

OPTIMIZACIÓN DEL SISTEMA DE REFRIGERACIÓN DE UN ÁLABE UTILIZANDO TÉCNICAS DE MALLAS SOLAPADAS Y COMPUTACIÓN EN PARALELO

OPTIMIZATION OF A BLADE COOLING PASSAGES SYSTEM USING AN OVERSET GRIDS TECHNIQUE AND PARALLEL COMPUTING

Bruno A. Storti, Luciano Garelli, Mario A. Storti y Jorge D'Elía

^a*Centro de Investigación de Métodos Computacionales, (CIMEC), (CONICET-UNL)
Colectora Ruta 168 s/n, Predio Conicet "Dr. Alberto Cassano", 3000 Santa Fe, Argentina,
bstorti@cimec.unl.edu.ar, <http://www.cimec.org.ar>*

Palabras clave: método Chimera, mallas solapadas, optimización en paralelo, álabe, pyOpt.

Resumen. El aumento del rendimiento de una turbina de gas depende casi estrictamente de la temperatura máxima de los gases de combustión. Dicha temperatura se encuentra limitada por la resistencia termo-mecánica de los álabes de la turbina. En este trabajo se propone un esquema de mallas solapadas tipo Chimera aplicado a la optimización de la conducción térmica de un álabe con el objetivo de minimizar su temperatura media. El método consiste en una malla de fondo fija, cubriendo todo el dominio, y mallas más finas rodeando pasajes refrigerantes que se mueven libremente sobre el perfil buscando la mejor configuración del sistema. El intercambio de información entre mallas se realiza mediante un algoritmo de interpolación de tercer orden. Se resuelve un ejemplo numérico a modo de evaluar la convergencia espacial del método. La estrategia de optimización implementada consiste en utilizar un algoritmo basado en población para acercarse a un probable mínimo global, y luego refinar la solución utilizando un algoritmo basado en gradiente. Tanto el cómputo de gradientes como las evaluaciones de la función objetivo de los optimizadores son realizados en paralelo. Se evalúan cuestiones como speedup y convergencia de los solvers.

Keywords: Chimera method, overlapping grids, parallel optimization, turbine vane, pyOpt.

Abstract. Improvement of gas turbines performance depends almost strictly on maximum temperature of the combustion gases. This temperature is limited by thermo-mechanical strength of turbine vanes. In this work a Chimera approach for overlapping grids is proposed in order to solve a heat conduction optimization problem for minimizing the average temperature of a vane. The scheme is based on a fixed background mesh covering the entire domain, while finer meshes surrounding coolant passages can move around over the airfoil looking for the best configuration. Exchange of information between meshes is made by a third order interpolation algorithm. A numerical example is solved assessing the method spatial convergence. The optimization strategy proposed is built on the use of a population based algorithm to get close to a probable global minimum and then a refinement of the solution is made using a gradient-based optimizer. Gradients for the gradient-based optimizer are evaluated in parallel, and function evaluations for the gradient-free optimizers are distributed on each core. Speedup and convergence matters of optimizers are evaluated and discussed.

1. INTRODUCCIÓN

La eficiencia térmica del ciclo de una turbina de gas depende de distintos factores. Poder incrementar la temperatura de los gases de combustión en el ingreso de la turbina es uno de los principales objetivos de ingenieros y estudios de investigación. De todos los componentes de una turbina, el álabe es el componente más solicitado debido a las elevadas cargas térmicas que en él se presentan. Para poder garantizar la vida útil de la pieza y evitar su falla es necesario un apropiado diseño/selección del material y una óptima configuración del sistema de refrigeración. Generalmente dicho sistema de refrigeración se logra haciendo fluir por dentro de los álabes aire a relativamente bajas temperaturas (alrededor de los 300 [K]) que fue previamente extraído del compresor. Una configuración óptima del sistema de refrigeración implicaría reducir las cargas térmicas del álabe (es decir la disminución de la temperatura media del perfil y los gradientes de temperatura) con la mínima cantidad posible de refrigerante. Por esto es que la geometría de los pasajes refrigerantes y su disposición es de gran importancia. Distintos métodos de optimización para dicha distribución y tamaños pueden encontrarse en la bibliografía. [Wang et al. \(2015\)](#) por ejemplo proponen la optimización de forma, ubicación y dimensiones de los tubos refrigerantes en busca de minimizar la temperatura media del álabe sin comprometer su resistencia termo-mecánica. Para lo cual estudian tres casos diferentes, el primero con geometrías circulares de los pasajes, el segundo con configuraciones de super-elipses propuestas por los autores, y el tercero con geometrías circulares de pequeño diámetro cercanas a la superficies, basándose en el trabajo de [Dennis et al. \(2003\)](#). El trabajo se basó en el uso del método de elementos finitos (FEM) para el análisis termo-mecánico sobre una geometría de dos dimensiones ubicada en la sección media del perfil, y para el proceso de optimización se utilizó el método de asíntotas móviles. Mientras que [Dennis et al. \(2003\)](#) estudió dos posibles configuraciones, una contemplando un máximo de 13 pequeños tubos cerca de la superficie del álabe, y otro con un máximo de 30 tubos. Se realiza un análisis termo-elástico usando FEM sobre una geometría de tres dimensiones y para la optimización se utilizan algoritmos genéticos. Se utiliza una condición de borde de flujo térmico convectivo para modelar el flujo térmico transferido al perfil. Por otro lado, [Nowak y Wróblewski \(2011\)](#) utilizaron un modelo reducido de transferencia térmica conjugada (RCHT) en el cuál se supone el flujo externo de los gases del álabe como fijo y solo se resuelve la conducción térmica dentro del álabe.

Todos los métodos antes mencionados implican un remallado del dominio y un apropiado re-ensamblado de la matriz de rigidez para cada evaluación de la función objetivo.

Teniendo en cuenta que los algoritmos de optimización basados en población, aunque tienen mayor probabilidad de encontrar el mínimo global respecto a métodos basados en gradiente por ejemplo, requieren de una gran cantidad de evaluaciones de la función objetivo (lo que implica un elevado costo computacional), proponemos en este trabajo (a fin de evitar el proceso de re-mallado y reducir dicho costo) un esquema de mallas solapadas tipo Chimera en el cual se generan mallas independientes para los objetos presentes en el dominio computacional y luego son acopladas mediante operadores de interpolación y de pegado para obtener una única solución del sistema. Este esquema de grillas solapadas es utilizado para evitar el exhaustivo trabajo de mallado de piezas complejas como ser el ala de un aeroplano o el aspa de una turbina eólica ([Andrew M. et al., 2009](#)). Otras posibles aplicaciones del método son: refinamiento local ([Kao et al., 1994](#); [Meakin, 1995](#)), componentes móviles ([Prewitt et al., 2000](#); [Blades y Marcum, 2007](#)) y optimización. El proceso de optimización es una aplicación directa donde distintos objetos, cada uno con su correspondiente malla, puede moverse libremente sobre una malla de fondo que cubre todo el dominio computacional, buscando la configuración óptima respecto a

una función objetivo.

Este trabajo está organizado de la siguiente manera: en la primera sección se realiza una descripción del sistema algebraico generado por el esquema Chimera, y se introducen los operadores de interpolación utilizados para transferir información entre las mallas. Se propone un algoritmo para acoplar las soluciones de ambos dominios resolviendo el sistema con un solver iterativo. También, con el fin de mantener el orden de convergencia de la solución, se propone un interpolador de tercer orden, el cual se implementa y se valida en un caso numérico simple. En la segunda sección, se introduce la plataforma de optimización *pyOpt* y se describen brevemente dos optimizadores utilizados en este trabajo. Un problema simple de optimización de conducción térmica en un dominio en dos dimensiones es resuelto aplicando el esquema Chimera evaluando convergencia y speedup de la implementación en paralelo. En la tercera sección, basado en los experimentos de [Hylton et al. \(1983\)](#), el sistema de refrigeración del álabe de una turbina es optimizado. Un modelo de dos dimensiones del plano medio del perfil es analizado. Tanto las cargas térmicas sobre el álabe como las condiciones físicas del refrigerante son obtenidos de datos experimentales.

2. METODOLOGÍA

Teniendo en cuenta que en la estrategia de optimización, cada evaluación de la función objetivo implicaría nuevas posiciones de los pasajes refrigerantes, lo cual obligaría al remallado de todo el dominio computacional si se utilizara un esquema tradicional de discretización por FEM. En el presente trabajo, con el fin de evitar dicho proceso de remallado, se propone un nuevo enfoque utilizando Métodos de Composición de Dominio (DCM) para grillas solapadas. El principal objetivo de los DCM es expresar la formulación de un problema en varios subdominios, para luego acoplarlos y obtener una única solución al sistema de ecuaciones. Los DCM pueden ser caracterizados en tres grupos según la formulación que utilicen: basados en malla, en ecuación y en el sistema algebraico. En las formulaciones basadas en malla, el acople entre los subdominios es obtenido como un pre-proceso en el nivel de mallado. En las formulaciones basadas en ecuación, la solución del problema es obtenida mediante el reordenamiento de las ecuaciones desde la formulación débil o fuerte del problema variacional. Mientras que para formulaciones basadas en el sistema algebraico, la solución se obtiene directamente del sistema de ecuaciones algebraicas de cada subdominio. Una explicación mas detallada de cada una de estas formulaciones fue realizada por [Houzeaux et al. \(2017\)](#). Un método Chimera con una formulación basada en el sistema algebraico para resolver la ecuación de Poisson en dos subdominios se presentará en la siguiente sección.

