

**APLICACIÓN DEL METODO NUMERICO DE RUNGE KUTTA PARA LA
DETERMINACION DEL DIAGRAMA INDICADOR DE UN MOTOR DE
COMBUSTION INTERNA ALTERNATIVO**

**EDWARD FABIAN OLARTE SILVA
FABIO ANDRES RIOS QUIJANO**



**UNIVERSIDAD PONTIFICIA BOLIVARIANA
ESCUELA DE INGENIERIA
FACULTAD DE INGENIERIA MECANICA
BUCARAMANGA**

2011

**APLICACIÓN DEL METODO NUMERICO DE RUNGE KUTTA PARA LA
DETERMINACION DEL DIAGRAMA INDICADOR DE UN MOTOR DE
COMBUSTION INTERNA ALTERNATIVO**

**EDWARD FABIAN OLARTE SILVA
FABIO ANDRES RIOS QUIJANO**

Trabajo de grado presentado para optar al título de Ingeniero Mecánico.

**Director:
JAVIER MAURICIO CASTELLANOS.
Ingeniero mecánico.**

**UNIVERSIDAD PONTIFICIA BOLIVARIANA
ESCUELA DE INGENIERIA
FACULTAD DE INGENIERIA MECANICA
BUCARAMANGA**

2011

Notas de aceptación

Firma del presidente del jurado

Firma del jurado

Firma del jurado

Bucaramanga, 4 de octubre de 2011

TABLA DE CONTENIDO

INTRODUCCION.

1. OBJETIVOS.	1
1.1. Objetivo general.	1
1.2. Objetivos específicos.	1
2. MARCO TEORICO.	3
2.1. Estudio de los motores de combustión interna.	3
2.1.1. Ciclo Real.	5
2.1.2. Ciclo teórico de gases.	6
2.1.2.1. Modelo de adición de calor a volumen constante (Ciclo de gas ideal).	6
2.1.2.2. Modelo de entrega Finita de Calor.	7
3.2. Métodos numéricos para la solución de ecuaciones diferenciales.	12
2.2.1. Método numérico de Runge-Kutta de cuarto orden.	12
2.3. Estudios precedentes.	13
3. DESARROLLO DE LA HERRAMIENTA COMPUTACIONAL.	16
4. RESULTADOS.	21
4.1. Aplicación de MATISSE 1.0 a problemas teóricos.	24
4.1.1. Variación del parámetro inicio de la combustión.	24
4.1.2. Variación del parámetro relación de compresión.	27
5. CONCLUSIONES Y RECOMENDACIONES.	30
6. BIBLIOGRAFÍA.	32

LISTA DE FIGURAS.

Figura 1: Ciclos del motor Otto cuatro tiempos.	4
Figura 2: Ciclo teórico de gas ideal.	7
Figura 3: Fracción de entrega de calor $x_b(\theta)$.	8
Figura 4: Comparación entre ciclo ideal de gas ideal y solución numérica.	11
Figura 5: Diagrama de flujo algoritmo MATISSE 1.0.	20
Figura 6: Comparación de curvas Presión vs Volumen.	21
Figura 7: Resultado obtenido por MATISSE 1.0.	23
Figura 8: Resultado obtenido por "Simple Heat Release Applet".	23
Figura 9: Diagrama Presión vs Volumen obtenido por MATISSE 1.0 para el caso de estudio 1.	25
Figura 10: Diagrama Presión vs Posición angular del cigüeñal obtenido por MATISSE 1.0 para el caso de estudio 1.	26
Figura 11: Diagrama Presión vs Volumen obtenido por MATISSE 1.0 para el caso de estudio 2.	28
Figura 12: Diagrama Presión vs Posición angular del cigüeñal obtenido por MATISSE 1.0 para el caso de estudio 2.	28

LISTA DE TABLAS.

Tabla 1: Comparación de Métodos Numéricos.	18
Tabla 2: Parámetros de operación de los motores evaluados por MATISSE 1.0 y "Simple Heat Release Applet".	22
Tabla 3: Parámetros de operación de los motores evaluados para el caso de estudio 1.	25
Tabla 4: Parámetros de operación de los motores evaluados para el caso. de estudio 1.	27

LISTA DE ANEXOS.

- Anexo a.** Manual de usuario.
- Anexo b.** Código computacional para el desarrollo de MATISSE 1.0 en Matlab.
- Anexo c.** Casos de estudio para la solución de ecuaciones diferenciales de primer orden mediante la aplicación del método numérico de Runge-Kutta de cuarto orden.

RESUMEN GENERAL

TITULO: APLICACIÓN DEL MÉTODO NUMÉRICO DE RUNGE-KUTTA PARA LA DETERMINACIÓN DEL DIAGRAMA INDICADOR DE UN MOTOR DE COMBUSTIÓN INTERNA ALTERNATIVO.

AUTORES: EDWARD FABIAN OLARTE SILVA.
FABIO ANDRÉS RÍOS QUIJANO.

FACULTAD: INGENIERÍA MECÁNICA.

DIRECTOR: JAVIER MAURICIO CASTELLANOS OLARTE

Este trabajo presenta el desarrollo de una herramienta computacional que simula el comportamiento del ciclo de gases de un motor de combustión interna mediante el uso del método numérico de Runge-Kutta de cuarto orden. Inicialmente se realizó una revisión bibliográfica de aplicaciones de software y temas relacionados con la solución numérica de los modelos de los ciclos de gases en motores de combustión interna; con base en esto se desarrollaron el modelo matemático, el algoritmo para la solución numérica de este modelo y la interfaz gráfica. Finalmente se obtuvieron los parámetros de operación y las curvas de desempeño que describen el comportamiento del motor de combustión interna alternativo.

PALABRAS CLAVES: MOTOR DE COMBUSTION INTERNA CICLO OTTO, MODELO DE ENTREGA FINITA DE ENERGÍA, METODO NUMERICO RUNGE-KUTTA DE CUARTO ORDEN, ALGORITMO.

GENERAL ABSTRACT

TITLE: USE OF THE RUNGE-KUTTA FOURTH ORDER NUMERICAL METHOD TO OBTAIN THE PV DIAGRAM OF AN INTERNAL COMBUSTION ENGINE.

AUTHORS: EDWARD FABIAN OLARTE SILVA.
FABIO ANDRÉS RÍOS QUIJANO.

FACULTY: MECHANICAL ENGINEERING.

DIRECTOR: JAVIER MAURICIO CASTELLANOS OLARTE.

This diploma work shows the development of a computational tool that simulates the gas cycle behavior on an internal combustion engine using a Runge-Kutta fourth order numerical method. Initially, a bibliographical review was conducted to identify the state of the art on software to numerically solve internal combustion engine gas cycles and related topics; based on that, a mathematical model that describes the system, an algorithm that numerically solves this model and a Graphic User Interface were developed. Finally, operation parameters were obtained along with performance graphics that describe the behavior of the engine.

