	184,9	183,4
3.000	186,7	185,0

Lo que se persigue realizar es lo siguiente:

- Se determinan 3 puntos que pertenezcan a la superficie libre de la lámina de agua para las dos simulaciones consideradas
- A partir de dichos puntos, se obtiene la ecuación del plano que los define
- Mediante la utilidad *funkySetFields* inicializamos el valor de *alpha.water* para cada una de las celdillas. Esta utilidad toma, en el caso que nos ocupa, como argumento el plano que define la interfase y, por tanto, divide

la malla en dos. A continuación, asigna a la variable que se considere (*alpha.water* en este caso) un valor para cada parte del dominio. Así, las celdas que están por debajo del plano tomarán un valor de *alpha.water* igual a la unidad, y el resto valdrán cero. Un punto (centroide de celdilla) está por debajo del plano si para ese mismo x,z el valor de y es inferior al correspondiente valor que se obtendría sustituyendo x,z en la ecuación del plano (Ilustración 7.3).

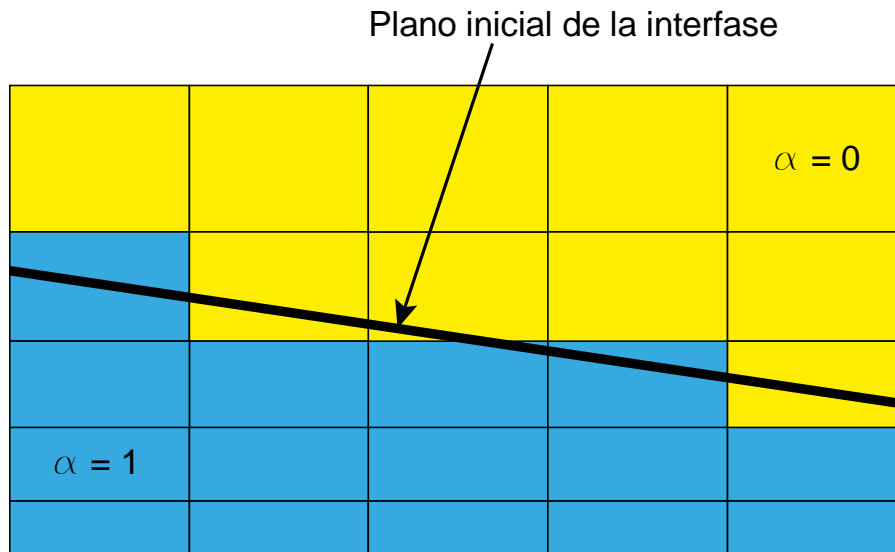


Ilustración 7.3.- Esquema ilustrativo del funcionamiento de *funkySetFields* y la inicialización del parámetro *alpha.water*

Es importante destacar que para poder emplear *funkySetFields* es necesario instalar la versión de OpenFOAM 4.0 (*foam-extend 4.0*)

- Se genera el archivo *alpha.water* que contiene la solución inicial de dicha variable.

Por último, se ejecuta la simulación. Para ello, se descompone el dominio con el comando `decomposePar` y posteriormente se lanza la simulación en paralelo con el comando `foamJob -p interFoam`.

La Ilustración 7.4 muestra un esquema general del proceso seguido en la configuración de la simulación:

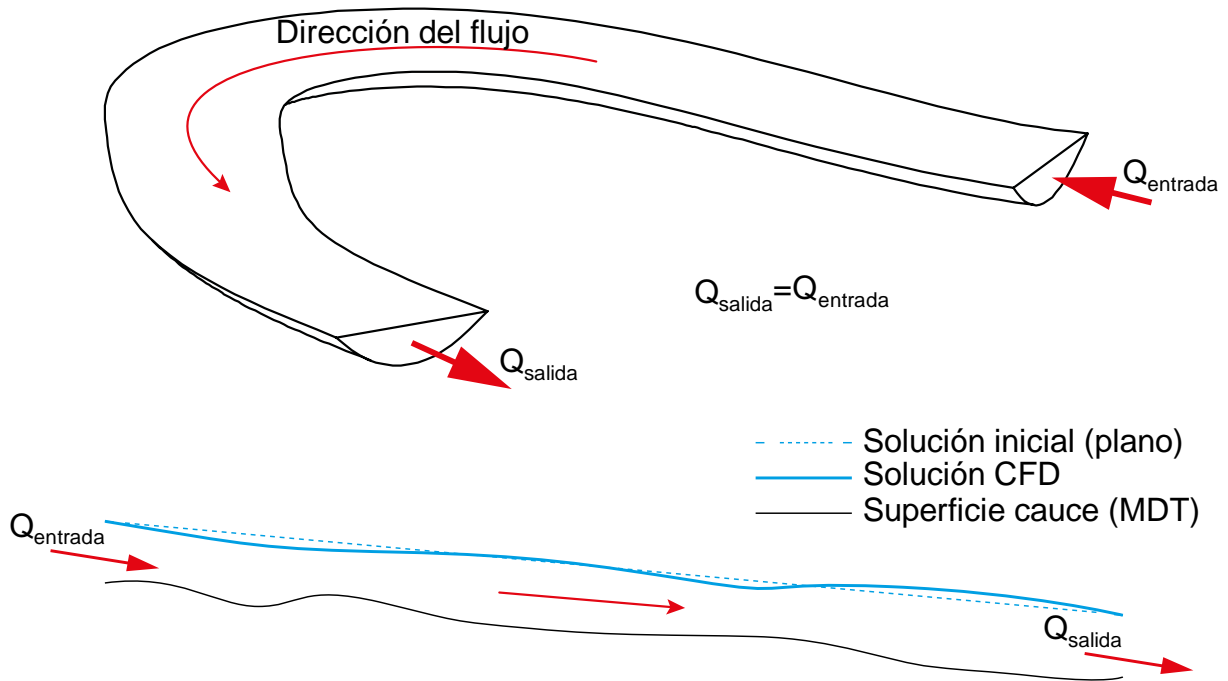


Ilustración 7.4.- Esquema general seguido en el proceso de configuración de la simulación

Por último, se exportan los datos a *EnSight*, que será el programa para el postprocesado de los mismos. Para ello, se ejecuta la utilidad: *foamToEnSight*.