2.1. Descripción del método de Chimera

Con el fin de simplificar la explicación del método, vamos a considerar solo dos subdominios (Ω_1 y Ω_2) con una pequeña región solapada ($\Omega_1 \cup \Omega_2$). Entonces consideremos el siguiente dominio Ω donde queremos resolver la ecuación de Poisson dada por:

$$\begin{aligned} \Delta\phi &= -f, \text{ in } \Omega, \\ \phi &= \bar{\phi}, \text{ in } \Gamma = \partial\Omega \end{aligned} \quad (1)$$

Vamos a considerar la partición del dominio en dos subdominios $\Omega_{1,2}$ (ver figura 1). Los dominios contienen una región que se encuentra superpuesta Ω_o . Las dos mallas no son coincidentes. Ahora vamos a considerar la solución del problema en el dominio Ω_1 . Una discretización están-

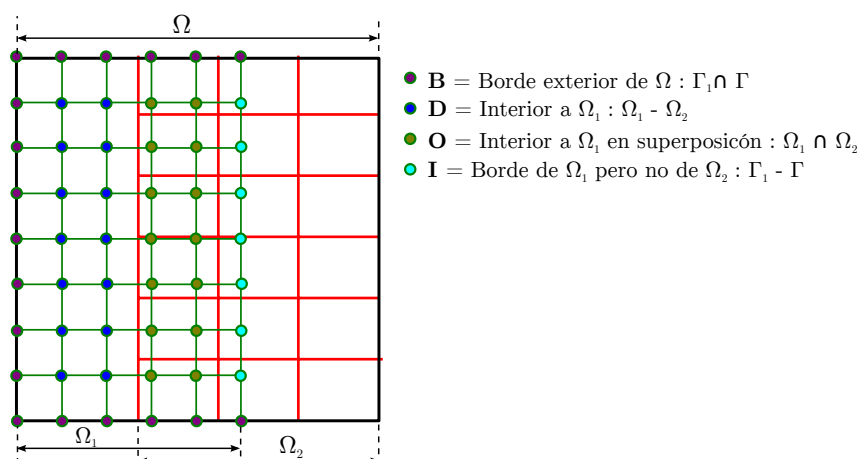


Figura 1: Descripción del método Chimera.

dar con el Método de Elementos Finitos nos lleva a la siguiente formulación matricial:

$$\mathbf{Ax} = \mathbf{b} \quad (2)$$

Los nodos de la malla pueden ser divididos en cuatro grupos: el grupo **B** que contiene los nodos del borde del dominio Ω_1 y que también pertenecen a la frontera de Ω , el grupo **D** compuesto por los nodos interiores a Ω_1 que no se encuentran en la región solapada, el grupo **O** también compuesto por los nodos interiores a Ω_1 pero solo aquellos que se encuentran sobre la región solapada y el grupo **I** el cual contiene los nodos en el borde de Ω_1 pero que no están contenidos en la frontera de Ω . Estos últimos nodos deben estar contenidos en el subdominio Ω_2 . A partir de esta clasificación podemos agrupar los nodos internos de los grupos **O** y **D** en un solo grupo que llamaremos **Z** el cual contiene todos los nodos interiores.

Luego, al dividir los nodos en grupos, el sistema lineal queda planteado de la siguiente forma:

$$\begin{bmatrix} \mathbf{A}_{BB} & \mathbf{A}_{BZ} & \mathbf{A}_{BI} \\ \mathbf{A}_{ZB} & \mathbf{A}_{ZZ} & \mathbf{A}_{ZI} \\ \mathbf{A}_{IB} & \mathbf{A}_{IZ} & \mathbf{A}_{II} \end{bmatrix} \begin{bmatrix} \mathbf{x}_B \\ \mathbf{x}_Z \\ \mathbf{x}_I \end{bmatrix} = \begin{bmatrix} \mathbf{b}_B \\ \mathbf{b}_Z \\ \mathbf{b}_I \end{bmatrix} \quad (3)$$

Del sistema planteado, los valores de \mathbf{x}_B son conocidos debido a las condiciones de borde establecidas. Si conociéramos los valores en los nodos de la interfaz \mathbf{x}_I , tendríamos un problema definido con condiciones tipo Dirichlet en toda la frontera, pudiendo encontrar la solución al sistema resolviendo:

$$\mathbf{A}_{ZZ}\mathbf{x}_Z = \mathbf{b}_Z - \mathbf{A}_{ZB}\mathbf{x}_B + \mathbf{A}_{ZI}\mathbf{x}_I \quad (4)$$

Como los valores en los nodos de la interfaz \mathbf{x}_I no son conocidos, deberemos interpolarlos de los valores en los nodos interiores al dominio Ω_2 . De esta forma se obtiene un sistema lineal completo en el interior de ambos dominios. Desde este momento usaremos los supra-índices para referirnos a cada subdominio.

$$\begin{aligned} \mathbf{A}_{ZZ}^1 \mathbf{x}_Z^1 &= \mathbf{b}_Z^1 - \mathbf{A}_{ZB}^1 \mathbf{x}_B + \mathbf{A}_{ZI}^1 \Pi_{I_{1,2}} \mathbf{x}^2, \\ \mathbf{A}_{ZZ}^2 \mathbf{x}_Z^2 &= \mathbf{b}_Z^2 - \mathbf{A}_{ZB}^2 \mathbf{x}_B + \mathbf{A}_{ZI}^2 \Pi_{I_{2,1}} \mathbf{x}^1, \end{aligned} \quad (5)$$

De la Ecuación 5, el operador $\Pi_{I_{1,2}}$ es un operador de proyección que interpola valores de un vector \mathbf{x}^2 a los nodos de la interfaz I^1 . Se debe tener en cuenta que para poder realizar dicha interpolación debe haber al menos una capa de elementos en la región solapada. De esta forma

la interpolación involucra solamente los nodos de los grupos B^2 y Z^2 , y no los del grupo I^2 . De esta forma el sistema lineal de la Ecuación 5 puede ser resuelto para los nodos interiores \mathbf{x}_Z^1 and \mathbf{x}_Z^2 .

2.2. Algoritmo

El esquema puede plantearse como un operador lineal que puede ser utilizado por un solver iterativo del tipo Matrix-free como ser el Generalized Minimal RESidual (GMRES) o el BiConjugate Gradient Stabilized method (BiCGStab). El sistema que queda planteado no es ni simétrico ni definido positivo, a pesar de que el problema original lo es. Sin embargo experiencias numéricas muestran que puede ser resuelto con BiCGStab con tasas de convergencia similares a las que se pueden obtener para un problema de malla conforme (sin solapamiento) resuelto con Gradiente Conjugado (CG). La Ecuación 5 puede ser re-escrita en forma de residuos como se muestra a continuación:

$$\begin{aligned} \mathbf{R}_Z^1(\mathbf{x}_Z^1, \mathbf{x}_Z^2) &= \mathbf{A}_{ZZ}^1 \mathbf{x}_Z^1 - \mathbf{b}_Z^1 + \mathbf{A}_{ZB}^1 \mathbf{x}_B - \mathbf{A}_{ZI}^1 \Pi_{I1,2} \mathbf{x}^2, \\ \mathbf{R}_Z^2(\mathbf{x}_Z^1, \mathbf{x}_Z^2) &= \mathbf{A}_{ZZ}^2 \mathbf{x}_Z^2 - \mathbf{b}_Z^2 + \mathbf{A}_{ZB}^2 \mathbf{x}_B - \mathbf{A}_{ZI}^2 \Pi_{I2,1} \mathbf{x}^1. \end{aligned} \quad (6)$$

El algoritmo para calcular los residuos con los valores interiores de ambos subdominios se muestra a continuación.

Algoritmo 1: Algoritmo Chimera

- 1: Calcular $\mathbf{r} = f(\mathbf{x})$
 - 2: Extraer $\mathbf{x}_Z^1, \mathbf{x}_Z^2$ de $\mathbf{x} = [\mathbf{x}_Z^1; \mathbf{x}_Z^2]$
 - 3: Interpolar \mathbf{x}_Z^2 en los nodos I^1
 - 4: Calcular el residuo \mathbf{r}_Z^1 en Z^1 de acuerdo a la línea 1.
 - 5: Interpolar \mathbf{x}_Z^1 en los nodos I^2
 - 6: Calcular el residuo \mathbf{r}_Z^2 en Z^2 de acuerdo a la línea 2
 - 7: Combinar $\mathbf{r} = [\mathbf{r}_Z^1; \mathbf{r}_Z^2]$
-

2.3. Esquema de interpolación

El orden de convergencia espacial de la solución esta directamente afectado por el esquema de interpolación utilizado según se pudo verificar en diversos experimentos numéricos. Con el fin de preservar el segundo orden de convergencia que posee el Método de Elementos Finitos en mallas no estructuradas, es necesaria una interpolación de tercer orden. A continuación se presenta una descripción de un esquema de interpolación utilizando operadores de Lagrange implementado y utilizado en este trabajo.

2.3.1. Formulación

Para su descripción se asume que existe una malla Ω_1 compuesta de elementos triangulares no estructurados (ver Figura 2) y se conocen los valores de una función suave ϕ en los vértices de los nodos (círculos blancos en la Figura 2). Ahora suponemos que existen ciertos puntos de prueba \mathbf{x}_j pertenecientes al dominio Ω_2 en los cuales se requiere conocer su valor a partir de una interpolación de tercer orden de los valores de ϕ . Esto requiere que el error de la interpolación sea $O(h^3)$, siendo h el tamaño de elemento característico de la malla Ω_1 .