KEYWORDS: INTERNAL COMBUSTION ENGINE OTTO CYCLE, SIMPLE HEAT RELEASE MODEL, NUMERICAL METHOD OF RUNGE-KUTTA DE FOURTH ORDER, ALGORITHM.

INTRODUCCION.

La mayoría de los adelantos en la ingeniería moderna se deben gracias al consecuente desarrollo y perfeccionamiento de los sistemas computacionales. Solo hasta que las computadoras y sus lenguajes permitieron la incorporación y solución de teorías y modelos matemáticos, se pudo remplazar los bancos experimentales por modelos de simulación informáticos, logrando así realizar los procesos de diseño y evaluación, en menos tiempo, de forma confiable y económico. De aquí la importancia de la implementación de estos sistemas, pues más que útiles, son necesarios en el que hacer común de la ingeniería moderna.

Este trabajo forma parte de un grupo de proyectos que pretenden abordar el estudio de la asignatura motores de combustión interna mediante sistemas computacionales. El desarrollo de la herramienta computacional MATISSE 1.0 tiene como objetivo proporcionar a los estudiantes de la asignatura un nuevo objeto de aprendizaje para fortalecer su proceso de formación académica y realizar las tareas de diseño y evaluación en menos tiempo y de forma confiable. De esta forma el estudiante comprenderá más rápidamente la influencia de ciertos parámetros sobre el comportamiento del motor.

El proyecto tiene como base la implementación del método numérico de Runge-Kutta de cuarto orden para la solución del sistema diferencial planteado que describe el ciclo de gases de un motor de combustión interna alternativo. Esto permite aproximar mejor el comportamiento real de los gases dentro del cilindro del motor al compararse con los resultados que se obtienen con los ciclos teóricos de gas ideal. Así mismo, para desarrollar este tipo de soluciones, la Universidad Pontificia Bolivariana Seccional Bucaramanga cuenta con el software especializado Matlab, cuyo uso en este proyecto servirá como punto de partida para futuros trabajos de simulación en el área de termofluidos.

1. OBJETIVOS

1.1. Objetivo General.

Desarrollar una herramienta computacional mediante el uso de métodos numéricos que modele el comportamiento del ciclo de gases de un motor de combustión interna, con el fin de proporcionar a los estudiantes de la asignatura de motores de combustión interna un nuevo objeto de aprendizaje.

1.2. Objetivos Específicos.

- Revisar el estado del arte en materia de modelos matemáticos aplicados a los motores de combustión interna alternativos y los métodos numéricos utilizados para su solución.
 - ✓ **RESULTADO:** Realización del Marco teórico.
 - ✓ **INDICADOR:** El marco teórico se fundamenta en información relevante obtenida en documentación científica como libros y artículos.

- Describir mediante ecuaciones diferenciales el ciclo de gases de un motor de combustión interna alternativo de cuatro tiempos.
 - ✓ **RESULTADO:** Sistema de ecuaciones diferenciales que describen el comportamiento de un motor de combustión interna alternativo.
 - ✓ **INDICADOR:** Las ecuaciones planteadas deben cumplir con las leyes físicas y el lenguaje matemático adecuado.

- Desarrollar el algoritmo para la solución numérica del sistema de ecuaciones diferenciales planteado, utilizando las herramientas de análisis, visualización y modelamiento que posee MATLAB.
 - ✓ **RESULTADO:** Desarrollo del algoritmo para la solución numérica en MATLAB.

- ✓ **INDICADOR:** El algoritmo debe obedecer el lenguaje de programación de MATLAB.
- Comparar los resultados obtenidos del algoritmo desarrollado con los resultados del ciclo teórico de gases ideales aplicados a las mismas condiciones.
 - ✓ **RESULTADO:** Diagrama Presión vs Volumen sobrepuesta entre los resultados obtenidos del algoritmo desarrollado y el ciclo teórico de gases ideales.
 - ✓ **INDICADOR:** Las curvas y parámetros de operación se evalúan de acuerdo con la literatura especializada de motores de combustión interna.

2. MARCO TERORICO

2.1. Estudio de los motores de combustión interna.

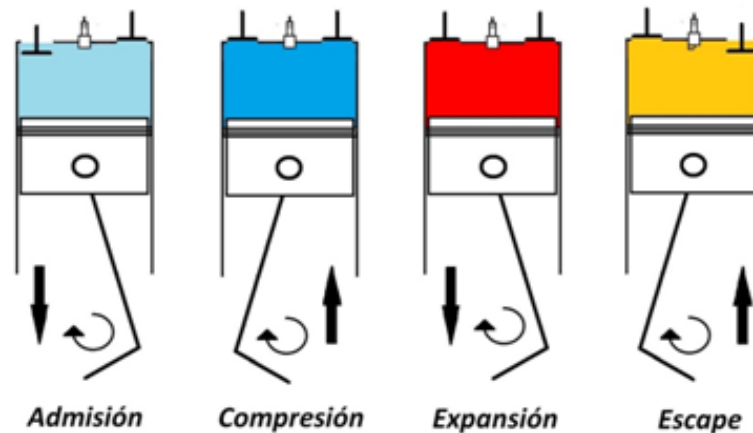
Un motor de combustión interna es definido como una máquina termodinámica cuyo propósito es transformar la energía química en energía mecánica, mediante la combustión producida por una mezcla de aire-combustible. Esta mezcla antes de la combustión y los gases quemados que resultan después de la combustión, son los fluidos de trabajo efectivo del sistema. La transferencia de trabajo que proporciona la potencia se produce directamente entre estos fluidos y los componentes mecánicos del motor.

Los motores de combustión interna que son el sujeto de estudio para este proyecto son los motores encendidos por chispa, denominados motor ciclo Otto en honor a su creador Nikolaus Otto (1832-1892) quien realizó importantes estudios sobre el motor de gas y en 1876 llevo a la práctica la construcción del primer motor de combustión interna de cuatro tiempos. De este modo creó una máquina motriz estática a partir de la cual desarrollaría el motor Otto. Éste se hizo famoso en todo el mundo como máquina para el accionamiento de vehículos, trenes, barcos y aviones, además este motor sirvió de base para la invención del Motor Ciclo Diésel. Otto es considerado el inventor de los motores de combustión interna modernos, y el fundador de la industria de motores de combustión interna. Debido a su simplicidad, robustez y alta potencia, este tipo de motores ha encontrado una amplia aplicación en el transporte y la generación de energía. [1, 2]

La figura 1 muestra la secuencia de operaciones que componen el ciclo del motor Otto. Inicialmente, en la carrera de admisión, la mezcla aire-combustible entra al cilindro a través de la válvula de admisión al tiempo en que el pistón se desplaza hasta el PMI (Punto muerto inferior). A continuación, en la carrera de compresión, las válvulas permanecen cerradas y el pistón se mueve hasta el PMS (punto muerto superior) comprimiendo la mezcla aire-combustible; cuando el pistón llega

al final de esta fase, la bujía se activa y enciende la mezcla. La siguiente etapa es la carrera de expansión o potencia, en donde se produce la combustión de la mezcla, liberando energía que provoca la expansión de los gases y el movimiento del pistón hacia el PMI. En esta etapa se produce la transformación de la energía química contenida en el combustible en energía mecánica transferida al pistón. Finalmente, en la carrera de escape, los gases quemados productos de la combustión, son expulsados a través de la válvula de escape por el pistón que se mueve hasta el PMS. [1, 2, 3]

