

DESARROLLO DE SOFTWARE PARA RESOLVER PROBLEMAS DE TRANSFERENCIA DE CALOR

Núñez Aguayo Miguel Serafín (1), Cano Banda Fernando (2), Hernández Guerrero Abel (3)

¹ [Licenciatura en Ingeniería Mecánica, Universidad de Guanajuato] | Dirección de correo electrónico: [miguel_nunez95@hotmail.com]

² [Departamento de Ingeniería Mecánica, División de Ingenierías, Campus Irapuato Salamanca, Universidad de Guanajuato] | Dirección de correo electrónico: [f.canobanda@gmail.com]

³ [Departamento de Ingeniería Mecánica, División de Ingenierías, Campus Irapuato Salamanca, Universidad de Guanajuato] | Dirección de correo electrónico: [abel@ugto.mx]

Resumen

El tema de esta investigación es el desarrollo de un software enfocado a estudiantes para resolver problemas de transferencia de calor en estado estable en dos dimensiones. El código está basado en el método de diferencias finitas, escrito en el lenguaje de programación MATLAB. El producto es un software funcional en el que se pueden plantear problemas de transferencia de calor por medio de imágenes y usando la interfaz gráfica del programa para describir el problema. La malla que se usa en cada problema depende de la resolución de las imágenes, usando la analogía de que cada pixel en la imagen representa un nodo, de este modo se pueden plantear de forma rápida e intuitiva los problemas. La validación de los resultados consiste en la comparación de distintos casos analizados en ANSYS-FLUENT y en el software desarrollado; la forma elemental de plantear los problemas y el funcionamiento de ambos programas, el propuesto en este trabajo y FLUENT, es esencialmente distinta así que se esperan variaciones. Los resultados varían con valores máximos encontrados en el orden de hasta 0.3% dependiendo de la complejidad de las geometrías, condiciones de frontera y las diferencias entre las mallas usadas en cada caso.

Abstract

The theme for this investigation is the development of a software to be used by students to solve problems of heat transfer in steady state in two dimensions. The code is based on the finite difference method and is written using the programming language MATLAB. The product is a working software in which it is possible to pose the heat transfer problems by the use of pictures and using the graphic interface of the software to describe the problem. The mesh used for each problem depends on the resolution of the pictures, using the analogy that each pixel in the picture represents a node, in this way it is possible to pose a problem quickly and intuitively. The validation of the results consists in the comparison of different cases analyzed on ANSYS-FLUENT and the developed software. The elemental way to pose the problems and the workings of both programs, the proposed in this work and FLUENT, is essentially different so variations are expected. The results vary with maximum found values in the order of 0.3% depending on the complexity of the geometries, boundary conditions and the differences between the meshes used in each case.

Palabras Clave

Método de diferencias finitas; Transferencia de Calor; Herramienta; 2D; Imágenes

INTRODUCCIÓN

Hoy en día los paquetes de software de ingeniería asistida por computadora (CAE por sus siglas en inglés) son ampliamente usados para encontrar la solución de problemas prácticos que son difíciles o imposibles de solucionar de manera exacta. Estos programas tienen amplias capacidades para resolver problemas complejos pero su uso correcto es también de una complejidad considerable para estudiantes que todavía no tienen las habilidades requeridas para usarlos, es por eso que surgió la idea de desarrollar una herramienta que fuera intuitiva y fácil de usar para apoyar el entendimiento que los estudiantes tienen de los fenómenos físicos y de las herramientas de CAE.

El método de diferencias finitas es de gran sencillez conceptual y se puede usar como una forma de encontrar la solución aproximada de los problemas de transferencia de calor en dos dimensiones. El software desarrollado sigue el mismo método descrito en el libro de Bergman, Incropera, Dewitt, y Lavine [1] además de las sugerencias encontradas en el libro de Holman [2] para plantear las ecuaciones de los distintos casos en los nodos. Se tomó nota del artículo de Gilbert, Moler y Schreiber [3] respecto a las ventajas de usar matrices dispersas para un más eficiente consumo de memoria y se usaron de referencia los ejemplos de Davis [4] para la implementación práctica y reducción del tiempo de computo del programa. El software obtenido es una herramienta sencilla que cuenta con la interfaz gráfica necesaria para plantear los problemas más comunes de transferencia de calor en estado estable en dos dimensiones que personas en proceso de aprendizaje de los conceptos de transferencia de calor pueden usar.

MATERIALES Y MÉTODOS

El algoritmo se basó en la implementación programática del método de diferencias finitas para transferencia de calor en estado estable en 2D, considerando las propiedades del fluido (Temperatura y coeficiente convectivo) como constantes conocidas.