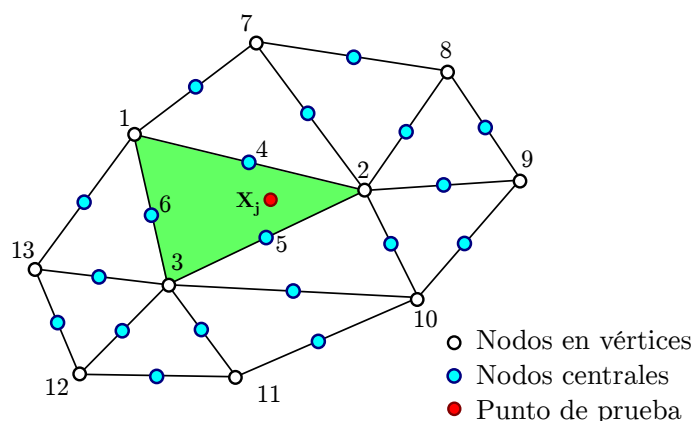


Figura 2: Esquema de interpolación para mallas no estructuradas

La interpolación se basa en reconstruir los valores de la función ϕ en los nodos centrales (en color cian en la Figura 2) y luego hacer una interpolación Lagrangiana P_2 . Para realizar dicha reconstrucción de valores en los nodos centrales, se aplica un ajuste por Mínimos Cuadrados a los nodos de los vértices junto a sus vecinos topológicos. Volviendo al punto de prueba x_j de la figura 2, para realizar una interpolación sobre dicho nodo (que se encuentra dentro del elemento coloreado en verde), se requieren los valores en los nodos centrales 4, 5 y 6 como también los valores de los nodos en los vértices 1, 2 y 3. Para reconstruir el valor de ϕ en el nodo central 5 por ejemplo, se realiza un ajuste por Mínimos Cuadrados utilizando los valores en los vecinos topológicos de los nodos 2 y 3, siendo éstos los nodos del 1 al 13 (descontando los nodos centrales). Sea $\tilde{\phi}$ la reconstrucción de segundo orden:

$$\tilde{\phi}(x, y) = a + bx + cy + dx^2 + e xy + f y^2. \quad (7)$$

Cuando es aplicada a los puntos $M = \{1, 2, \dots, 13\}$ la reconstrucción nos da:

$$\tilde{\phi} = \Psi \mathbf{a}. \quad (8)$$

donde

$$\tilde{\phi} = \begin{bmatrix} \tilde{\phi}_1 \\ \vdots \\ \tilde{\phi}_m \end{bmatrix}, \in \mathbb{R}^m \quad \mathbf{a} = \begin{bmatrix} a \\ b \\ \vdots \end{bmatrix}, \in \mathbb{R}^{n_a} \quad (9)$$

$$\Psi = \begin{bmatrix} 1 & x_1 & y_1 & x_1^2 & x_1 y_1 & y_1^2 \\ \vdots & & & & & \\ 1 & x_m & y_m & x_m^2 & x_m y_m & y_m^2 \end{bmatrix}, \in \mathbb{R}^{m \times n_a}$$

siendo $m = |M| = 10$ el número de puntos (vértices de la arista y sus vecinos, en el ejemplo de los vértices 2 y 3) y $n_a = 6$ es el número de parámetros de la interpolación. La reconstrucción no puede imponerse sobre los 10 puntos ya que el sistema quedaría sobredeterminado. Por esto es que los parámetros son elegidos de forma que la reconstrucción en los nodos sea la que mejor se ajusta a la función ϕ en términos de mínimos cuadrados. Sin embargo, para la reconstrucción en los nodos que se encuentran en los extremos de la arista (nodos 2 y 3 en el ejemplo para la reconstrucción del nodo central 5) se impone el valor exacto. A partir de esto los vértices son separados en dos grupos disjuntos, los extremos a la arista $E = 2, 3$ y $N = M - E$. De esta

manera los parámetros de la reconstrucción son determinados de forma que el ajuste es exacto para los vértices E de la arista y óptimos por Mínimos Cuadrados para los nodos vecinos N .

$$\mathbf{a} = \operatorname{argmin} \sum_N |\tilde{\phi}_k - \phi_k|^2, \quad (10)$$

sujeto a las restricciones

$$\tilde{\phi}_k = \phi_k, \quad \text{for } k \in E. \quad (11)$$

Utilizando multiplicadores de Lagrange, el funcional Lagrangiano a optimizar quedaría planteado de la siguiente forma:

$$\mathcal{L}(\mathbf{a}, \boldsymbol{\lambda}) = \|\Psi_N \mathbf{a} - \phi_N\|^2 - \boldsymbol{\lambda}^T (\Psi_E \mathbf{a} - \phi_E) \quad (12)$$

luego las ecuaciones son:

$$\begin{bmatrix} (\Psi_N^T \Psi_N) & \Psi_E^T \\ \Psi_E & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{a} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \Psi_N^T \phi_N \\ \phi_E \end{bmatrix}. \quad (13)$$

De esta forma queda planteado un sistema cuadrado de $n_a + m_E$ que generalmente es no singular y bien condicionado excepto quizás para los nodos cercanos a los bordes. Si el sistema llegara a ser singular o mal condicionado, más capas de nodos pueden ser agregadas para realizar el ajuste. Una vez resuelto el sistema los parámetros \mathbf{a} son utilizados para la reconstrucción del valor del nodo central de la arista (nodo 5 en el ejemplo). Para un punto de prueba como el nodo x_j del ejemplo, se requiere reconstruir los nodos centrales 4, 5 y 6.

2.3.2. Ejemplo numérico. Error de interpolación

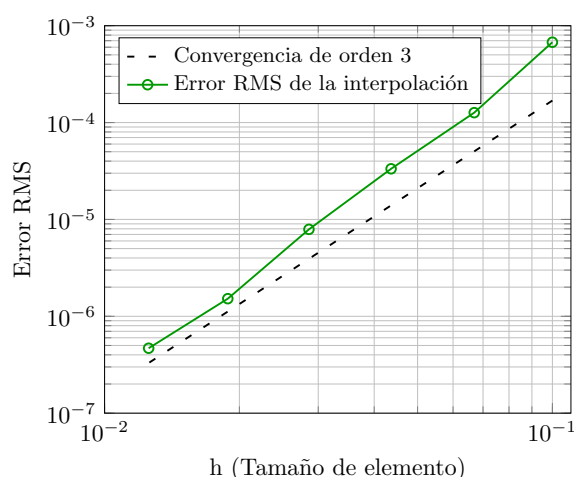


Figura 3: Error RMS de la interpolación propuesta.

Con el fin de evaluar el esquema de interpolación anteriormente propuesto, sobre un dominio cuadrado $\Omega = [0, 1] \times [0, 1]$, se crean mallas no estructuradas con número creciente de nodos compuestas por elementos triangulares. Luego se escoge aleatoriamente dentro del dominio un número dado de puntos de prueba ($N = 100$), se calcula una cierta función $\phi(x, y) =$

$\text{sen}(\pi x) \text{sen}(\pi y)$ en los nodos de la malla y se interpolan los valores en los puntos de prueba. El error cuadrático medio (RMS) es calculado como:

$$\epsilon_{\text{RMS}}^2 = \frac{1}{N} \sum_{j=1}^N |\tilde{\phi}_j - \phi_j|^2 \quad (14)$$

De la figura 3 se puede ver que la convergencia obtenida es $O(h^3)$. El interpolador se implementó utilizando el lenguaje C++, haciendo uso de las librerías Eigen (Guennebaud et al., 2010).

2.4. Convergencia

Un extensivo análisis de convergencia en malla en conjunto con el estudio de convergencia del solver iterativo ya fue realizado por los autores de este trabajo (Storti et al., 2016) para mallas solapadas estructuradas. En esta sección se realizará el mismo estudio para mallas no estructuradas utilizando el esquema de interpolación propuesto anteriormente. Para el estudio de convergencia del método se utilizan dos mallas con distinto tamaño de elementos, con una región central solapada (ver figura 4a) y se va incrementando el número de celdas en ambos dominios simultánea y progresivamente. La convergencia de la malla Chimera se compara con una malla similar conforme (ver figura 4b) y otra malla homogénea (ver figura 4c). Las mallas no estructuradas son creadas automáticamente partiendo de una malla regular estructurada, y luego agregando una pequeña perturbación aleatoria a los nodos.

El problema consiste en la resolución de la ecuación de Poisson sobre los dominios de dos dimensiones presentados en la figura 4. Los bordes permanecen con condición de Neumann homogénea, y en su interior se define una fuente singular dada por la forma:

$$\begin{aligned} f &= \frac{\delta}{\cosh((r - 0,2)/\delta)^2}, \\ r &= \sqrt{(x - x_c)^2 + (y - y_c)^2}, \\ \delta &= 0,025; \end{aligned} \quad (15)$$

La ecuación 15 representa una fuente concentrada en un círculo de radio $r = 0,2$ con su centro posicionado en (x_c, y_c) . En este caso el centro de la fuente se encuentra posicionado en el centro del dominio, es decir, en $(x_c = y_c) = 0,5$. Las matrices de los sistemas para cada dominio son calculadas haciendo uso del software libre *PETScFEM* (Storti et al., 2007), mientras que el algoritmo Chimera es implementado mediante scripts ad-hoc.