Figura 1: Ciclos del motor Otto cuatro tiempos



Fuente: Los autores

El aire entra al motor a través del múltiple de admisión, y mediante un conjunto de conductos la mezcla de aire se distribuye a cada cilindro. El combustible, generalmente gasolina, se mezcla utilizando un inyector de combustible o carburador con la entrada de aire en el múltiple de admisión, antes de la válvula de admisión o directamente en el cilindro. Cuando la mezcla aire-combustible se enciende por la chispa, una llama turbulenta se desarrolla y se propaga a través de la mezcla, elevando la temperatura y la presión del cilindro. La llama se extiende cuando esta llega a las paredes del cilindro. Si la presión inicial es demasiado alta, los gases comprimidos alrededor de la llama automáticamente se

encienden, causando un fenómeno llamado golpeteo (knock). Este tipo de problemas limita la relación de compresión máxima de un motor ciclo Otto. Por último, los gases quemados salen del motor a través de la válvula de escape ubicada en cada cilindro, luego pasan por el múltiple de escape hasta salir al medio ambiente.

En el ciclo Otto, la válvula que controla la cantidad de entrada de aire se llama mariposa. Cuando esta es cerrada, el cantidad de aire que entra a cada cilindro disminuye, causando una reducción proporcional en la presión de cada cilindro. Puesto que el flujo de combustible se dosifica en proporción al flujo de aire, la válvula mariposa en el ciclo Otto, en esencia, es la que controla la potencia del motor. [1, 2, 3]

En la actualidad existen varias formas de abordar el estudio de motores de combustión interna, las más sobresalientes son la obtención experimental de parámetros de operación y el uso de ciclos teóricos que incorporan modelos matemáticos para obtener una aproximación del comportamiento real del motor.

2.1.1. Ciclo Real

El ciclo real se obtiene experimentalmente mediante la adquisición de datos transmitidos por diversos aparatos indicadores (sensor de presión, encoder, etc), capaces de reproducir el diagrama de presión de los gases dentro del cilindro del motor en función del volumen instantáneo del mismo. Este diagrama refleja las condiciones reales del comportamiento de los ciclos del motor, por lo cual, tiene en cuenta, las pérdidas de calor, la duración de combustión, las pérdidas causadas por el rozamiento del fluido, la duración del tiempo de apertura de las válvulas, el tiempo de encendido, así como de inyección y las pérdidas del escape.

La precisión que refleja el ciclo real en los diagramas P-V, se traduce en costos elevados pues se requiere de bancos de ensayos con la instrumentación y sistemas de adquisición de datos necesarios para la realización de las pruebas.

2.1.2. Ciclo teórico de gases

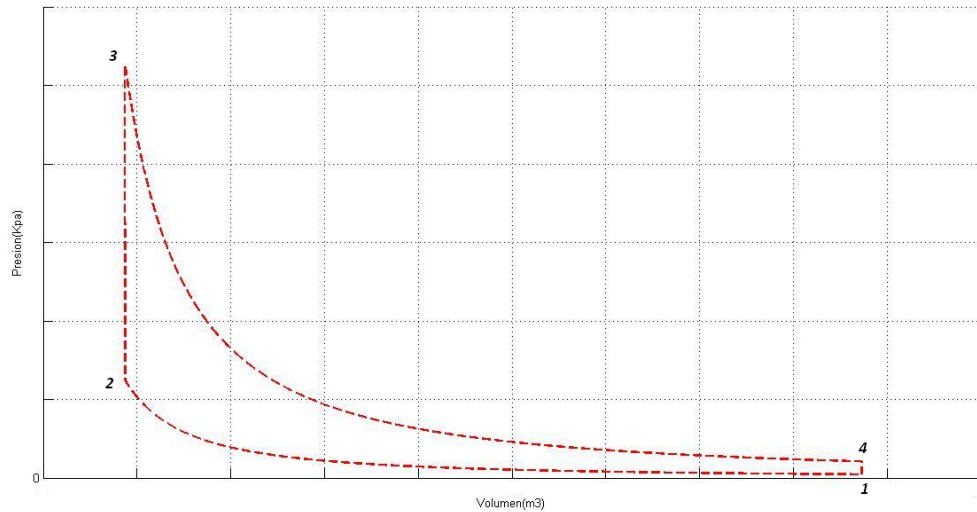
El estudio de los ciclos de gases como modelo de los motores de combustión interna es útil para ilustrar algunos de los parámetros más importantes que influyen en su rendimiento. Los cálculos del ciclo de gases simplifican el proceso de la combustión haciéndolo equivalente a una transferencia finita de calor. La combustión se modela como proceso a volumen constante y como proceso de entrega finita de calor. Este último análisis simplifica los detalles físicos y químicos de la combustión para facilitar la solución del modelo.

2.1.2.1. Modelo de adición de calor a volumen constante (Ciclo de gas ideal).

Se conoce generalmente como ciclo Otto de aire teórico. El fluido de trabajo está constituido por aire, y la combustión y el escape se modelan como adición y pérdida de calor respectivamente, en los otros procesos se supone que no se presentan pérdidas de calor. Aquí la combustión se supone tan rápida que el pistón permanece estático, por lo tanto se asume que ocurre a volumen constante. Teniendo en cuenta las suposiciones que se hacen para este ciclo ideal, se obtienen, para las temperaturas y presiones máximas, valores mayores, pero aproximados a los correspondientes a un ciclo real. [2]

La figura 2 muestra los cuatro procesos teóricos que componen este ciclo son. De 1 a 2 se tiene una compresión Isoentropica, el proceso de 2 a 3 se conoce como adición de calor a Volumen Constante, el proceso 3 a 4 es una expansión Isoentropica y por último, de 4 a 1 se tiene rechazo de calor a Volumen Constante.