8 Resultados de la simulación y comparativa con estudios previos

Comienza el presente capítulo con la descripción del software empleado para el postprocesado de los datos de la simulación para, posteriormente, mostrar los resultados más relevantes de la simulación. Por último, se comparan los resultados obtenidos en el presente estudio con los que se obtuvieron en ocasiones previas, concretamente con los resultados de Aranda (2016).

8.1 Software empleado para el postprocesado: EnSight

Al ejecutar la utilidad *foamToEnSight*, se crea una carpeta llamada *EnSight*, en el directorio de la simulación. Esta carpeta incluye datos de todas las variables fluidas de interés, a saber:

- *alpha.water*
- *nut*
- *nuTilda*
- *p_rgh*
- *U*

Para cada uno de los instantes de tiempo en los que se han guardado los datos de la simulación (valor de *writeInterval* del fichero *controlDict*, dentro de la carpeta *system* de la simulación). Además, incluye un fichero *mesh* que contiene los datos de la malla. El formato de estos archivos es binario, lo que significa que no se pueden leer con un editor de texto, al contrario que los archivos en formato *ascii*, que se pueden abrir con el bloc de notas o cualquier otro editor de texto. Por tanto, tan sólo se pueden interpretar mediante el software EnSight.

El software EnSight es un software de postprocesado de datos procedentes de simulaciones CFD, simulaciones FEM o, en general, cualquier software de CAE. Es muy utilizado por un gran número de universidades a nivel mundial, así como multitud de industrias importantes como son la NASA, Airbus, BMW, Boeing y Ford, entre muchas otras. Como se ha comentado anteriormente, OpenFOAM permite la exportación de los resultados de las simulaciones a este software, por lo que nos es

de gran utilidad. Bien es cierto que se podía haber optado por emplear el software de postprocesado propio de OpenFOAM: ParaView, pero se ha elegido EnSight por ser un software dedicado exclusivamente a postprocesado CAE.

8.2 Resultados de la simulación. $Q = 500 \text{ m}^3/\text{s}$

A continuación, se muestran los resultados que se han obtenido tras simular el proceso de inundación de la zona objeto de estudio, cuando el caudal que desembalsa la presa de Marmolejo es de $500 \text{ m}^3/\text{s}$.

La Ilustración 8.1 muestra los valores del parámetro α en las secciones de entrada y salida. Un valor de α igual a la unidad (rojo) indica que toda la celdilla computacional está llena de agua, mientras que un valor nulo (azul) indica que se encuentra ocupada por aire. Un valor intermedio, por tanto, indicará que parte de la celdilla computacional se encuentra llena de agua, y el resto de aire.

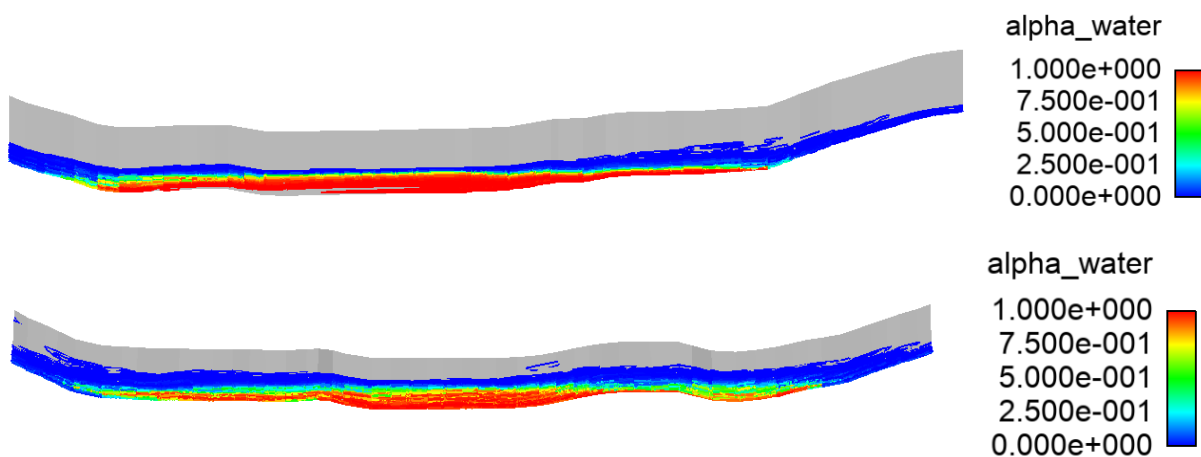


Ilustración 8.1.- Valores del parámetro α para un caudal de $500 \text{ m}^3/\text{s}$. Superior: sección de entrada. Inferior: sección de salida

Se observa que la distribución del parámetro α a la entrada es mucho más homogénea que a la salida. Esto se debe a que se están analizando las propias secciones de entrada y salida, y debido a la naturaleza de los métodos numéricos (tanto FVM como FEM), los resultados en las secciones del contorno no suelen, con carácter general, representar fielmente la solución del problema. El principio de Saint-Venant, que se aplica en problemas de elasticidad, también se puede aplicar a problemas de flujo. Éste establece que, a una distancia suficientemente alejada de las condiciones de contorno, es independiente cómo se apliquen dichas condiciones. En

el caso que nos ocupa, significa que sea cual sea la forma en la que el caudal entra al dominio computacional, a una distancia moderadamente alejada de la sección de entrada, la distribución de las variables fluidas de interés es independiente de la forma en la que el agua entra en la malla. En la configuración de la simulación se indica la forma en la que el agua entra a la malla. Esta información se encuentra en “*O/U*” indicando que la condición de contorno del parámetro U en la entrada es *variableHeighFlowRateInletVelocity*.