En la figura 5a se muestran los resultados obtenidos del análisis de convergencia. El error se presenta en función de un tamaño promedio de elemento en todo el dominio. Para cada tamaño de malla se computó el error 10 veces, ya que de acuerdo a la aleatoriedad de su generación el error variaba, y en base a los resultados se obtuvo un error medio. De los resultados de la figura 5a puede verse que para el esquema propuesto, la convergencia de segundo orden se mantiene, y que su error es muy similar al obtenido por una malla homogénea conforme. Por otro lado, los DCM pueden provocar que la convergencia del método iterativo se vea afectada. El sistema resultante del acople de ambos dominios resulta no ser simétrico definido positivo, sin embargo experiencias numéricas demuestran que la solución puede ser obtenida utilizando el método de Gradientes BiConjugados estabilizado (BICGStab). En la figura 5b se compara la convergencia del método iterativo para un mismo problema, para cuatro casos diferentes. Por un lado se analiza la convergencia de Chimera con BICGStab, siendo el tamaño de elemento

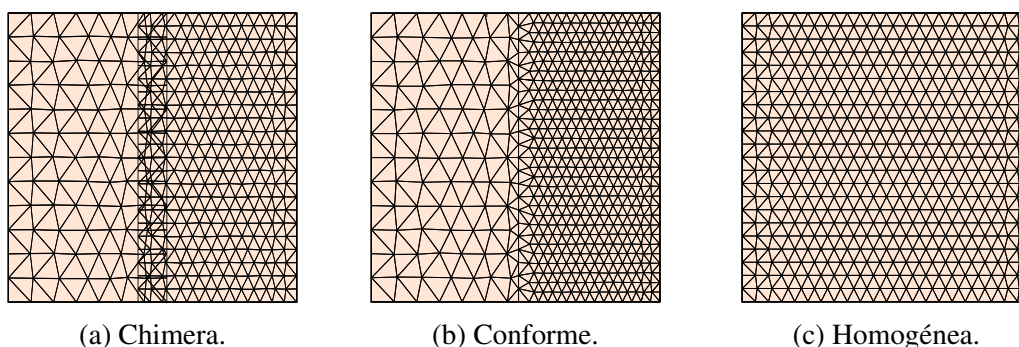


Figura 4: Mallas utilizadas para el análisis de convergencia.

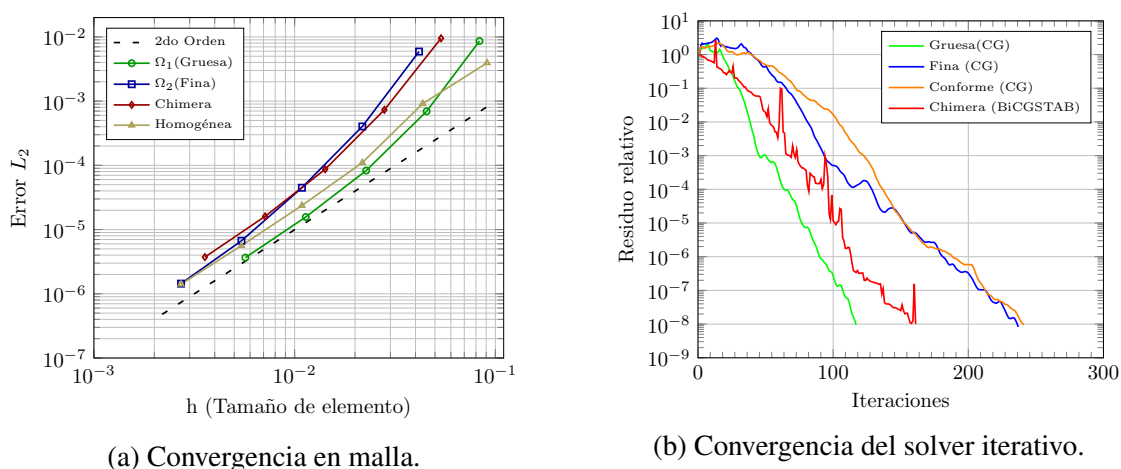


Figura 5: Análisis de convergencia en malla (izquierda) y en iteraciones (derecha).

del subdominio Ω_2 aproximadamente la mitad del tamaño de elemento de Ω_1 . Luego se evaluó la convergencia de dos casos sobre mallas homogéneas resueltos con CG cuyos tamaños de elementos se asemejaban al tamaño de elementos de ambos subdominios de la malla solapada. Es decir, los tamaños de elementos de la malla gruesa y fina son similares al tamaño de elemento en los subdominios Ω_1 y Ω_2 respectivamente. Por último se analizó la convergencia en una malla conforme lo más similar posible a la obtenida por Chimera, es decir, con similar cantidad y distribución de los elementos. Un resumen de los resultados se pueden observar en la tabla 1.

Malla	Elementos	Iteraciones	Error
Gruesa	3872	116	1.0288e-04
Fina	16928	236	2.5301e-05
Chimera	11584	160	1.0454e-04
Conforme	11290	240	1.2644e-04

Tabla 1: Resultados del análisis de convergencia.

De los resultados puede observarse una buena tasa de convergencia del esquema de mallas solapadas, con una cantidad de iteraciones similar a la requerida por la malla gruesa cubriendo todo el dominio. Esto es importante de resaltar ya que los costos por iteración de BiCGStab son equivalentes al de CG, como también lo es el requerimiento de memoria.

3. PROBLEMA DE OPTIMIZACIÓN

Como se mencionó en la introducción de este trabajo, diferentes aplicaciones pueden aprovechar las ventajas del método Chimera. La optimización para la configuración de un sistema es una aplicación directa donde una cantidad dada de objetos, cada uno con su respectiva malla (conformando la malla patch), se mueven libremente sobre una malla de fondo que cubre enteramente el dominio computacional, en busca de la mejor configuración respecto a una función objetivo del problema de optimización. A continuación introduciremos el código abierto de optimización *pyOpt* (Perez et al., 2012) y explicaremos brevemente algunos de sus solvers utilizado en este trabajo. Ellos pueden diferenciarse en dos grandes grupos: optimizadores basados en gradiente y los optimizadores libres de gradientes. Para el primer grupo, *pyOpt* tiene implementada la clase de diferencia finita para calcular el gradiente de la función objetivo respecto a cada variable de diseño.

3.1. Optimizadores

Dentro de los optimizadores implementados en *pyOpt* se encuentran: SNOPT, NLPQL, SLSQP, FSQP CONMIN, MMA/GCMMA, KSOPT, COBYLA, SOLVOPT, ALPSO, NSGA2, ALHSO, MIDACO. En el presente trabajo se utilizó el optimizador basado en población ALPSO (Augmented Lagrangian Particle Swarm Optimizer) para aproximarse a un mínimo global, y luego, a partir de la solución obtenida, hacer un refinamiento utilizando el optimizador basado en gradientes SLSQP (Sequential Least Square Quadratic Programming). El algoritmo de ALPSO desarrollado por Sedlaczek y Eberhard (2006) es utilizado para la resolución de problemas no suaves con restricciones. El algoritmo utiliza multiplicadores de Lagrange para resolver problemas de optimización sujetos a restricciones de igualdad y desigualdad. Al estar basado en el método PSO (Eberhart y Kennedy, 1995) es más probable de encontrar el mínimo global de la función objetivo. El algoritmo se basa en el comportamiento social de bandadas de aves o cardúmenes de peces en busca de alimento mediante la cooperación social. En el trabajo de Sedlaczek y Eberhard (2006) se concluye que el PSO tiene un comportamiento similar a otros algoritmos estocásticos de optimización, especialmente a algoritmos basados en estrategias de evolución.

El optimizador SLSQP, de la familia de los algoritmos de Programación Secuencial Cuadrática (Kraft y Schnepfer, 1989), es una modificación del solver implementado por Lawson y Hanson (1995). El algoritmo utiliza el método de Han-Powell quasi-Newton con una actualización BFGS de la matriz-B y una función de test L1 en el algoritmo del line-search. Para calcular los gradientes dos opciones se encuentran implementadas: Diferencia Finita y Paso Complejo. Éstas pueden ser calculadas en serie o en paralelo. En las siguientes páginas se mostrarán algunos resultados evaluando estas cuestiones.

3.2. Ejemplo numérico

El algoritmo de Chimera junto a la plataforma de optimización fueron implementadas en Seshat (2018), un cluster de 69 nodos ubicado en el Centro de Computación del instituto CI-MEC. Cada nodo tiene un procesador Intel(R) Xeon(R) CPU E5-1620 v2 @ 3.70GHz (1 CPU x 4 cores) y 16 GB de RAM (DDR3 y 1600 Mhz). El cluster consta de red InfiniBand, lo que es fundamental para computación de alto rendimiento. A continuación se presenta un ejemplo numérico sencillo a fines de evaluar convergencia y speedup.