Figura 2: Ciclo teórico de gas ideal



Fuente: Los autores

2.1.2.2. Modelo de entrega Finita de Calor

El modelo de entrega finita de calor se fundamenta en la implementación de un modelo diferencial que simula la adición de calor en el ciclo de potencia provocada por la combustión. Esto permite una mejor aproximación al ciclo real del motor que el modelo de adición de calor a volumen constante mencionado anteriormente. [2]

El modelo diferencial $x_b(\theta)$ denominado fracción de entrega de calor, permite ver la fracción de calor liberada en cualquier instante de la combustión y se representa de la siguiente forma. [2]

$$x_b(\theta) = 1 - \exp\left[-a \left(\frac{\theta - \theta_s}{\theta_d}\right)^n\right] \quad (1)$$

Donde:

$\theta =$ Posición angular del cigüeñal

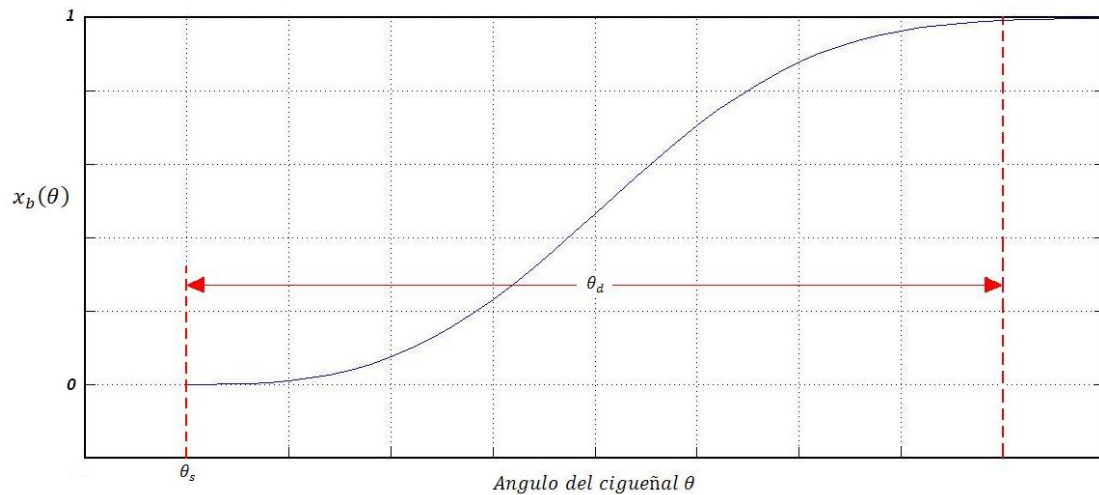
$\theta_s =$ Posición angular en la que comienza la emisión de calor

$\theta_d = \text{duracion emision de calor en grados de giro de cigueñal}$

$n = \text{factor de forma de Weibe}$

$a = \text{factor de eficiencia de Weibe}$

Figura 3: Fracción de entrega de calor $x_b(\theta)$



Fuente: Los autores

La figura 3, muestra la forma de la función de entrega de calor. El comienzo de la entrega de calor ocurre en $\theta = \theta_s$ en el cual $x_b(\theta) = 0$ y se aproxima asintóticamente a 1, para el final de la combustión, esta es definida por un límite arbitrario que puede ser 90% o 99% que corresponde a $x_b(\theta) = 90$ o $x_b(\theta) = 99$, esto depende del valor que se defina para el factor de eficiencia de Weibe que puede ser $a = 2.3$ o $a = 4.6$ respectivamente. De acuerdo con Heywood, valores de 5 y 3 para a y n respectivamente, permiten obtener resultados muy aproximados a la realidad. [2]

La tasa de entrega de calor como función del ángulo del cigüeñal se obtiene mediante la diferenciación de la entrega acumulada de calor (función de Weibe).

[2]

$$\frac{dQ}{d\theta} = Q_{in} \frac{dx_b}{d\theta} \quad (2)$$

$$\frac{dQ}{d\theta} = na \frac{Q_{in}}{\theta_d} (1 - x_b) \left(\frac{\theta - \theta_s}{\theta_d} \right)^{n-1} \quad (3)$$

La ecuación diferencial de conservación de la energía para un sistema cerrado está dada por:

$$\delta Q - \delta W = dU \quad (4)$$

Donde:

$$\delta W = PdV \quad (5)$$

$$dU = mC_v dT \quad (6)$$

Remplazando se obtiene:

$$\delta Q - PdV = mC_v dT \quad (7)$$

Representando el fluido de trabajo como gas ideal

$$PV = mRT \quad (8)$$

Derivando se tiene:

$$mdT = \frac{1}{R}(PdV + VdP) \quad (9)$$

Por lo tanto la ecuación de la energía queda descrita como:

$$\delta Q - PdV = \frac{Cv}{R}(PdV + VdP) \quad (10)$$

Por unidad del ángulo del cigüeñal se tiene entonces:

$$\frac{dQ}{d\theta} - P \frac{dV}{d\theta} = \frac{Cv}{R} \left(P \frac{dV}{d\theta} + V \frac{dP}{d\theta} \right) \quad (11)$$

Despejando $\frac{dP}{d\theta}$ se obtiene:

$$\frac{dP}{d\theta} = -\gamma \frac{P}{V} \frac{dV}{d\theta} + \frac{\gamma-1}{V} \left(\frac{dQ}{d\theta} \right) \quad (12)$$

Describiendo el volumen instantáneo en términos de la geometría del motor se obtiene:

$$V(\theta) = \frac{V_d}{r-1} + \frac{V_d}{2} [1 + \cos \theta (R^2 - \sin^2 \theta)^{-1/2}] \quad (13)$$

Derivando se tiene

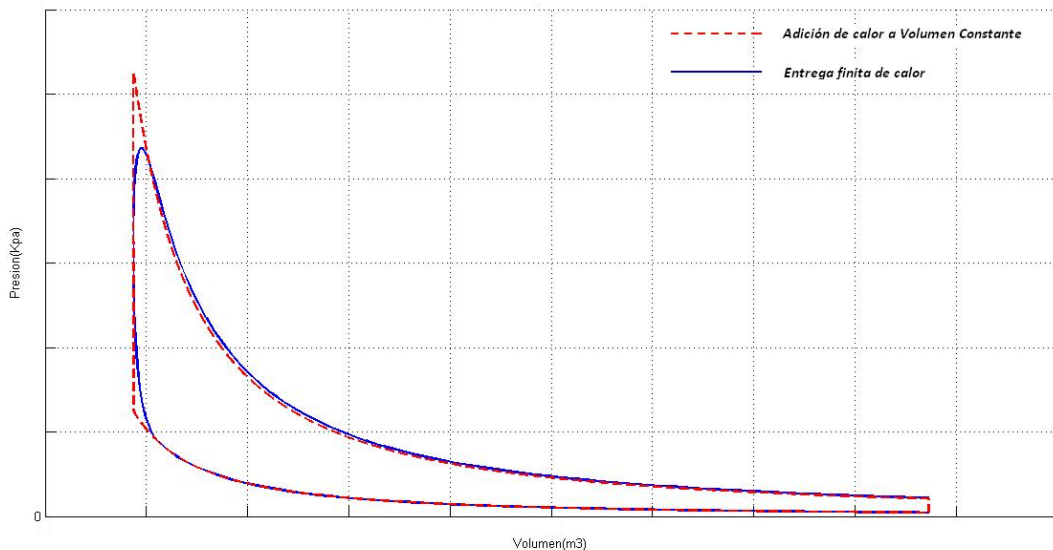
$$\frac{dV}{d\theta} = \frac{V_d}{2} \sin \theta [1 + \cos \theta (R^2 - \sin^2 \theta)^{-1/2}] \quad (14)$$

Las ecuaciones así obtenidas conforman el sistema diferencial de entrega finita de calor que permite describir el ciclo de gases de un motor de combustión interna.