Por otro lado, y basándonos en la propia definición del parámetro α se puede obtener la superficie de la lámina libre del agua, o lo que es lo mismo, la interfase del agua y el aire, como el lugar geométrico de las celdillas donde $\alpha = 0,5$. De este modo, se obtiene la zona inundable para un caudal de 500 m³/s, como muestra la Ilustración 8.2:

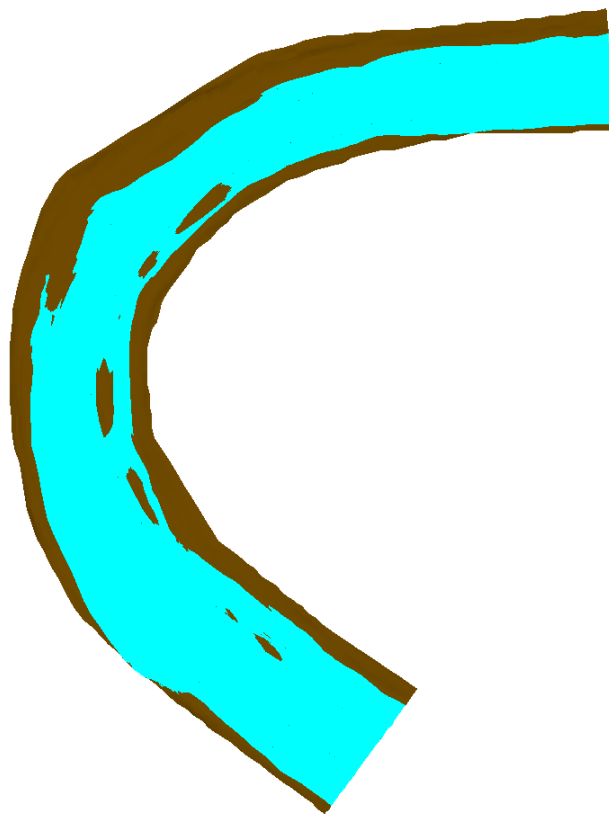


Ilustración 8.2.- Superficie inundada para un caudal de 500 m³/s.

Otra de las muchas opciones de postprocesado CFD que nos permite EnSight es obtener las líneas de corriente. Las líneas de corriente nos muestran, al menos de un modo cualitativo, de cuál es la dirección principal de flujo, así como donde son las zonas de mayor calado. Se observa en la Ilustración 8.3 que la mayor parte del flujo

circula por el cauce natural del río, mientras que un caudal menor lo hace por la zona inundada. Una manera de estimar el caudal, de un modo totalmente cualitativo, es mediante la densidad de líneas de corriente: a mayor densidad de líneas de corriente, mayor será el caudal que circula por dicha zona.

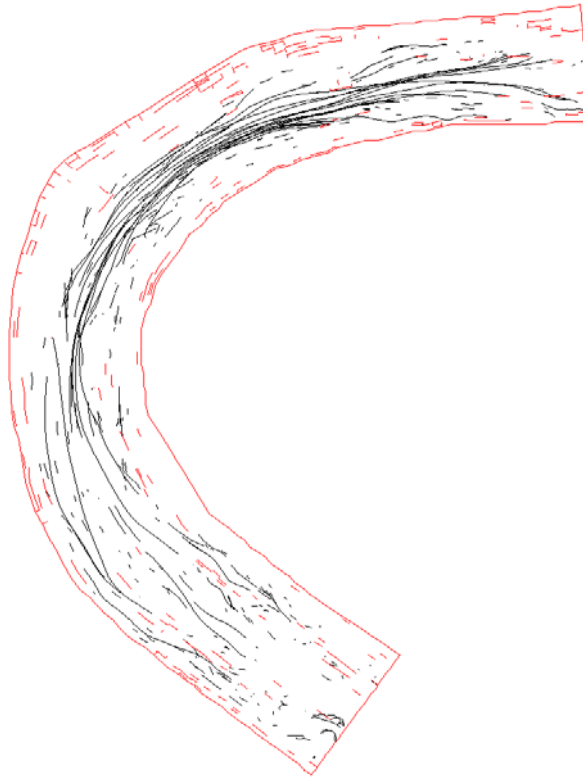


Ilustración 8.3.- Líneas de corriente de algunas partículas de agua para un caudal de 500 m³/s

Además de las líneas de corriente, se puede generar un diagrama que contenga a los vectores velocidad de las partículas de la superficie libre del agua. Esto nos permite tener una idea de cuál es la distribución de velocidades, tanto en módulo como en dirección. Los resultados se muestran en la Ilustración 8.4. Se observa, tanto en la Ilustración 8.3 como en la Ilustración 8.4 (superior) que, efectivamente, la malla está alineada con la dirección principal del flujo. Esto valida, por tanto, las hipótesis que se realizaron en apartados anteriores donde la dirección del flujo se suponía conocida.

La Ilustración 8.4 (inferior) muestra los isocontornos de velocidad para las partículas de la superficie libre del agua. Se observa que el flujo a lo largo del codo dista mucho de ser unidimensional, localizándose velocidades próximas a los 8 m/s en determinados puntos.