El problema que se presenta a continuación (ver ecuación 16) consiste en minimizar la temperatura máxima del dominio de dos dimensiones Ω en el cual se resuelve la ecuación de con-

ducción de calor con una conductividad constante en todo el dominio igual a la unidad, para un estado estacionario. Dicho dominio tiene una fuente de calor constante igual a la unidad. Los bordes de Ω_1 se encuentran completamente aislados (ver figura 6), y una cantidad de huecos, cada uno con su respectiva malla, tienen impuesto en sus nodos interiores una temperatura baja (en este caso $\Gamma_2 = 0.1[K]$) a modo de actuar de sumidero.

$$\begin{aligned} & \underset{\bar{x}}{\text{mín}} && f(\bar{x}) = \text{máx}(\bar{\phi}) \\ \text{subject to} && g_j(\bar{x}) = c_j - c_{\text{mín}} \geq 0 && j = 1, \dots, m \\ && x_{i_L} \leq x_i \leq x_{i_U}, && i = 1, \dots, n \end{aligned} \quad (16)$$

Como restricción al problema, la distancia entre cada par de centros de huecos (c_j) debe ser mayor o igual a un valor dado ($c_{\text{mín}}$) con el fin de evitar colisiones entre ellos (ver ecuación 16). Entonces en el problema se tiene un total de $m = ((n_h - 1) \times n_h)/2$ restricciones de desigualdad, siendo n_h la cantidad de sumideros. De la ecuación 16, x_{i_L} y x_{i_U} son los valores mínimos y máximos que pueden tomar las variables. Para este ejemplo, dichos valores fueron fijados en 0.2 y 0.8 respectivamente. Esto quiere decir que los centros pueden moverse solo dentro del cuadrado $[0.2, 0.8] \times [0.2, 0.8]$. Estos valores límites fueron establecidos con el fin de mantener Ω_2 siempre superpuesto a Ω_1 . En este problema no se tienen restricciones de igualdad.

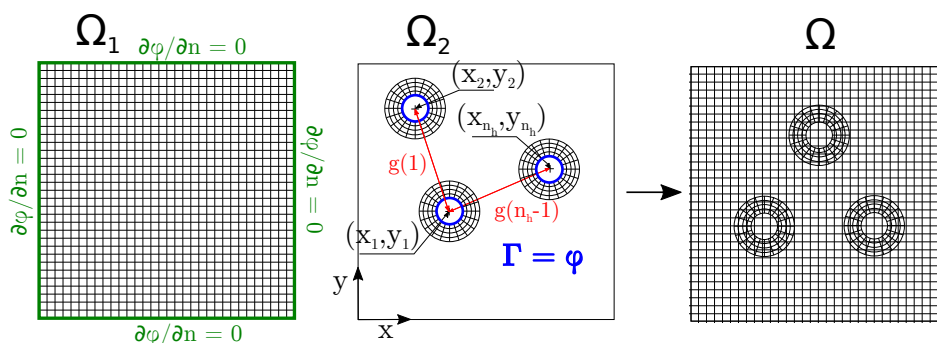


Figura 6: Mallas de fondo y malla patch para el esquema de optimización .

Las variables de diseño se condensan en un vector (ver ecuación 17) de tamaño $n = n_h \times 2$, siendo n_h la cantidad de huecos, que contiene las coordenadas en x e y de los centros de cada orificio. En este caso se optimizaron 16 variables de diseño ($n_h = 8$).

$$\bar{x} = [x_1, y_1, x_2, y_2, \dots, x_{n_h}, y_{n_h}] \quad (17)$$

En la figura 7 se puede ver que para una condición inicial aleatoria (izquierda), la temperatura máxima del dominio alcanza un valor de 0.1761 [K], mientras que para una posición optimizada de los orificios (derecha), la temperatura máxima disminuye a 0.1494 [K].

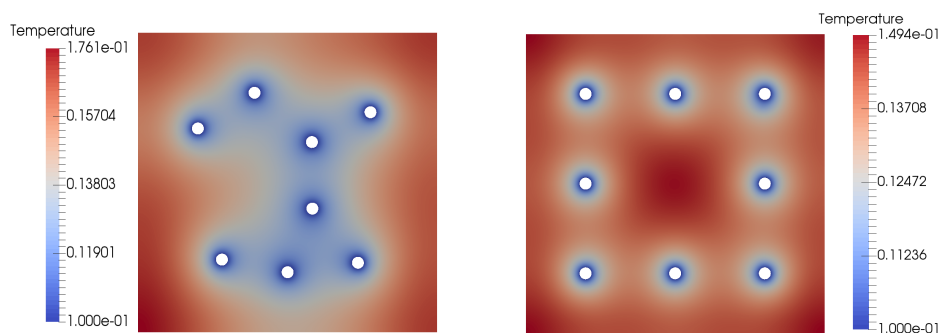


Figura 7: Condición inicial (izquierda) y resultado de la optimización (derecha) del problema de optimización para 8 orificios.

3.3. Convergencia

Para analizar la convergencia de los optimizadores debemos primero definir el error relativo de la función objetivo, como se muestra en la ecuación 18. Al no tener una solución analítica para la función objetivo, una solución muy refinada se utilizó para el cálculo de dicho error.

$$\varepsilon_f = \left\| \frac{f(x^*) - f(x_{\text{exact}})}{f(x_{\text{exact}})} \right\|_2 \quad (18)$$

Puede verse de la figura 8 y la tabla 2 que para un algoritmo basado en población, como en este caso el ALPSO, se requiere un gran número de evaluaciones de la función objetivo para converger a la solución, mientras que para un optimizador basado en gradientes como ser el SLSQP, requiere menor número de evaluaciones pero es menos probable que se acerque a un mínimo global. Por esto es que para este problema se utilizó una estrategia de refinamiento automático implementada en *pyOpt* en la cual se utiliza un optimizador basado en población para intentar acercarse al mínimo global, y luego a partir de su solución, realizar un refinamiento con un optimizador basado en gradiente.

Optimizer	Func. evaluations	error
ALPSO	784	1.6030e-04
SLSQP	36	6.6881e-08

Tabla 2: Convergencia de los optimizadores.

3.4. Speed-up

De acuerdo a la ley de Amdahl, el speedup ideal para un cálculo en paralelo se define como:

$$S(n) = \frac{1}{(1-p) + \frac{p}{n}} \quad (19)$$

donde $S(n)$ es el speed-up teórico, p es la fracción del algoritmo que puede ser paralelizada y n es el número total de cores. En el caso del optimizador SLSQP, la porción de código paralelizable corresponde al cálculo del gradiente, que para este caso representa un 77% del tiempo total de cada evaluación. Esta fracción depende de la cantidad de variables de diseño, eso quiere decir que cuanto mayor cantidad de variables de diseño el problema tenga, mayor será ese

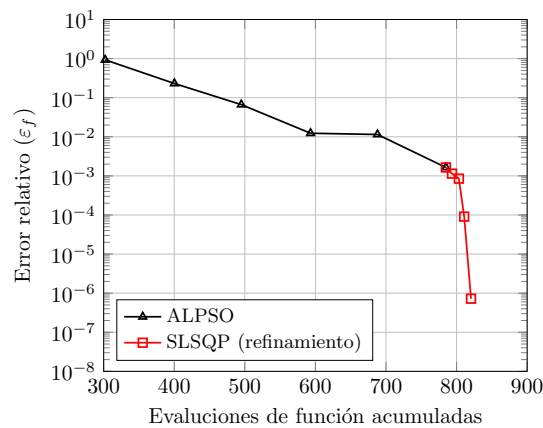
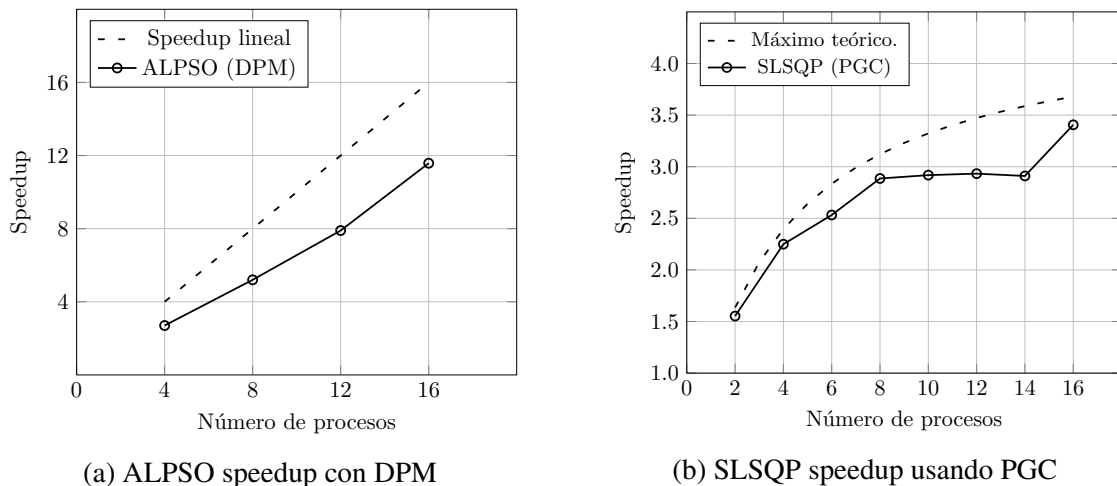


Figura 8: Convergencia del ALPSO con refinamiento por SLSQP.

porcentaje y más paralelizable será el cálculo del gradiente. Debido a la distribución estática implementada en *pyOpt* para el cálculo del gradiente, puede verse de la figura 9b que cuando la cantidad de variables de diseño es un múltiplo entero del número de procesos, la escalabilidad se aproxima a la teórica. Cuando esa condición no se cumple, el incremento de procesos no muestra una mejora. Por otro lado, de la figura 9a se puede ver que el optimizador ALPSO al tener implementada la Distribución Dinámica de Procesos (DPM), presenta un speedup cercano al lineal.