Este modelo es el objeto de estudio para el desarrollo del presente trabajo, pues gracias a él, se aportan características del comportamiento real del motor que los modelos teóricos de gas ideal no podrían incorporar. Un ejemplo de esto es la influencia que tiene el encendido de la chispa o la transferencia de calor sobre el trabajo y la eficiencia del motor de combustión interna. Para la obtención de las cantidades de interés se recurre a técnicas numéricas de solución de ecuaciones diferenciales con condiciones iniciales o de frontera. [2]

La figura 4 muestra las diferencias entre el ciclo teórico de gas ideal y la solución numérica del sistema de ecuaciones que incorpora el modelo de entrega finita de energía. Al observar la solución numérica se encuentra una mayor semejanza con los resultados experimentales de los ciclos reales.

Figura 4: Comparación entre ciclo ideal de gas ideal y solución numérica.



Fuente: Los autores

2.2. Métodos numéricos para la solución de ecuaciones diferenciales ordinarias.

Muchos de los fenómenos que ocurren a nuestro alrededor se describen mediante ecuaciones diferenciales ordinarias. En ingeniería por ejemplo, procesos estáticos y dinámicos como deflexión de vigas, movimientos amortiguados, vaciado de recipientes, plantas térmicas e incluso los ciclos de gases de un motor de combustión interna alternativo, son descritos mediante este tipo de ecuaciones.

Los métodos numéricos juegan un rol importante para la descripción de estos fenómenos pues es bien sabido, que la mayoría de las ecuaciones diferenciales no tienen una solución exacta.

Debido a las características que presenta el sistema de ecuaciones diferenciales que describen el ciclo de gases de un motor de combustión interna, se emplean las técnicas numéricas de solución de ecuaciones diferenciales con condiciones iniciales o de frontera, denominados problemas de valor inicial. Para este tipo de problemas se desarrollaron los métodos de Euler, Taylor y Runge-Kutta de cuarto orden, siendo este último, el más apropiado para el desarrollo de aplicaciones computacionales por sus ventajas frente a los otros métodos. [4]

2.2.1. Método numérico de Runge-Kutta de cuarto orden

Es un método iterativo que permite obtener aproximaciones de soluciones de ecuaciones diferenciales ordinarias mediante un proceso de ponderación de pendientes.

El método numérico de Runge-Kutta de cuarto orden esta dado por la ecuación:

$$y_{i+1} = y_i + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \quad (15)$$

Donde:

$$k_1 = f(x_i, y_i) \quad (16)$$

$$k_2 = f\left(x_i + \frac{h}{2}, y_i + \frac{hk_1}{2}\right) \quad (17)$$

$$k_3 = f\left(x_i + \frac{h}{2}, y_i + \frac{hk_2}{2}\right) \quad (18)$$

$$k_4 = f(x_i + h, y_i + hk_3) \quad (19)$$

Este método permite evaluar el punto y_{i+1} conociendo el valor de y_i más el producto del tamaño del intervalo h por una ponderación de pendientes k_1, k_2, k_3, k_4 con pesos de 1, 2, 2, 1, respectivamente, dando lugar a una recta de pendiente $(k_1 + 2k_2 + 2k_3 + k_4)/6$. [4]

2.3. Estudios precedentes

La consulta del estado del arte sobre el estudio de motores de combustión interna mediante modelos matemáticos, arroja información relevante particularmente en documentación científica como libros y artículos.

Científicos e investigadores han observado que mediante modelos matemáticos se pueden conocer parámetros del comportamiento del motor que los ciclos tradicionales teóricos de gas ideal no pueden mostrar, por ende cada vez son más los modelos desarrollados para acercar sus resultados a la realidad.

Norberto Nigro, Mario A. Storti y Leonardo Ambroggi, (1999), en su trabajo pretenden abordar el estudio de motores de combustión interna mediante el uso de sistemas computacionales. Su trabajo desarrolla un código computacional para

la resolución de problemas de dinámica de gases a través de ductos y toberas; con el fin de generar un software capaz de simular los ciclos de combustión y de expansión. El modelo matemático está gobernado por las ecuaciones de Euler, flujo invíscido, con la inclusión de efectos como la transferencia de calor a través de las paredes del ducto, la variación de la sección transversal y la fricción en las paredes del mismo. Para la solución de dicho modelo se utilizaron técnicas numéricas como discretización espacial unidimensional en elementos finitos del tipo Petrov-Galerkiri con una discretización temporal según un esquema de Lax-Wendroff de dos pasos. [5]

Araque Maldonado, J. O., Fygueroa Salgado, S. J. y Martín Valera, M. J.,(2007), presentan dentro de su trabajo el diseño de una herramienta computacional, que permite evaluar algunas características del proceso de combustión de un motor a gasolina como los perfiles de masa y velocidad de quemado en función de la riqueza de la mezcla utilizando datos de presión experimental. Para su desarrollo se utilizaron las ecuaciones básicas de primera Ley de la Termodinámica, continuidad y gases ideales. Igualmente, emplean rutinas para el cálculo de la composición y propiedades de la mezcla de trabajo, la ecuación de Annand como modelo para la transferencia de calor, y relaciones matemáticas entre los parámetros físicos como, riqueza, densidad, presión, temperatura y fracción de gases residuales. [6]

Sundeeep Ramachandran, (2009), presenta un modelo termodinámico para la simulación de un motor encendido por chispa que funciona con un combustible alternativo. Su objetivo principal es desarrollar un modelo de simulación simple, rápido y preciso, sin necesidad de cálculos matemáticos complejos ni el conocimiento preciso de datos geométricos del motor. Se basa en el modelo Zero-dimensional donde la transferencia de calor en el cilindro, la energía y tasa de liberación de energía también se consideran. Las suposiciones más importantes se relacionan con el fluido de trabajo (se considera una mezcla de 14 especies,

O₂, N₂, CO₂, H₂O, H₂, OH, NO, CO, O, H, N, Ar, NO₂, HO₂); las 14 especies se consideran como gases ideales y los combustibles alternativos se limitan a las especies C-H-O-N. También supone que en cualquier instante de tiempo durante la combustión, el volumen del cilindro se divide en dos zonas, la zona que consta de gas sin quemar (mezcla aire-combustible) y la que consta de gas quemado (mezcla de 14 especies). Los gases quemados se asumen en equilibrio químico durante la combustión y para la carrera de expansión, mientras que al final de la carrera de expansión la mezcla se asume en composición fría. Las ecuaciones que conforma la base de este modelo son las relaciones de conservación de energía y masa y las ecuaciones de estado. Los datos finalmente calculados se utilizan para graficar los parámetros como la presión, temperatura y trabajo, todos ellos en función de la variable independiente que es el ángulo del cigüeñal. [7]