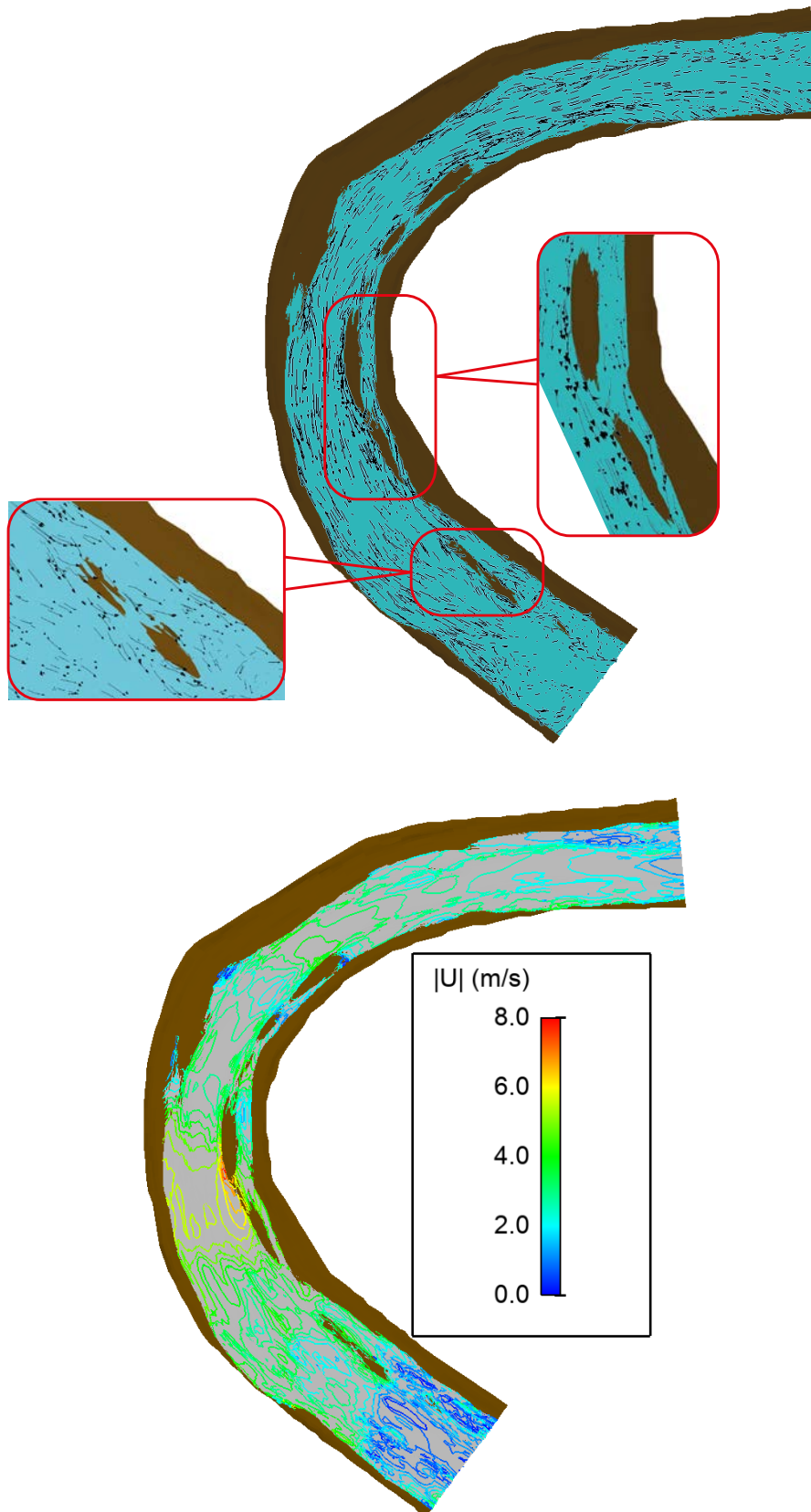


Ilustración 8.4: Velocidad de las partículas de la superficie libre del agua para un caudal de 500 m³/s. Superior: Vectores de velocidad. Inferior: isocontornos de velocidad.

Realizando un corte con un plano vertical en el codo, se obtienen los siguientes vectores de velocidad, así como el parámetro α en la sección:

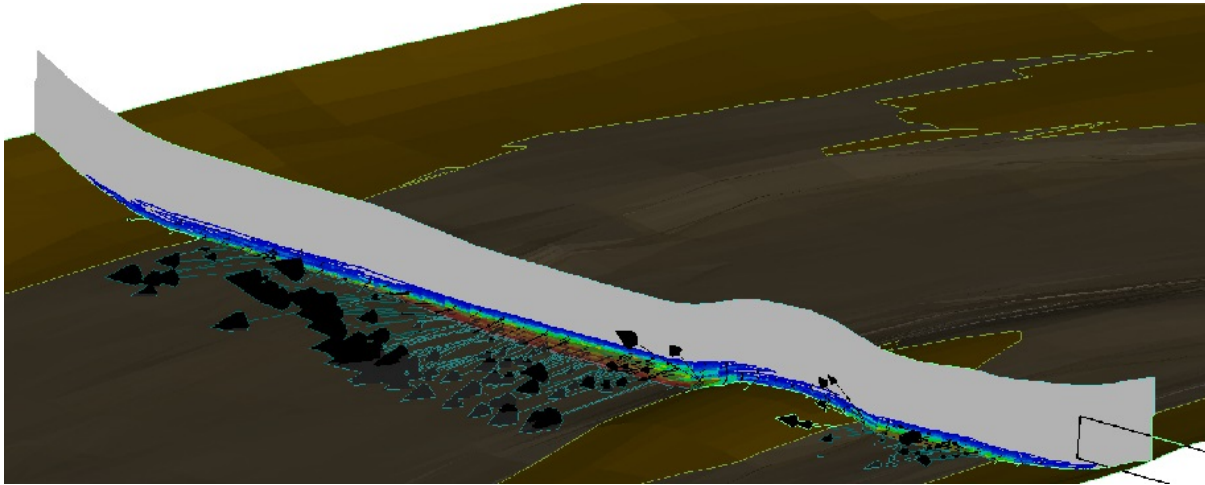


Ilustración 8.5.- Vectores de velocidad en una sección del codo

La Ilustración 8.5 es quizás una de las más interesantes de este trabajo, pues en ella se aprecia la práctica totalidad de lo que se ha venido exponiendo a lo largo del presente TFM.

Por un lado, se observa la irregularidad del cauce, con zonas más o menos planas, pendientes pronunciadas en los extremos y zonas donde se separa el flujo. Esto último obliga a que la simulación sea, al menos, bidimensional, según establece la Guía Metodológica para el desarrollo del SNCZI, en su capítulo séptimo.

Por otro lado se observa el fenómeno de difusión numérica que, aunque la malla es estructurada y se ha modificado el parámetro $cAlpha$ en la configuración de la simulación, es inevitable que ocurra, debido al carácter multifásico y altamente turbulento del flujo.

Por último, aunque no menos importante, se observa que el flujo tiene una componente vertical de la velocidad que, aunque es pequeña en magnitud, no es despreciable. Esta componente tan sólo se puede capturar con simulaciones tridimensionales y, como se ha mencionado anteriormente, es importante en fenómenos erosivos y de transporte de sedimento, como son los que tienen lugar en la presente zona de estudio.