(a) ALPSO speedup con DPM

(b) SLSQP speedup usando PGC

Figura 9: Speed-up de la paralelización de los algoritmos de optimización.

4. OPTIMIZACION DE LA TRANSFERENCIA TÉRMICA EN UN ÁLABE

La mejora en el rendimiento de las turbinas de gas se encuentra en parte limitada por la máxima temperatura de los gases de combustión que entran a ella. Esta limitación está vinculada a la resistencia termo-mecánica de los álabes de la turbina. A modo de reducir la temperatura de trabajo del perfil, un número de pasajes refrigerantes se encuentran dispuestos en su interior. De esta forma, mejorar su resistencia termo-mecánica implica una optimización en la distribución de las tensiones y temperaturas en él. En 1983 Hylton et al. (1983) trabajaron en un experimento en los laboratorios de la NASA en los cuales disponían de un arreglo de tres álabes de una turbina sometidos a distintas condiciones de trabajo. La superficie de los álabes se

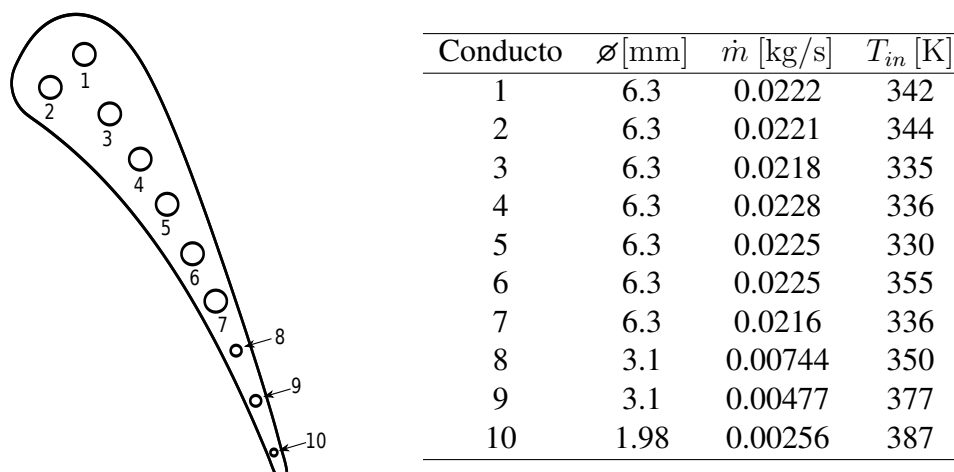


Figura 10: Geometría del Alábe C3X (izquierda) y datos del refrigerante (derecha).

encontraba instrumentada por aproximadamente 80 termocuplas y 30 orificios para la medición de la presión estática. Todo el instrumental se encontraba dispuesto sobre el plano medio del alábe. Además, cada tubo refrigerante se encontraba también instrumentado con termocuplas y orificios de presión estática dispuestos en su entrada y salida. Los valores de los flujos máscicos para cada tubo fueron obtenidos utilizando placas orificios.

4.1. Geometría y condiciones de carga

Dos geometrías diferentes fueron analizadas durante la experiencia en la NASA. En este trabajo se escogió el alábe C3X para su optimización. Los parámetros de funcionamiento como ser la distribución de la temperatura sobre la superficie del perfil, flujo máscico y temperatura del refrigerante, fueron elegidos de la experiencia *RUN 157*. Tanto la geometría del alábe, como también las condiciones del fluido refrigerante para cada pasaje se encuentran detalladas en la figura 10.

Resolver un análisis de Transferencia de Calor Conjugado (CHT) para cada evaluación de la función objetivo sería computacionalmente muy costoso. Por esto es que para el siguiente estudio se realizan algunas suposiciones. El fluido refrigerante se asume que presenta un comportamiento estacionario (esto significa que sus propiedades físicas no varían en las simulaciones). También el flujo máscico y la temperatura son consideradas constantes y uniformes en cada pasaje. A continuación se muestra cómo es calculado el coeficiente pelicular en el interior de los tubos.

$$h_{\text{film}} = \frac{\lambda N_u}{d} \quad (20)$$

Siendo λ la conductividad térmica del refrigerante, d el diámetro hidráulico y N_u es el número de Nusselt que se define experimentalmente de acuerdo a la correlación de Dittus-Boelter para flujos turbulentos que establece:

$$N_u = 0,023 Re^{0,8} Pr^{0,4} \quad (21)$$

donde Pr y Re se refiere a los números de Prandtl y Reynolds respectivamente, donde el número de Reynolds se define cómo:

$$Re = \frac{\rho \nu d}{\mu} \quad (22)$$

siendo ρ y μ la densidad y la viscosidad del refrigerante, y v la velocidad del fluido. El número de Prandtl se define como:

$$Pr = \frac{C_p \mu}{\lambda} \quad (23)$$

Se debe tener en cuenta que la correlación de Dittus-Boelter es válida cuando se satisfacen las siguientes condiciones para el fluido:

- $0,6 < Pr < 160$
- $Re > 10000$
- $L/D > 10$

De las ecuaciones 20-23 se puede ver que para valores constantes de la conductividad térmica, densidad, viscosidad y calor específico del refrigerante, el coeficiente pelicular de transferencia térmica de cada hueco es sólo función de su diámetro. Además se define una condición de borde tipo Dirichlet constante sobre la superficie del álabe de acuerdo a datos experimentales medidos por las termocuplas. Se realiza una interpolación cúbica con el fin de imponer dicha condición de borde en todos los nodos de la superficie.

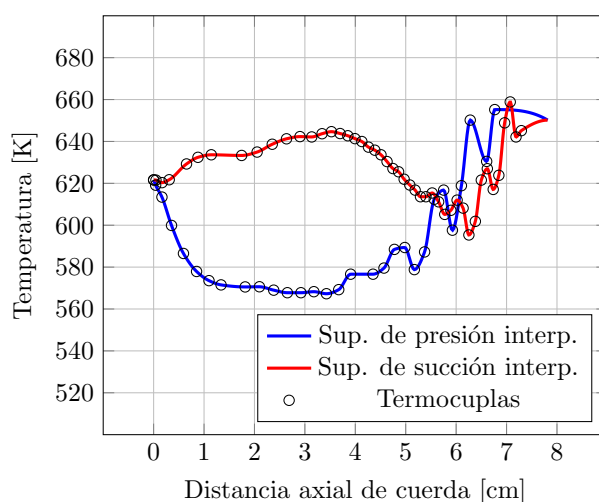


Figura 11: Interpolación de la temperatura medidas por las termocuplas para el análisis de FEM.

La conductividad térmica del acero inoxidable ASTM 310 (Harvey, 1982) se aproxima utilizando la siguiente función lineal:

$$\lambda_s = 0,0115 T + 9,9105 \quad [\text{W}/(\text{m K})] \quad (24)$$

La variación de la viscosidad μ y la conductividad térmica λ del fluido refrigerante en función de la temperatura puede aproximarse mediante la fórmula de Sutherland (Sutherland, 1893) que establece:

$$\mu(T) = \mu_0 \left(\frac{T}{T_0} \right)^{3/2} \frac{T_0 + S}{T + S} \quad (25)$$

$$\lambda_f(T) = \lambda_0 \left(\frac{T}{T_0} \right)^{3/2} \frac{T_0 + S}{T + S} \quad (26)$$

siendo para el aire: $\mu_0 = 1,7894 \times 10^{-5}$ [Pa · s], $T_0 = 273,11$ [K], $\lambda_0 = 0,0261$ [W/(m K)] y $S = 110,56$.

4.2. Modelo matemático del problema de optimización

El modelo matemático para el problema de optimización de la refrigeración del perfil puede definirse como:

$$\begin{aligned} \underset{\bar{x}}{\text{mín}} \quad & T_{media}(\bar{x}) \\ \text{sujeto a} \quad & g_j(\bar{x}) = c_j - c_{min} \geq 0, & j = 1, \dots, m_e, \\ & \nabla T(\bar{x}) \leq \sigma_{max}, & j = m_e + 1, \dots, m, \\ & x_{i_L} \leq x_i \leq x_{i_U}, & i = 1, \dots, n \end{aligned} \quad (27)$$

La función objetivo a minimizar es la temperatura media del perfil (T_{media}), sujeta a la restricción para que el gradiente de la temperatura no supere un valor específico (σ_{max}). Como las tensiones termomecánicas de un componente dependen directamente de los gradientes térmicos en él, para asegurar valores aceptables de dichas solicitaciones en el álabe, los gradientes térmicos se imponen como una restricción en el problema de optimización. Las variables de diseño están condensadas en el vector \bar{x} (ver ecuación 28).

$$\bar{x} = [\underbrace{r_1, \theta_1, d_1}_{\text{Conducto 1}}, \underbrace{r_2, \theta_2, d_2}_{\text{Conducto 2}}, \underbrace{x_3, y_3, d_3}_{\text{Conducto 3}}, \dots, \underbrace{x_{n_h}, y_{n_h}, d_{n_h}}_{\text{Conducto } n_h}] \quad (28)$$

Para los conductos 1 y 2, es decir para $i = 1, 2$, las variables r_i y θ_i corresponden a la posición de sus centros en coordenadas polares para el dominio transformado (ver figura 12) y d_i para sus diámetros. Mientras que para el resto de los conductos, las variables x_i e y_i corresponden a las coordenadas del centro del pasaje i (para $i = 3, 4, \dots, 10$) en coordenadas cartesianas y el diámetro del conducto también se encuentra representado por la variable d_i . Los límites máximos y mínimos (x_{i_U} y x_{i_L}) que puede tomar cada variable vendrá dado por la transformación de coordenadas explicada en la siguiente sección.