Ismet Sezer y Atila Bilgin, (2010) desarrollaron de una herramienta informática para un motor de combustión interna de encendido por chispa, a partir del análisis de la exergía en el ciclo de potencia del motor, con lo que se evalúa la cantidad de energía química transformada en energía mecánica. La herramienta computacional tiene como base un modelo matemático que simula los ciclos de compresión, combustión y expansión para los cuales se utilizaron las ecuaciones de Ferguson, principios de la primera y segunda ley de la termodinámica. Igualmente, la combustión se considera un proceso de propagación de llama turbulenta en la se hace uso de las ecuaciones de Blizard & Keck (1974), en la carrera de admisión y compresión se trabaja con un método simple de aproximación Bayraktar & Durgun. La solución al modelo matemático se hizo a través de la subrutina DVERK la cual hace uso de las formulas de Runge-Kutta de quinto y sexto orden para la solución de ecuaciones diferenciales de primer orden. [8]

Una herramienta importante que debe destacarse fue desarrollada por el departamento de Ingeniería Mecánica de la Universidad de Colorado de los

Estados Unidos. Se trata de una aplicación en Java llamada "Simple Heat Release Applet", capaz de calcular el rendimiento del motor y comparar los efectos de cambios en la combustión y en los parámetros geométricos. El código computacional se desarrolló a partir de un modelo matemático soportado en las ecuaciones de conservación de la energía, de estado de gas ideal y de la fracción de entrega de calor que hace parte del modelo de entrega finita de calor. El applet calcula el comportamiento del ciclo de gases mediante la integración numérica de la ecuación (12). [2, 10]

Esta ecuación diferencial de primer orden de la forma $dP/d\theta = f(\theta, P, Q)$ describe la presión dentro cilindro en función de la posición angular del cigüeñal (θ), presión de los gases dentro del cilindro (P) y de la entrega de calor (Q). Para su evaluación se emplean técnicas numéricas de solución de ecuaciones diferenciales con condiciones iniciales o de frontera, específicamente el método numérico de Runge-Kutta de cuarto orden. [2, 10]

3. DESARROLLO DE LA HERRAMIENTA COMPUTACIONAL.

A continuación se describe la metodología y los aspectos más relevantes que se llevaron a cabo para el desarrollo de la herramienta computacional MATISSE 1.0, junto con una presentación general de esta.

MATISSE 1,0 es una herramienta computacional diseñada y construida con el objetivo de proporcionar a los estudiantes de la asignatura motores de combustión interna un nuevo objeto de aprendizaje para fortalecer su proceso de formación académica y realizar las tareas de diseño y evaluación de motores en menos tiempo y de forma confiable. Está construida sobre la base de un algoritmo desarrollado en Matlab que efectúa la solución numérica del sistema de ecuaciones diferenciales planteado para describir el ciclo de gases de un motor de

combustión interna. Para esto se empleo el método numérico de Runge-Kutta de cuarto orden que permite una solución bastante aproximada al sistema diferencial mencionado anteriormente.

Usando MATISSE 1.0, se obtienen los diagramas de Presión vs Angulo del cigüeñal y Presión vs Volumen de motores de combustión interna alternativo. La comparación mediante graficas sobrepuestas de los ciclos de los motores estudiados permite al usuario comprender y analizar mejor la influencia de los parámetros del motor sobre el comportamiento del ciclo de gases. Para esto el usuario debe conocer la geometría del motor (radio de compresión, diámetro del pistón, distancia de carrera, etc) y las características descriptivas del proceso de combustión (temperatura inicial, presión inicial, duración de la combustión, parámetros de Weibe, etc).

La herramienta fue validada mediante la comparación directa de sus resultados con los obtenidos en el ciclo teórico de gas ideal para motores de combustión interna y posteriormente, comparado con los resultados obtenidos a partir de la aplicación desarrollada por el Departamento de ingeniería mecánica de la universidad de Colorado.

La revisión de la literatura permitió establecer el modelo de interés a partir del sistema de ecuaciones diferenciales que describen el fenómeno físico. Por otra parte los estudios precedentes directamente relacionados con el tema, permitieron identificar modelos similares y las herramientas de solución por métodos numéricos que fueron utilizadas.

La selección del método numérico entonces se soportó en la evaluación de 3 características principales, a) simplicidad al momento de su aplicación, b) proporcione excelente aproximación y d) no requiera de cálculos complejos. La

tabla 1 muestra una matriz de ventajas y desventajas cuya evaluación determinó la elección del método de Runge-Kutta de cuarto orden.

Tabla 1: Comparación de Métodos Numéricos

Método Numérico	Ventaja	Desventaja
Euler	El más simple de todos los métodos numéricos para resolver problemas de valor inicial.	No proporciona una buena aproximación.
Taylor	Proporciona una buena aproximación	Menos practico de los tres métodos pues incluye derivadas largas y complicadas de solucionar.
Runge-Kutta de cuarto orden	<ul style="list-style-type: none"> • Proporciona una buena aproximación. • No requiere de derivaciones complicadas de funciones. 	Mayor numero de cálculos matemáticos.