8.3 Comparativa con estudios previos

En este apartado se van a comparar los resultados obtenidos en el presente TFM con los resultados que obtuvo Aranda (2016). Para ello, se ha simulado la misma zona y el mismo caudal. El caudal de las inundaciones es de $1.400 \text{ m}^3/\text{s}$. Es importante recordar el título del presente trabajo: *influencia de la calidad de malla y MDE en la simulación multifásica 3D del flujo del río Guadalquivir*, puesto que es en este apartado donde se determina la influencia de la calidad de la malla. A continuación, se muestran ilustraciones análogas a las que se observan en el apartado anterior, pero en este caso para un caudal de $1.400 \text{ m}^3/\text{s}$.

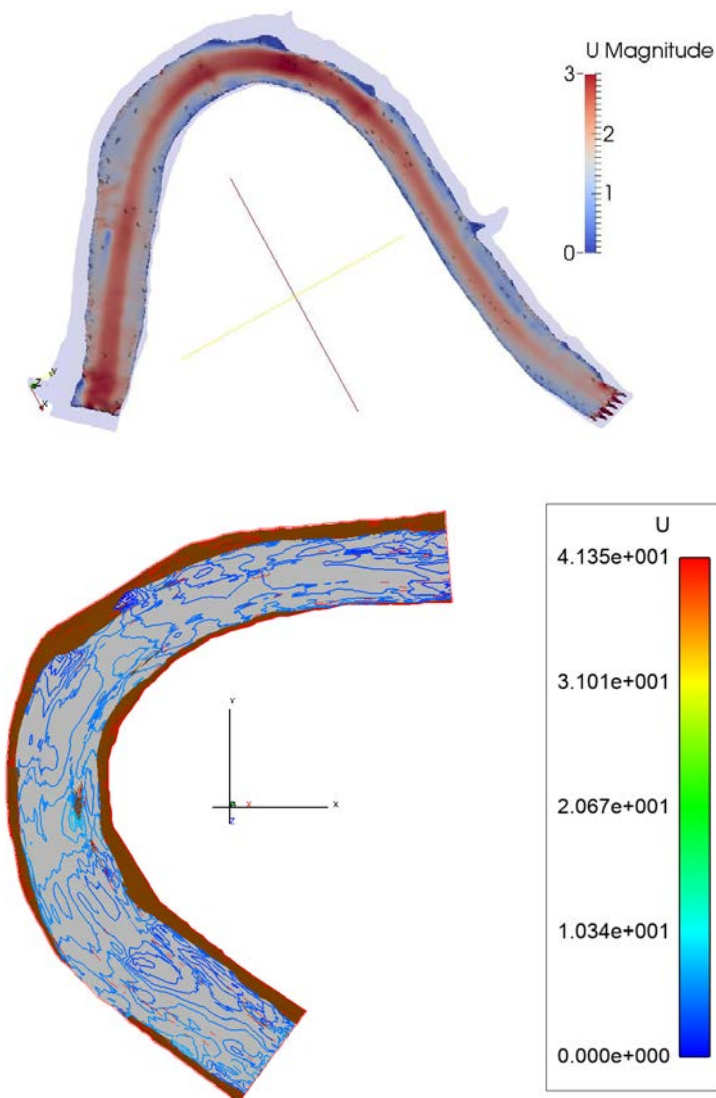


Ilustración 8.6.- Velocidad de las partículas de la superficie libre del agua para un caudal de $1.400 \text{ m}^3/\text{s}$. Superior: Resultados de Aranda (2016). Inferior: Resultados en el presente TFM.

De la Ilustración 8.6 se desprenden varias conclusiones. Por un lado, se observa que la interfase agua-aire no es la misma en ambos casos. Esto es debido, fundamentalmente, a que se han empleado diferentes MDTs, siendo el segundo más preciso que el primero, permitiéndolo capturar con más detalle la orografía del terreno, como puede ser los pequeños montículos alrededor de los cuales se divide el flujo.

Por otro lado, se observa que la distribución de velocidades no es la misma, siendo más irregular en el segundo caso. Esto puede ser debido a diversos motivos como son:

- Definición de la malla
- Modelos numéricos empleados en la simulación
- Modelo Digital del Terreno

En cuanto a la definición de la malla, es importante que ésta sea fina en la zona de la interfase agua-aire, puesto que de otro modo no capturarán correctamente el fenómeno que se está estudiando.

Los métodos numéricos empleados en la simulación condicionan el resultado de la misma de un modo notable. Cuanto mayor sea la difusión numérica, más ‘ancha’ será la interfase agua-aire y los resultados serán menos precisos. Además, la turbulencia en el cauce es muy importante, y el modelo matemático que se emplee para modelarla afecta directamente al resultado.

Por último, el MDT es determinante en la simulación. En el caso que nos ocupa, el MDT obtenido a partir de datos LiDAR es mucho más preciso que el que se obtiene por correlación fotogramétrica (Ilustración 2.4) y, por tanto, es menos suave, debido a la naturaleza del terreno. Esto implica que la malla es mucho más irregular, con los problemas que ocasiona desde el punto de vista de la simulación este hecho, ya que los métodos numéricos empleados en CFD asumen una regularidad en la malla.

9 Conclusiones

El presente estudio, cuyo objetivo es la generación de una malla computacional óptima y la posterior simulación numérica (modelo tipo Volumen Fluido para flujo agua-aire en régimen turbulento y tridimensional) en el cálculo de avenidas de ríos y delimitación de áreas inundables se ha cumplido satisfactoriamente.

Se ha comenzado por analizar el problema que presenta el proceso de generación de la malla, a la vez que se hacía hincapié en el interés de la zona de estudio debido a dos factores fundamentalmente: el daño ocasionado por las inundaciones que tienen lugar periódicamente y el reto que supone, desde el punto de vista ingenieril, la generación de una malla estructurada en una geometría altamente irregular, como es un meandro y estrechamientos del cauce tipo “cañón”.