4.3. Transformación del sistema de coordenadas de las variables de diseño

A modo de restringir el movimiento de los tubos en la optimización, se realiza una transformación de coordenadas para cada variable de diseño similar a la realizada por Wang et al. (2015) donde el centro de cada pasaje tiene permitido moverse dentro de cierta región (ver figura 12). Para los agujeros 1 y 2, se realiza una transformación polar según la ecuación 29.

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x_o \\ y_o \end{bmatrix} + \begin{bmatrix} r \cos \alpha \\ r \sin \alpha \end{bmatrix}. \quad (29)$$

Mientras que para los centros de los pasajes 3 a 10, se realiza la transformación de la ecuación 30.

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x_o \\ y_o \end{bmatrix} + \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \cdot \begin{bmatrix} x' \\ y' \end{bmatrix} \quad (30)$$

Un detalle de las subregiones asignadas para cada centro de variable y su transformación de coordenadas puede verse en la figura 12.

4.4. Convergencia en malla

Una vez configurado el modelo de FEM, se realizó un estudio de convergencia en malla. Para este estudio se utilizó una temperatura homogénea en la superficie del perfil de 600 [K].

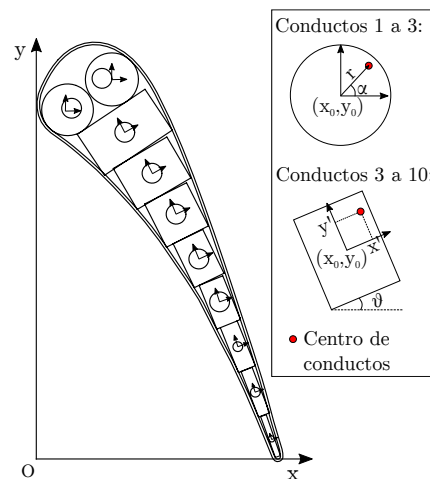


Figura 12: Transformación del sistema de coordenadas de los huecos.

No se utilizaron las temperaturas obtenidas en las termocuplas para evitar que la interpolación introduzca errores en el estudio de convergencia. Se analizaron distintas mallas no estructuradas con número creciente de elementos triangulares, variando de 5600 a 77000 celdas. La malla de fondo del perfil se genera usando la plataforma libre *SALOME* (Ribes y Caremoli, 2007), mientras que la malla superpuesta de los pasajes se genera por un script ad-hoc. El número total de celdas se obtiene de la suma total de elementos de la malla de fondo y elementos de la malla patch. Para cada caso, la malla patch estaba compuesta por aproximadamente la quinta parte de celdas respecto a la malla de fondo. Puede verse de los resultados (ver figura 13) que para las primeras tres mallas, las variaciones en la temperatura media del dominio son muy grandes. Mientras que para mallas de número mayor a 26.000 celdas, la variación relativa en los valores de salida se mantienen debajo del 0.098 %. Procurando priorizar la robustez del método de optimización utilizando Chimera, se utilizó una malla compuesta por un total de 61.758 elementos. La malla de fondo se encuentra compuesta por 48.960 elementos triangulares

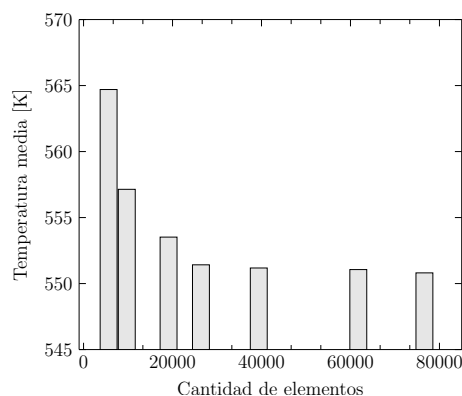


Figura 13: Convergencia en malla de la temperatura media del perfil.

lineales, mientras que la malla patch se compone por un total de 12.800 elementos lineales triangulares lineales (1280 elementos rodeando cada tubo). A modo de representación se puede ver un esquema de las mallas utilizadas en la figura 14, que al superponerlas y resolver, se tiene la solución a un único sistema.

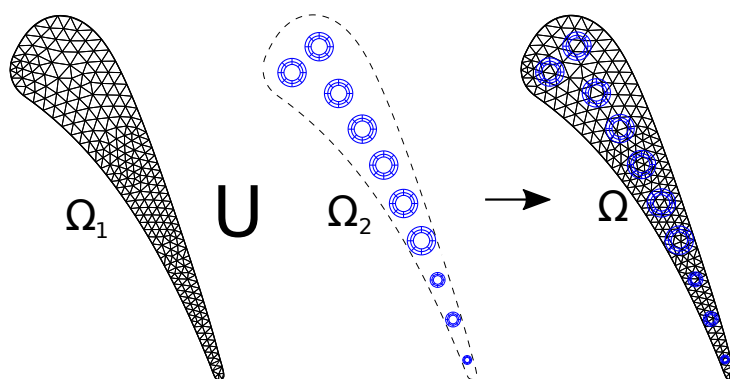


Figura 14: Malla de fondo y malla patch del esquema Chimera

4.5. Resultados

El optimizador efectuó más de mil evaluaciones de la función objetivo, cubriendo gran parte del dominio computacional (ver figura 15a). Los resultados de convergencia del optimizador son presentados en la figura 15b. En la figura 16b donde puede verse que para una refrigeración óptima del álabe, el diámetro de los pasajes tiende a su mínimo, mientras que en el peor caso obtenido por el optimizador (ver figura 16c), los diámetros tienden al máximo posible. Esto se debe a que la disminución en los diámetros de los tubos refrigerantes conllevan a un aumento en el coeficiente pelicular de transferencia térmica, que pesa más que la disminución de la superficie de transferencia. Se observa que si bien los pasajes 8, 9 y 10 prácticamente se mantienen equidistantes, tienden a desplazarse hacia el borde de fuga donde se encuentran las mayores temperaturas. En cuanto a los huecos 1 y 2, sus posiciones no varían en forma significativa respecto a la posición original, mientras que los pasajes 3, 4 y 5, tienden a desplazarse hacia la superficie de succión donde las temperaturas son mayores.

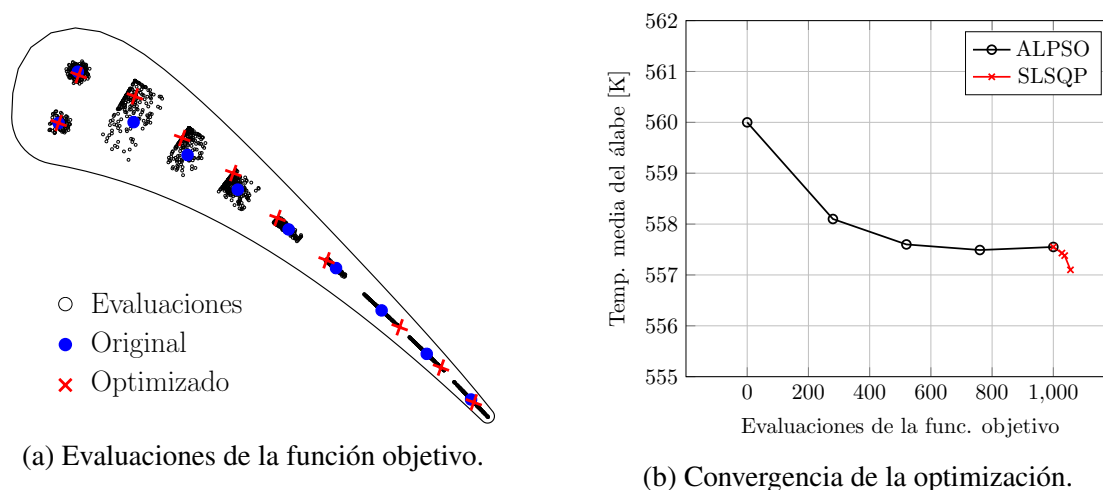


Figura 15: Resultados de la convergencia del esquema de optimización.