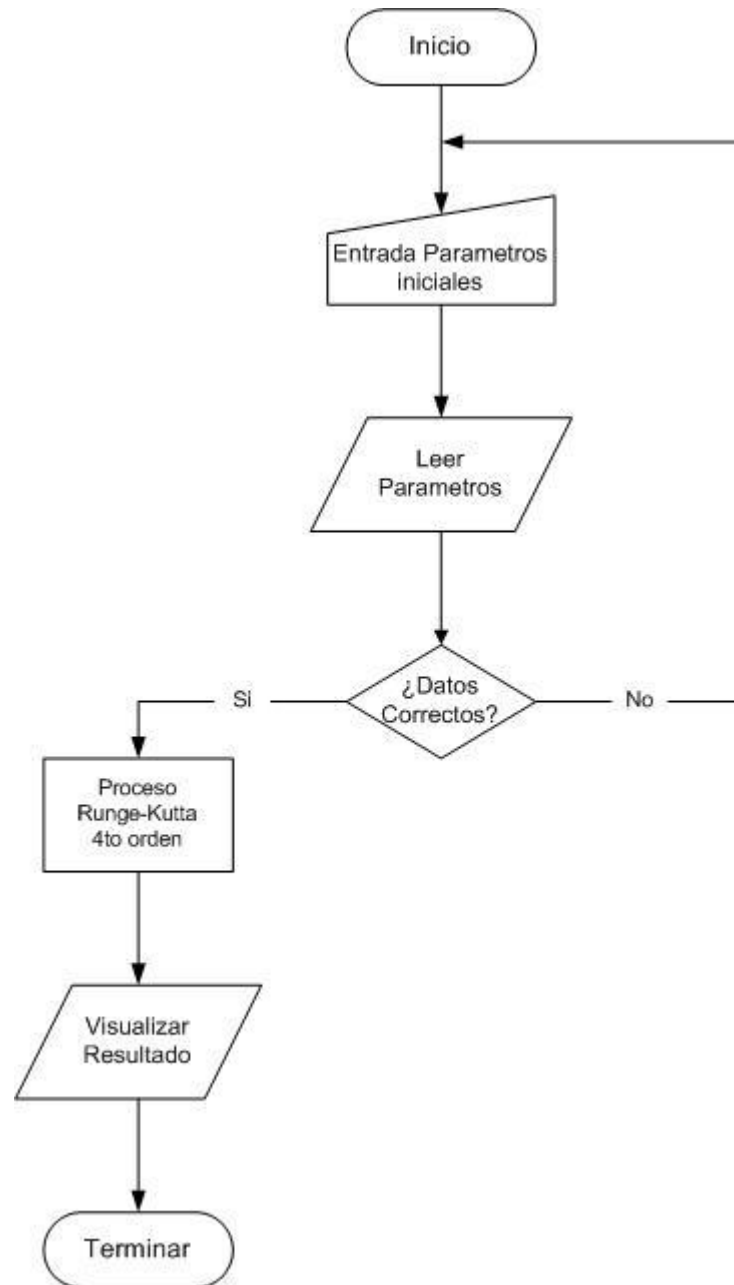
Fuente: Los autores

Una vez que el método fue estudiado y posteriormente dominado, se diseñó y desarrolló el algoritmo para la aplicación y creación del objeto de aprendizaje. Su implementación en Matlab obedeció a que este software cumple con un número elevado de características que se ajustan a las necesidades del proyecto. Por una parte es un lenguaje de alto nivel para la computación técnica, permite desarrollar algoritmos y resolver problemas más rápido que con los lenguajes de programación tradicionales, tales como C, C++ y Fortran; presenta un entorno de desarrollo para la gestión de códigos, archivos y datos; posee funciones practicas para realizar operaciones matemáticas y análisis de datos; posee herramientas para la creación de interfaz graficas (GUI) con imágenes de alta calidad fáciles de utilizar, logrando así una mejor interacción humano-maquina; presenta las

propiedades graficas necesarias para la correcta visualización de datos en 2D; posee herramientas de ayuda que apoyan y fortalecen el proceso de desarrollo del proyecto; por otro lado la universidad cuenta con este software y cumple con los requisitos legales para hacer uso de ella. [9]

Con base en los fundamentos de la técnica de Runge-Kutta de cuarto orden y en un estudio minucioso de un código computacional desarrollado por el Departamento de ingeniería mecánica de la universidad del sur de la florida [11], se diseño y escribió el algoritmo en Matlab que permite solucionar el sistema de ecuaciones diferenciales del modelo físico. La figura 6, muestra el diagrama de flujo (de acuerdo con la norma ISO 5807) del programa básico de la herramienta computacional.

Figura 5: Diagrama de flujo algoritmo MATISSE 1.0.



Fuente: Los autores

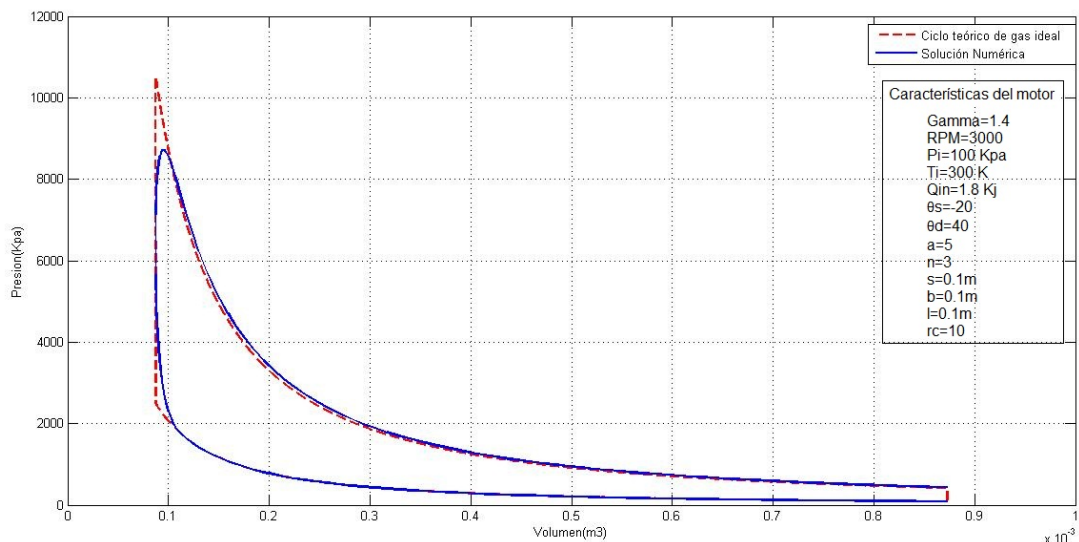
Con todos los elementos anteriores y con el código escrito en Matlab se construyó la interfaz grafica (GUI) que permite al usuario utilizar el objeto de aprendizaje, desde el ingreso de los datos de los parámetros requeridos, hasta la manipulación de las graficas finales con los resultados de evaluación.

Finalmente se elaboró el manual de usuario con las instalaciones necesarias para que la herramienta sea utilizada.

4. RESULTADOS.

Los diagramas de Presión vs Volumen y Presión vs Posición angular del cigüeñal obtenidos mediante MATISSE 1.0 y que describen el ciclo de gases del motor de combustión interna, son validados mediante comparación directa con los ciclos teóricos de gas ideal y con los diagramas generados por la herramienta computacional “Simple Heat Release Applet” [10].

Figura 6: Comparación de curvas Presión vs Volumen.



Fuente: Los autores

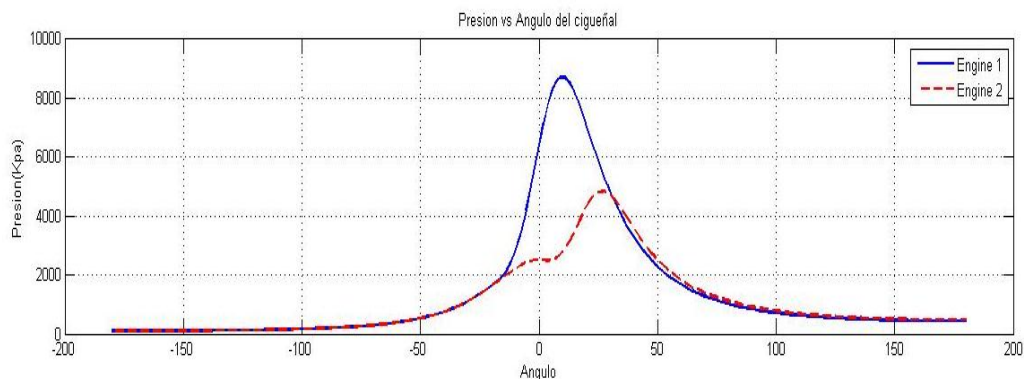
La figura 6 presenta los diagramas de Presión vs Volumen para el cilindro de un motor obtenidos mediante la herramienta computacional MATISSE 1.0 y el ciclo teórico de gas ideal a las mismas condiciones de trabajo. La línea azul que representa la solución numérica deja ver como el modelo diferencial $x_b(\theta)$ denominado fracción de entrega de calor, simula la adición de calor producida por la combustión de manera más aproximada a la realidad que con el ciclo de gas ideal que se asume a volumen constante. Según revisión bibliográfica realizada en el ciclo teórico de gas ideal, se obtienen valores de presión y áreas bajo la curva Presión vs Volumen mayores que los correspondientes al ciclo real, lo que permite evidenciar que la solución numérica del modelo de entrega finita de calor se aproxima mejor a la realidad que el ciclo de gas ideal.