Posteriormente se ha estudiado la información del terreno de que se disponía: datos LiDAR y cartografía clásica. Se ha analizado la precisión de cada una de las técnicas, y se ha justificado la utilización de los datos LiDAR, debido a las ventajas que presentan frente a los MDT generados a partir de correlación fotogramétrica de imágenes. Además, los datos que proporciona y recomienda utilizar el Ministerio son los procedentes de los vuelos LiDAR.

A continuación, se ha descrito detalladamente el proceso de generación de la malla. Desde la descarga de los datos que proporciona el Ministerio de Fomento, la importación de dichos datos en el software *Global mapper*, la eliminación de un puente, según establece la guía metodológica para el desarrollo del SNCZI hasta la posterior exportación de dichos datos a un fichero .stl para ser leído con Pointwise.

Con este último programa, se ha generado una malla estructurada manual, intentando sacarle el máximo partido a las ventajas que presenta una malla estructurada frente a una no-estructurada, teniendo en cuenta que se podía estimar la solución, en base a estudios previos contrastados con evidencias de inundaciones previas, permitiéndonos acotar la geometría del problema al dominio fluido de interés.

El siguiente paso ha sido describir, de un modo exhaustivo, los indicadores de calidad de malla que se emplean en la mayoría de los paquetes de software CFD. En

concreto se han descrito los que muestra la guía de usuario de ANSYS Fluent. A partir de este punto, se han analizado tanto la malla que realizó Aranda como la generada en el presente TFM. A la vista de los resultados de la malla aquí generada, se procedió a generar otra malla, esta vez buscando maximizar la regularidad, disminuyendo la *oblicuidad (skewness)* de las celdillas.

Posteriormente, se compararon las dos mallas generadas en el presente TFM empleando el propio software de generación de mallas aquí utilizado: Pointwise.

Una vez generada y analizada la malla, se procedió a describir el funcionamiento de un código CFD, compuesto fundamentalmente por el pre-procesador, un procesador, y un post-procesador. El software empleado para cada etapa ha sido fundamentalmente PointWise, OpenFOAM y EnSight respectivamente. Todos ellos son paquetes de software dedicados expresamente a la tarea correspondiente, lo que les confiere capacidades con las que un software más generico no puede competir.

Después de describir el funcionamiento de un código CFD, se ha explicado, paso a paso, el proceso seguido en la configuración de la simulación en OpenFOAM, para, posteriormente ejecutar dos simulaciones a distintos caudales: 500 y 1.400 m³/s.

Por último, se han expuesto y comparado los resultados obtenidos con simulaciones previas, y se han observado y comentado las analogías y diferencias.

Como resultado de todo lo anterior, se puede concluir lo siguiente:

1. Para que una simulación por FVM sea probable que funcione correctamente, es crucial que la malla computacional que modela el volumen fluido objeto de estudio esté bien realizada.
2. Las mallas estructuradas son mucho más eficientes que las no estructuradas por los siguientes motivos:
 - a. Permiten un control del proceso de creación de malla muy superior, especialmente en el interior del dominio computacional
 - b. Debido a la forma en la que están almacenadas en la memoria del ordenador, la conectividad entre celdillas está definida implícitamente, por lo que utiliza la potencia de cálculo del

ordenador de un modo mucho más eficiente reduciendo, por tanto, el coste computacional de la simulación

- c. Permiten capturar el fenómeno de capa límite en las proximidades del contorno muchísimo mejor que las mallas no estructuradas. De hecho, en muchas simulaciones que no están bien formuladas, las mallas no estructuradas no capturan la capa límite.
 - d. Para un mismo volumen fluido, una malla estructurada requiere un número muy inferior de celdillas, debido a que los hexaedros son menos compactos que los tetraedros, siendo estos últimos la geometría de celdilla preferente en mallas no estructuradas. Esta característica es de enorme importancia, puesto que el coste computacional de una simulación crece potencialmente con el número de celdillas.
 - e. Permiten generar una malla más vasta en las zonas donde el gradiente de la magnitud fluida objeto de interés (presión, velocidad, etc...) es menor.
 - f. Si se conoce la dirección principal del flujo, y se genera la malla de modo que los vectores normales de las caras de las celdillas estén alineados con los vectores velocidad del flujo, los métodos numéricos convergen mucho más rápidamente, disminuyendo la difusión numérica y aumentando la precisión de la simulación
 - g. A las mallas estructuradas, por norma general, se les pueden aplicar todos los métodos numéricos que son aplicables a mallas no estructuradas, mientras que no ocurre lo mismo en caso contrario: muchos métodos numéricos son sólo aplicables a mallas estructuradas, no siendo válidos para mallas no estructuradas.
 - h. Las mallas estructuradas soportan métodos numéricos de alto orden (aproximaciones parabólicas, cúbicas, etc...) con las ventajas que ello conlleva.
3. Los modelos digitales del terreno generados a partir de datos LiDAR son, por norma general, mucho más precisos que los obtenidos mediante correlación digital de imágenes. Además, permiten clasificar en suelo según:
 - a. Agua

- b. Superficie sólida
 - c. Vegetación
4. Las mallas estructuradas están fuertemente condicionadas por la geometría objeto de estudio. Ésta debe ser aproximadamente prismática para poder ser generada. Geometrías altamente irregulares no son modelables mediante mallas estructuradas o, al menos, mediante una única malla estructurada. Generalmente en estos casos es necesario descomponerla en múltiples mallas estructuradas y posteriormente definir la conectividad entre ellas. Lo habitual hoy día son las mallas híbridas, donde parte de la malla es estructurada (grandes gradientes de magnitudes fluidas) mientras que el resto del volumen fluido se rellena con una malla más vasta no estructurada.
 5. Los indicadores de calidad de malla son útiles, pero no son la única herramienta a la hora de comprar mallas. Su potencia radica en determinar zonas irregulares en la malla, así como comparar mallas equivalentes. Esto es, por ejemplo, la comparativa que se ha realizado entre la primera malla estructurada generada y la segunda. No pueden ser utilizadas como único indicador para comparar mallas muy distintas, puesto que la malla está fuertemente condicionada por la naturaleza del problema.
 6. Los malladores automáticos permiten una enorme versatilidad y una velocidad en el proceso de mallado que dista mucho de los procesos de generación manual de mallas. Lo que ocurre es que los malladores automáticos no conocen el fenómeno que se pretende modelar, por lo que sus criterios para mallar son totalmente diferentes. Por ejemplo, la triangulación de Delaunay, muy empleada por malladores automáticos, descompone el dominio de modo que los tetraedros que la forman sean lo más regulares posible, pero no tiene en cuenta zonas en las que el gradiente de magnitud fluida sea importante ni fenómenos como la capa límite, puesto que los desconoce. En el caso de elementos finitos, ocurre algo parecido: no tienen en cuenta tampoco fenómenos como los concentradores de tensiones. Por tanto, si se quiere realizar una simulación (tanto por volúmenes finitos como por elementos finitos) de calidad, son necesarios dos requisitos:

- a. Tener una idea de la solución del problema, para lo cual se puede realizar una simulación más vasta con una malla automática, y
 - b. Generar una malla adecuada que sea capaz de capturar los fenómenos de interés. Este proceso puede ser iterativo, modificar la malla y volver a simular hasta que converjan los resultados. En los problemas de mecánica de fluidos, por ejemplo, para determinar el parámetro y^+ suele ser habitual realizar varias iteraciones (malla – simulación) hasta que se captura lo que se busca.
7. Los parámetros de configuración de una simulación CFD son determinantes a la hora de que la simulación converja o no, así como de la precisión de los resultados. Es importante conocer el fenómeno objeto de estudio para poder aplicar los métodos numéricos (*solvers*) apropiados a cada caso.
8. Cuando se puede estimar, a priori, la solución de la simulación (por ejemplo, con estudios previos o evidencias de campo), es una buena opción inicializar las variables conocidas antes de ejecutar la simulación; pues se reduce drásticamente el coste computacional de la misma. Esto es lo que se ha realizado mediante *funkySetFields* de OpenFOAM para la variable fluida α , de modo que se le ha indicado al software cuál era la condición inicial de llenado del cauce, evitando tener que simular todo el proceso de llenado. El tiempo de convergencia de la simulación es muy superior (del orden de 600 s) frente al tiempo que se requeriría si se tuviese que llenar el cauce.

10 Anejo 1: Generación de mallas mediante SnappyHexMesh

SnappyHexMesh es un mallador automático, implementado dentro del software libre OpenFOAM. La estrategia general que sigue este mallador se puede resumir en: *“Mallado automático por intersección. Refinamiento y mapeo de la superficie”*

El proceso de creación de mallas de este programa está automatizado, siendo los pasos seguidos en el proceso de generación de malla los que se muestran, de un modo esquemático, en la Ilustración 10.1:

1. Importación de la superficie (3D) o curva (2D)
2. División del dominio computacional en celdas estructuradas
3. Las celdas que pertenecen parcialmente al dominio computacional, se dividen
4. Se procede del mismo modo que en 2. para toda la curva (2D) o superficie (3D) de intersección
5. Se eliminan las celdillas que no pertenecen al dominio computacional.
6. Se repiten los pasos 2-4 tantas veces como se considere, teniendo en cuenta que cada vez que se itera, aumenta el número de celdillas.
7. Hasta el punto anterior, toda la malla estaba formada por celdas rectangulares (2D) o hexaédricas (3D). En este paso, se desplazan los nodos de las celdillas a la superficie y se eliminan los que quedan fuera del dominio computacional. De ahora en adelante, la malla está formada por más tipos de celdillas, a saber: triángulos (2D) o poliedros (3D).