Se propusieron las ecuaciones de nodo para casos más generales que los descritos en los libros llegando a identificar hasta 60 casos diferentes, incluyendo diferentes configuraciones del mismo caso y el manejo de defectos en la malla, que posteriormente se asignan a cada nodo analizando las condiciones descritas por el usuario en un proceso de barrido de la imagen. También se propusieron ecuaciones para reducir los errores debidos a geometrías complicadas como lo son paredes inclinadas y puntas que usualmente tienen los errores más considerables en mallas con elementos cuadrados que es como se maneja para todos los casos en este software.

Tanto el código base como la interfaz gráfica se escribieron en el lenguaje de programación MATLAB.

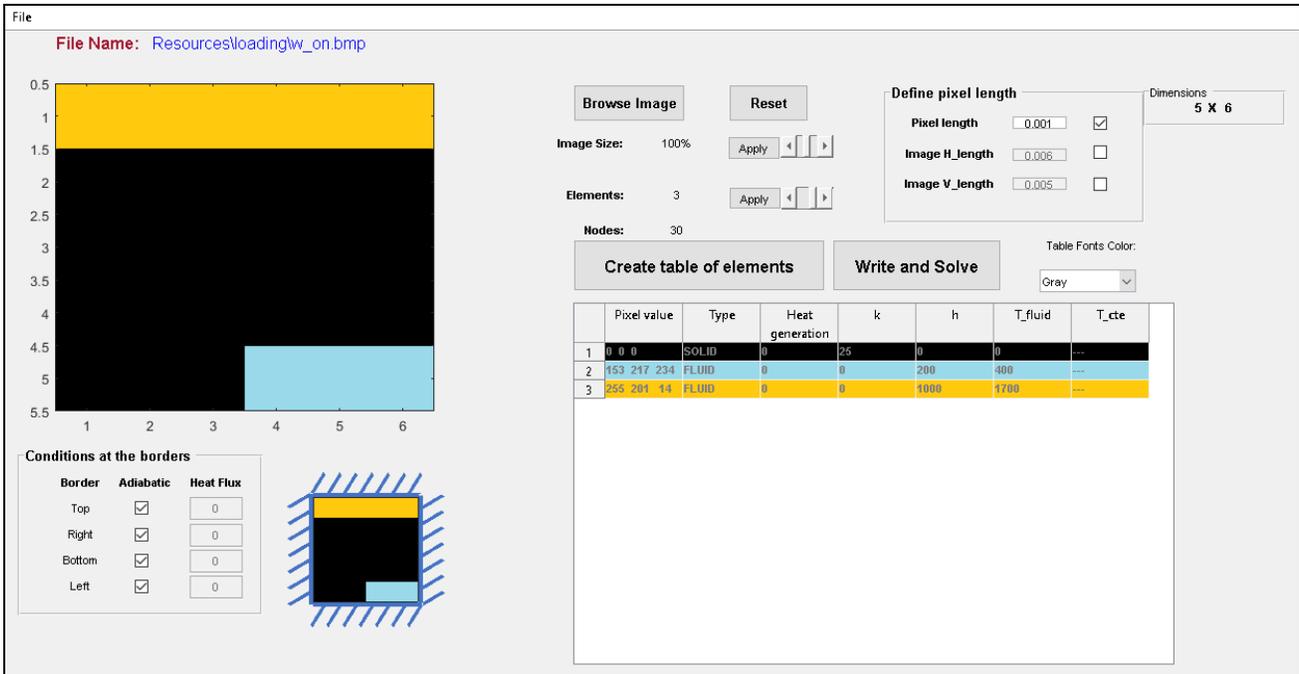
Se usaron técnicas de manejo de matrices dispersas para mejorar el desempeño de uso de memoria en el proceso de creación de las matrices que representan el sistema de ecuaciones a resolver del problema.

DEPURACIÓN DE ERRORES

La validación consistió en la elaboración de problemas más complejos en que las debilidades del software desarrollado se pusieran a prueba usando geometrías relativamente más complejas y diversas condiciones de frontera. El comportamiento anormal se corrigió revisando las ecuaciones planteadas para los casos particulares de cada nodo y corrigiendo los errores encontrados, llevándose de este modo el proceso de depuración.

EL SOFTWARE

La interfaz gráfica del programa permite definir el problema a detalle por medio de la generación de una tabla que identifica los colores que son únicos en la imagen por su valor en la escala RGB en cada pixel. En la tabla el usuario puede describir de forma concisa qué representa cada color, si es un sólido o un fluido, si tiene generación interna de energía, el valor de su conductividad térmica si es que es un sólido o de su coeficiente convectivo si es que es un fluido, la temperatura a la que está el fluido o si es que es un sólido con una temperatura que se considera constante. Además, se puede definir la longitud que representa un pixel para determinar la escala de figura o alternativamente si se conoce la longitud real en metros de alguno de los lados, horizontal o vertical. También se definen las condiciones en los bordes de la figura, ya sean paredes aisladas o paredes con un flujo de calor. También hay un área para hacer ligeras modificaciones a la imagen como cambiar la escala y reducir el número de colores únicos en la imagen por si se encuentran más elementos distintos de los que se perciben a simple vista, aunque se recomienda el tratamiento previo de la imagen en cualquier otro software especializado para edición de imágenes con mejores herramientas disponibles.