Un resumen de los resultados pueden verse en la tabla 3. La temperatura media del álabe logró reducirse en más de 3 [K] (ver figura 16b), mientras que en el peor caso analizado por el optimizador (ver figura 16c) la temperatura media del perfil llega a aumentar en 8 [K]. El peor escenario es presentado con el fin de comprender mejor el fenómeno térmico. Puede verse que para las condiciones físicas dadas, una mala refrigeración viene acompañada con el incre-

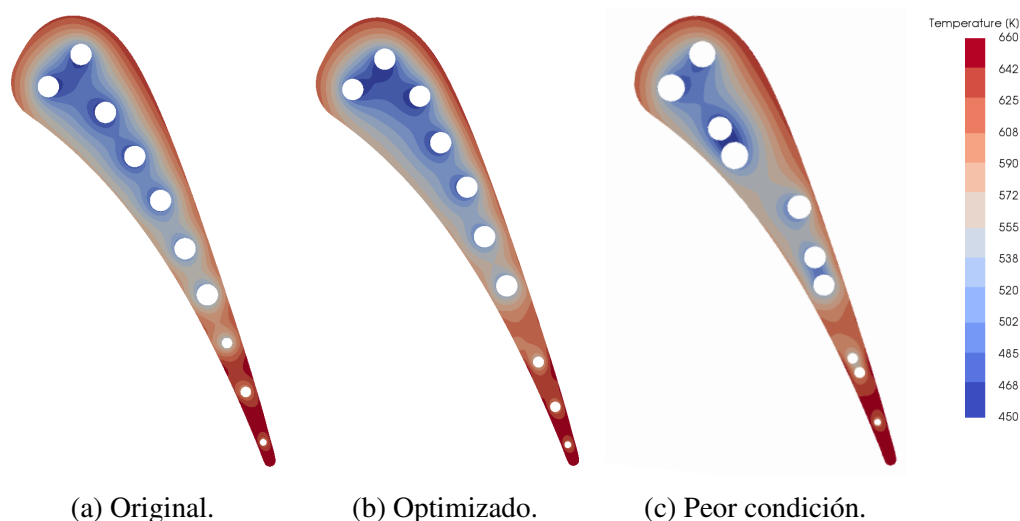


Figura 16: Campos de temperatura para la configuración original, la optimizada, y la peor condición obtenida por el optimizador.

mento en los diámetros de los tubos, debido a la disminución del coeficiente pelicular, y de la inapropiada separación entre ellos.

	Original	Optimizado	Peor cond.
T_{med} [K]	560.10	557.02	568.03

Tabla 3: Temperatura media del álabe de los distintos casos.

5. CONCLUSIONES

Un esquema Chimera para mallas superpuestas fue presentado como también su formulación algebraica y un algoritmo para resolver el sistema con un solver iterativo. El esquema propuesto fue evaluado resolviendo la ecuación de Poisson sobre un dominio de dos dimensiones con solapamiento entre mallas. Los resultados muestran la conservación de la convergencia de segundo orden espacial, y un error en la norma L_2 similar al obtenido por una malla conforme semejante. Para esto fue necesaria la implementación de un interpolador de tercer orden para mallas no estructuradas. También se estudió la convergencia del solver para mallas solapadas, probando una muy buena convergencia respecto a la resolución de un problema de malla conforme.

Luego se realizó una breve introducción de la plataforma de optimización y de algunos de sus optimizadores utilizados en este trabajo. El acople del optimizador con el algoritmo de Chimera fue evaluado en un problema simple de dos dimensiones para un dominio cuadrado con sus bordes completamente aislados, con huecos refrigerantes en su interior conformando la malla patch. El problema consistió en optimizar la ubicación de dichos agujeros con el fin de minimizar la temperatura máxima del dominio. Los resultados fueron los esperados, es decir, una distribución simétrica y equidistante de los mismos. También fue evaluado el speedup tanto para el cálculo del gradiente del optimizador SLSQP, como para la evaluación de la función objetivo para el optimizador ALPSO. El optimizador basado en población, mediante la utilización del distribuidor dinámico de procesos (DPM) mostró un speedup casi lineal, mientras que para el optimizador basado en gradiente, su speedup resulta muy cercano al teórico cuando la cantidad

de variables de diseño es múltiplo del número de procesos, en caso contrario, el speedup no mejora con la adición de más procesadores.

Por último se aplicó el esquema a un problema de optimización del sistema de refrigeración del álabe de una turbina de gas. Tanto las condiciones de contorno como las propiedades del álabe y del aire refrigerante fueron obtenidas de los experimentos. Se redujo la temperatura media del álabe en 3 [K], que si bien porcentualmente no representa una gran diferencia respecto a la temperatura media del sistema original, en el trabajo de Wang et al. (2015) se muestra que el método es válido para, a partir de los resultados obtenidos del modelo de dos dimensiones, realizar un análisis completo de CHT en tres dimensiones y así lograr reducir en mayor porcentaje la temperatura máxima del álabe, que es el principal objetivo a optimizar en una condición de funcionamiento real. Por otro lado, para el peor caso obtenido por el optimizador, la temperatura media del álabe aumentó en 8 [K], lo cual implicaría que en un análisis completo por CHT se esperase un marcado aumento en la temperatura máxima sobre la superficie del perfil. La estrategia de optimización utilizando el esquema de Chimera y técnicas de paralelismo presentó buenos resultados respecto a técnicas de remallado tradicionales utilizadas comúnmente para abordar problemas de optimización.

AGRADECIMIENTOS

Este trabajo ha recibido aporte financiero del Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET, Argentina, grant PIP 11220150100588CO), Agencia Nacional de Promoción Científica y Tecnológica (ANPCyT, Argentina, grants PICT 2660–14, PICT-2015-2904, PICT 2016-0708.), y Red CYTED 2015 CAD-ING 516RT0512. Este trabajo ha sido realizado mediante el uso de software de *Free Software Foundation/GNU-Project* como GNU–Linux OS, GNU–GCC, GNU–Octave, compiladores GCC/G++, así como otros recursos de código abierto (Open Source) tales como PETSfEM, ParaView, L^AT_EX, pyOpt, SALOME, entre otros.

REFERENCIAS

- Andrew M. W., Aaron J. K., William M. C., y Robert L. M. Validation of the Strand Grid Approach. *AIAA Computational Fluid Dynamics*, 2009.
- Blades E.L. y Marcum D.L. A sliding interface method for unsteady unstructured flow simulations. *International Journal for Numerical Methods in Fluids*, 53(3):507–529, 2007.
- Dennis B.H., Egorov I.N., Dulikravich G.S., y Yoshimura S. Optimization of a large number of coolant passages located close to the surface of a turbine blade. En *ASME Turbo Expo 2003, collocated with the 2003 International Joint Power Generation Conference*, páginas 13–19. American Society of Mechanical Engineers, 2003.
- Eberhart R. y Kennedy J. A new optimizer using particle swarm theory. En *Micro Machine and Human Science, 1995. MHS'95., Proceedings of the Sixth International Symposium on*, páginas 39–43. IEEE, 1995.
- Guennebaud G., Jacob B., et al. Eigen v3. <http://eigen.tuxfamily.org>, 2010.
- Harvey P.D. *Engineering properties of steel*. American Society for Metals Metals Park, OH, 1982.
- Houzeaux G., Cajas J., Discacciati M., Eguzkitza B., Gargallo-Peiró A., Rivero M., y Vázquez M. Domain decomposition methods for domain composition purpose: chimera, over-set, gluing and sliding mesh methods. *Archives of Computational Methods in Engineering*, 24(4):1033–1070, 2017.

- Hylton L., Mihelc M., Turner E., Nealy D., y York R. Analytical and experimental evaluation of the heat transfer distribution over the surfaces of turbine vanes. 1983.
- Kao K.H., Liou M.S., y Chow C.Y. Grid adaptation using chimera composite overlapping meshes. *AIAA journal*, 32(5):942–949, 1994.
- Kraft D. y Schnepfer K. Slsqp—a nonlinear programming method with quadratic programming subproblems. *DLR, Oberpfaffenhofen*, 1989.
- Lawson C.L. y Hanson R.J. *Solving least squares problems*, volumen 15. Siam, 1995.
- Meakin R. An efficient means of adaptive refinement within systems of overset grids. En *12th Computational Fluid Dynamics Conference*, página 1722. 1995.
- Nowak G. y Wróblewski W. Optimization of blade cooling system with use of conjugate heat transfer approach. *International Journal of Thermal Sciences*, 50(9):1770–1781, 2011.
- Perez R.E., Jansen P.W., y Martins J.R.R.A. pyOpt: A Python-based object-oriented framework for nonlinear constrained optimization. *Structures and Multidisciplinary Optimization*, 45(1):101–118, 2012. doi:10.1007/s00158-011-0666-3.
- Prewitt N.C., Belk D.M., y Shyy W. Parallel computing of overset grids for aerodynamic problems with moving objects. *Progress in Aerospace Sciences*, 36(2):117–172, 2000.
- Ribes A. y Caremoli C. Salome platform component model for numerical simulation. En *Computer Software and Applications Conference, 2007. COMPSAC 2007. 31st Annual International (http://www.salome-platform.org/)*, volumen 2, páginas 553–564. IEEE, 2007.
- Sedlaczek K. y Eberhard P. Using augmented lagrangian particle swarm optimization for constrained problems in engineering» using augmented lagrangian particle swarm optimization for constrained problems in engineering. *Structural and Multidisciplinary Optimization*, 32(4):277–286, 2006.
- Seshat. <http://www.cimec.org.ar/c3/seshat/>, 2018.
- Storti L., Garelli M., Storti J., y D’Elía. A chimera method based on dirichlet-dirichlet coupling and pasting penalization operators. *ENIEF 2016 - XXII Congreso sobre Métodos Numéricos y sus Aplicaciones*, 2016.
- Storti M., Nigro N., Paz R., Dalcin L., y Lopez E. Petsc-fem, a general purpose, parallel, multi-physics fem program. *International Center of Computational Method in Engineering (CIMEC), Argentina http://www.cimec.org.ar/petscfem*, 2007.
- Sutherland W. Lii. the viscosity of gases and molecular force. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 36(223):507–531, 1893.
- Wang B., Zhang W., Xie G., Xu Y., y Xiao M. Multiconfiguration shape optimization of internal cooling systems of a turbine guide vane based on thermomechanical and conjugate heat transfer analysis. *Journal of Heat Transfer*, 137(6):061004, 2015.