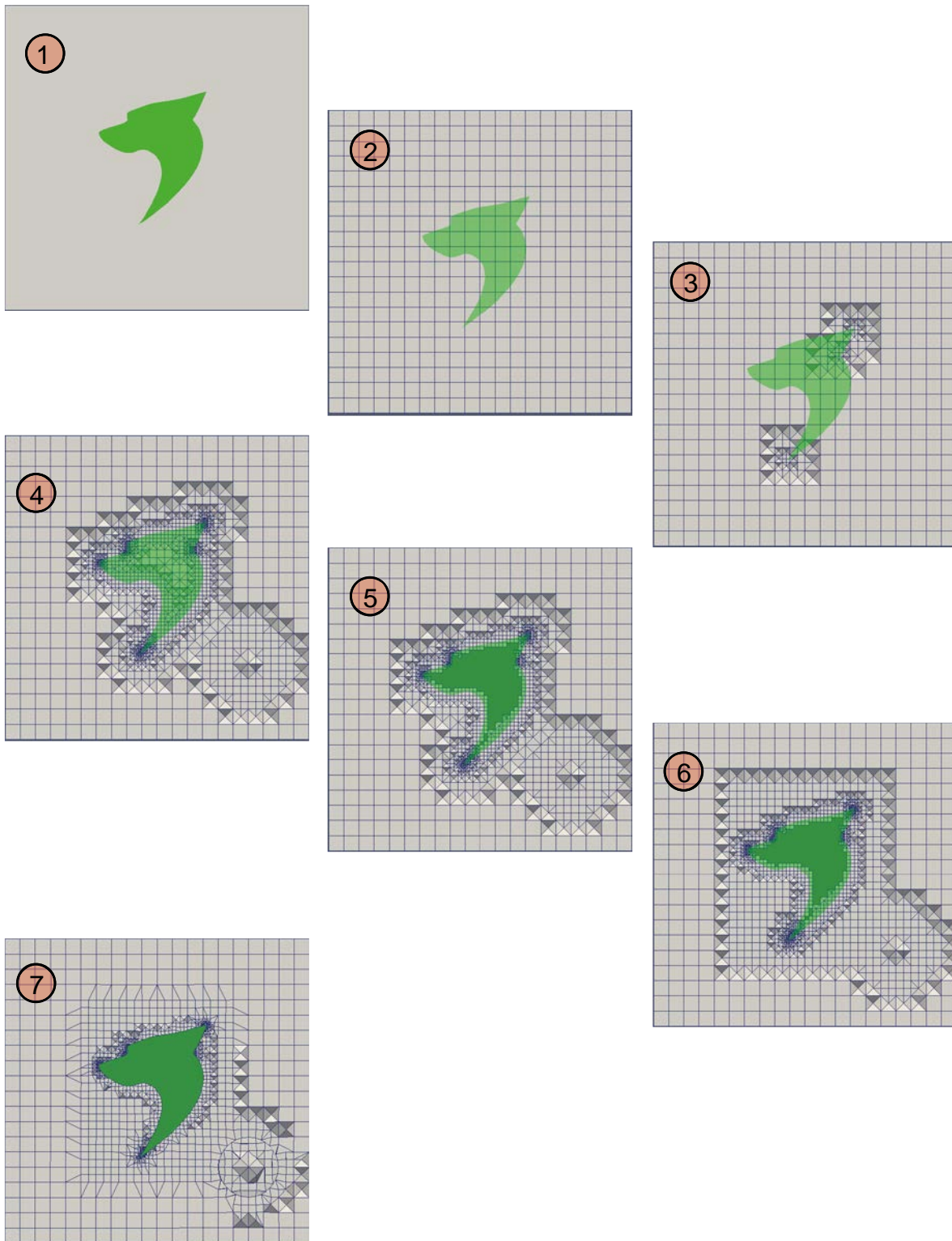


Ilustración 10.1.- Esquema simplificado del proceso de generación de malla con SnappyHexMesh

11 Referencias

- Aranda Ochando, A. (2016). *TFG: Simulación tridimensional multifásica del río Guadalquivir durante crecidas extremas en tramos tipo cañón*. Jaén.
- Bohórquez Rodríguez de Medina, P. (2010). *Numerical Simulation of Sediment Transport in Free-Surface Flow*. LAP Lambert Academic Publishing.
- Bohórquez Rodríguez de Medina, P. (2016). Paleohydraulic Reconstruction of Modern Large Floods at Subcritical Speed in a Confined Valley: Proof of Concept. *Water*, 1-18.
- Bohórquez Rodríguez de Medina, P., & del Moral Erenca, J. D. (2017). 100 Years of Competition between Reduction in Channel Capacity and Streamflow during Floods in the Guadalquivir River (Southern Spain). *Remote Sensing*, 1-23.
- Calvente González, R. (2016). *TFG: Simulación numérica no hidrostática del flujo sobre el aliviadero de la presa de Marmolejo*. Jaén.
- Crespo, A. (2006). *Mecánica de Fluidos*. Thomson Paraninfo.
- Eymard, R., Gallouët, T., & Herbin, R. (2003). *Finite Volume Methods*. Marseille.
- Greenshields, C. J. (2016). *OpenFOAM User Guide version 4.0*.
- Holzinger, G. (2015). *OpenFOAM A little User-Manual*. Linz: Johannes Kepler University.
- Junta de Andalucía. (05 de julio de 2017). *Consejería de agricultura, pesca y desarrollo rural*. Obtenido de http://www.juntadeandalucia.es/agriculturaypesca/ifapa/ria/servlet/FrontController?action=Static&url=fechas.jsp&c_provincia=23&c_estacion=16
- Junta de Andalucía, Consejería de medio ambiente y Ordenación del territorio. (02 de Julio de 2017). *juntadeandalucia.es/medioambiente*. Obtenido de http://www.juntadeandalucia.es/medioambiente/site/rediam/menuitem.04dc44281e5d53cf8ca78ca731525ea0/?vgnnextoid=e87cd8cb9b1ec110VgnVCM1000001325e50aR CRD&vgnnextchannel=48f87d087270f210VgnVCM1000001325e50aR CRD&vgnnextfmt=rediam&lr=lang_es
- Komzsik, L. (2009). *What every engineer should know about computational techniques of finite element analysis*. Boca Raton: CRC Press.
- La contra de Jaén. (Octubre de 2017). *La contra de Jaén*. Obtenido de <http://lacontradejaen.com/nuevo-clamor-las-inundaciones/>
- La contra de Jaén. (Septiembre de 2017). *La contra de Jaén*. Obtenido de <http://lacontradejaen.com/freno-al-problema-las-inundaciones/>
- Lawrence Dingman, S. (2009). *Fluvial Hydraulics*. Oxford University Press.
- Lázaro López, A. (1997). *Manual de hidráulica*. Alicante: Universidad de Alicante. Servicio de publicaciones.

- Marcucci, N. (02 de Julio de 2017). *aviationmapping.blogspot.com*. Obtenido de <http://aviationmapping.blogspot.com.es/2009/12/what-is-orthophotography-remote-sensing.html>
- Maric, T., Höpken, J., & Mooney, K. (2014). *The OpenFOAM technology Primer*. sourceflux.
- Mercado Pérez, J. A. (2015). *TFG: Simulación numérica bidimensional del río Guadalquivir a su paso por la presa de Marmolejo*. Jaén.
- Ministerio de medio ambiente y medio rural y marino. (2011). *Guía metodológica para el desarrollo del sistema nacional de cartografía de zonas inundables*. Madrid: Ministerio de medio ambiente y medio rural y marino.
- Plaza Jiménez, J. A. (2015). *TFG: Dinámica fluvial del río Guadalbullón a su paso por las Infantas*. Jaén.
- PointWise. (s.f.). *Pointwise Tutorial Workbook*.
- Rodríguez Rico, Ó. O. (2009). *LiDAR vs fotogrametría en la obtención de MDE para estudios hidrológicos*. Salamanca: Universidad de Salamanca.
- S. H. Lo, D. (2015). *Finite Element Mesh Generation*. Boca Raton: CRC Press.
- Saeta Palacios, A. (2016). *TFG: Simulación numérica unidimensional de la dinámica fluvial del río Guadalquivir a su paso por la presa de Marmolejo*. Jaén.
- Thompson, J., Warsi, Z., & Wayne Mastin, C. (s.f.). *Numerical grid generation. foundations and applications*.
- Versteeg, H., & Malalasekera, W. (1995). *An introduction to Computational Fluid Dynamics*. New York: Longman Group.
- White, F. M. (2004). *Mecánica de Fluidos*. McGraw-Hill.