File Name: Resources/loadinglw_on.bmp

Dimensions: 5 X 6

Pixel length: 0.001

Image H_length: 0.006

Image V_length: 0.005

Image Size: 100%

Elements: 3

Nodes: 30

Table Fonts Color: Gray

	Pixel value	Type	Heat generation	k	h	T _{fluid}	T _{cte}
1	0 0 0	SOLID	0	25	0	0	---
2	153 217 234	FLUID	0	0	200	400	---
3	255 201 14	FLUID	0	0	1000	1700	---

Conditions at the borders

Border	Adiabatic	Heat Flux
Top	<input checked="" type="checkbox"/>	0
Right	<input checked="" type="checkbox"/>	0
Bottom	<input checked="" type="checkbox"/>	0
Left	<input checked="" type="checkbox"/>	0

IMAGEN 1: Interfaz gráfica del software desarrollado planteando un problema sencillo.

El software ofrece como resultados una representación de la imagen con el campo de temperaturas correspondientes, así como una visualización de la malla equivalente al problema y un archivo con las temperaturas y las coordenadas correspondientes tomando como origen la esquina inferior izquierda de la imagen.

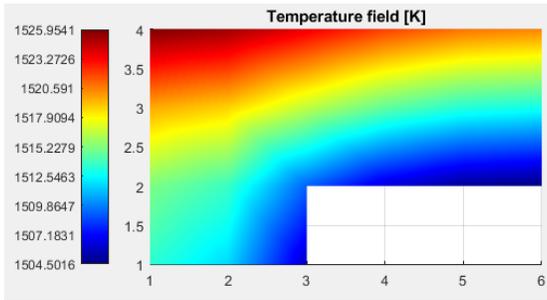


IMAGEN 2: Campo de temperatura obtenido para un problema simple.

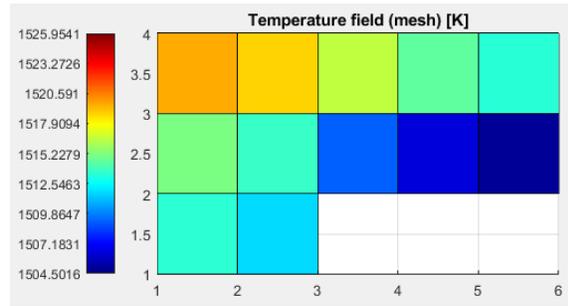


IMAGEN 3: Representación de la malla de un problema simple (los colores representan la temperatura promedio en la celda).

RESULTADOS Y DISCUSIÓN

El caso presentado a continuación es un ejemplo clásico de un problema de transferencia de calor. Una pared con flujo de energía de 3000 W/m^2 en un lado y aletas de distintas formas en contacto con un fluido de propiedades conocidas. Cada caso fue analizado tanto en ANSYS-FLUENT como en el software desarrollado para comparar los resultados.

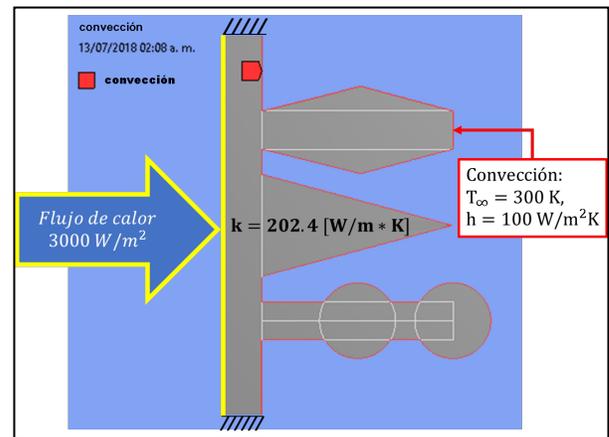


IMAGEN 4: Ejemplo de problema de geometría complicada usado en la validación de resultados.

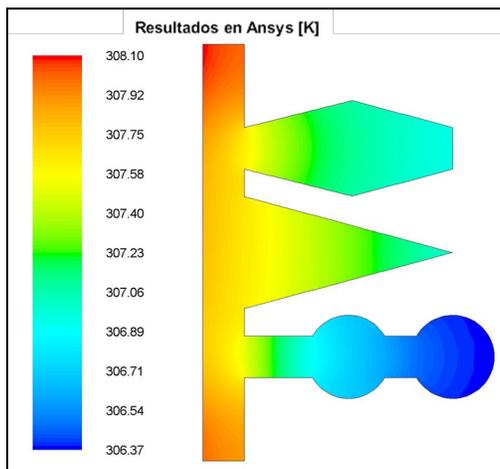


IMAGEN 5: Campo de temperatura obtenido en ANSYS-FLUENT.

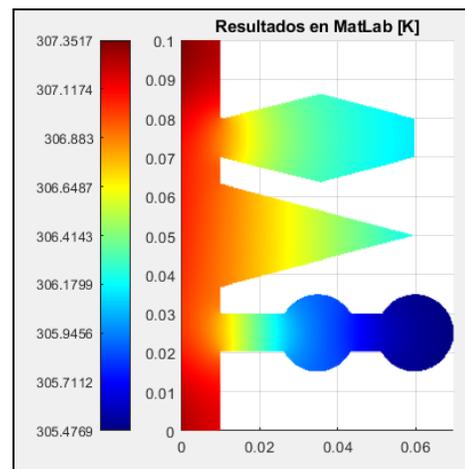


IMAGEN 6: Campo de temperatura obtenido en el software desarrollado.

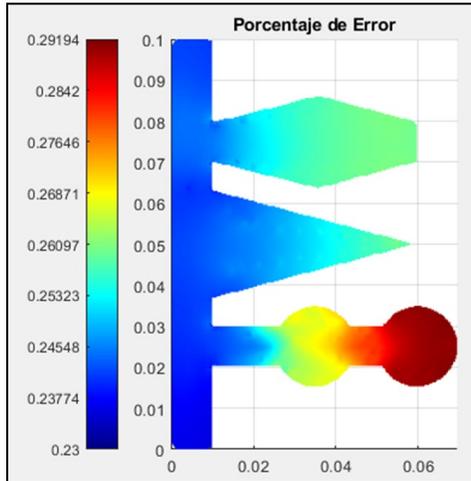


IMAGEN 7: Porcentaje del error calculado en base al resultado de ANSYS-FLUENT.

Para apreciar las diferencias de manera más exacta se exportaron ambos resultados y se calculó el porcentaje de la diferencia tomando como referencia el valor obtenido en ANSYS-FLUENT para cada nodo.

Se puede observar que el valor del error es más o menos consistente en la geometría con un mínimo de 0.23% y un máximo de 0.29%, siendo el valor más alto localizado en la zona curva de la geometría, lo que se puede explicar con el hecho de que se está usando una malla de elementos cuadrados para representar zonas curvas. Este error puede disminuir con mallas más finas en ambos programas. También es importante notar que la similitud exacta del problema no es posible de lograr en geometrías complicadas ya que no se comparten las mismas mallas debido a la naturaleza distinta de cada software, por lo que errores mucho menores son posibles de obtener si es que esta similitud se llegara a cumplir, por ejemplo, en geometrías que se puedan mallar completamente por elementos cuadrados como formas sin curvas ni inclinaciones.

CONCLUSIONES

El software fue desarrollado con la idea de crear una herramienta fácil y rápida de usar para estudiantes. Se puede concluir que los resultados son satisfactorios para poder apreciar los fenómenos de transferencia de calor y poner a prueba el entendimiento que los estudiantes tienen de dichos fenómenos y así aportar a su aprendizaje.

AGRADECIMIENTOS

El presente trabajo se llevó a cabo bajo la supervisión del Dr. Abel Hernández Guerrero, Dr. José Luis Luviano Ortiz y el M. en I. Fernando Cano Bando a quienes me gustaría agradecer por dedicar su tiempo y su ayuda en decidir la orientación del proyecto, así como en la discusión y análisis de los resultados.

REFERENCIAS

- [1] Bergman, T. L., Incropera, F. P., DeWitt, D. P., & Lavine, A. S. (2011). Fundamentals of heat and mass transfer (7th ed.) Hoboken, NJ: John Wiley & Sons.
- [2] Holman, J. P. (2002). Heat transfer (9th ed.) New York: McGraw-Hill.
- [3] Gilbert, J. R., Moler, C., & Schreiber, R. (1992). Sparse matrices in MATLAB: Design and implementation. SIAM Journal on Matrix Analysis and Applications, 13(1), 333-356.
- [4] Davis, T. (2007). Creating sparse finite-element matrices in MATLAB. Guest blog in: Loren Shure: Loren on the Art of MATLAB Google Scholar. Recuperado de <https://blogs.mathworks.com/loren/2007/03/01/creating-sparse-finite-element-matrices-in-matlab/>