Tabla 2: Parámetros de operación de los motores evaluados por MATISSE 1.0 y "Simple Heat Release Applet"

PARÁMETROS DE OPERACIÓN	MOTOR 1	MOTOR 2
Inicio de la combustión	-20	0
Duración de la combustión	40	40
Parámetro de Weibe a	5	5
Parámetro de Weibe n	3	3
Temperatura inicial [C]	27	27
Presión inicial [Kpa]	100	100
Calor de entrada [Kj]	1.8	1.8
Gamma	1.4	1.4
Carrera [m]	0.1	0.1
Diámetro del pistón [m]	0.1	0.1
Longitud de la biela [m]	0.15	0.15
Relación de compresión	10	10

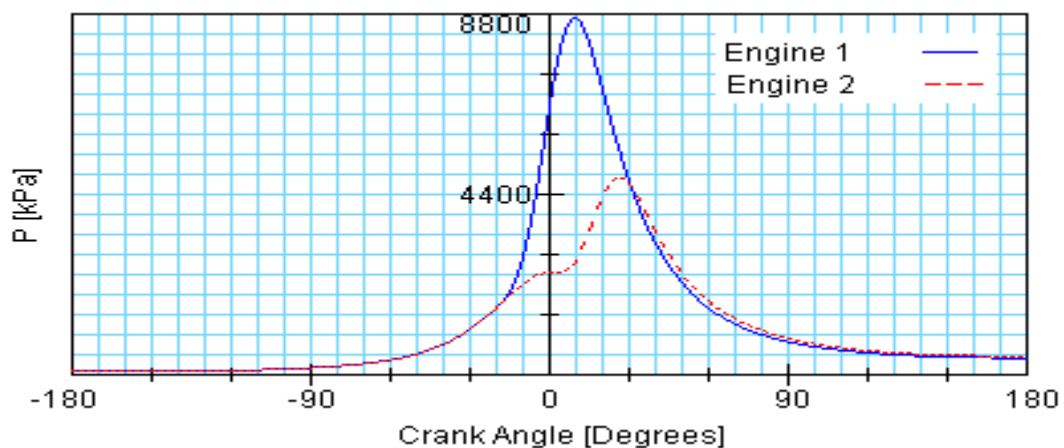
Fuente: Los autores

Figura 7: Resultado obtenido por MATISSE 1.0



Fuente: Los autores

Figura 8: Resultado obtenido por "Simple Heat Release Applet"



Fuente: Colorado State University

Las figuras 7 y 8 muestran los diagramas Presión vs Posición angular del cigueñal para dos motores de combustión interna ciclo Otto operando bajo las mismas condiciones de trabajo (ver tablas) y evaluados con dos herramientas computacionales diferentes. La figura 7 presenta el resultado obtenido por la herramienta computacional MATISSE 1.0 y la figura 8 presenta el resultado software "Simple Heat Release Applet". Las dos herramientas tienen como base el

modelo de entrega finita de calor solucionado mediante el método numérico de Runge-Kutta de cuarto orden. Los diagramas Presión vs Posición angular del cigüeñal obtenidos por las dos herramientas computacionales son prácticamente iguales.

4.1. Aplicación de MATISSE 1.0 a problemas teóricos.

Mediante el uso de la herramienta computacional MATISSE 1.0, el usuario comprenderá y analizará mejor la influencia de los parámetros del motor sobre el comportamiento del ciclo de gases. A continuación, se presentan dos casos de estudio en los que se evalúan los diagramas Presión vs Volumen y Presión vs Posición angular del cigüeñal de un motor de combustión interna para el que varía uno de sus parámetros de operación, con el objeto de ver la influencia que tiene este sobre el comportamiento del ciclo de gases.

4.1.1. Variación inicio de la combustión.

Para el primer caso de estudio se evalúa un motor de combustión interna alternativo con las mismas condiciones de trabajo exceptuando el parámetro de inicio de la combustión, mediante la herramienta computacional MATISSE 1.0. Analizando los diagramas obtenidos se espera conocer la influencia que tiene este parámetro sobre el comportamiento del ciclo de gases dentro del cilindro del motor.

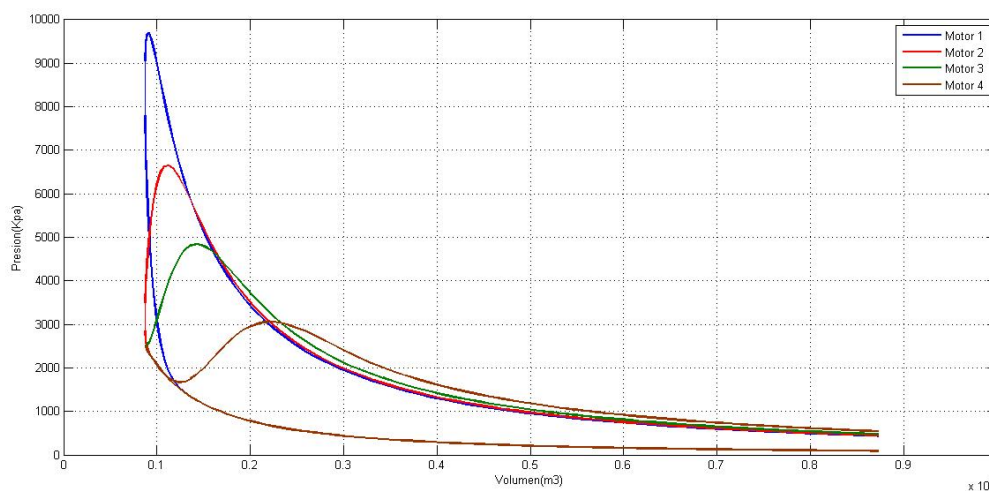
Tabla 3: Parámetros de operación de los motores evaluados para el caso de estudio 1.

PARÁMETROS DE OPERACIÓN	MOTOR 1	MOTOR 2	MOTOR 3	MOTOR 4
Inicio de la combustión	-25	-10	0	15
Duración de la combustión	10	10	10	10
Parámetro de Weibe a	5	5	5	5
Parámetro de Weibe n	3	3	3	3
Temperatura inicial [C]	20	20	20	20
Presión inicial [Kpa]	100	100	100	100
Calor de entrada [Kj]	1.8	1.8	1.8	1.8
Gamma	1.4	1.4	1.4	1.4
Carrera [m]	0.1	0.1	0.1	0.1
Diámetro del pistón [m]	0.1	0.1	0.1	0.1
Longitud de la biela [m]	0.15	0.15	0.15	0.15
Relación de compresión	10	10	10	10

Fuente: Los autores

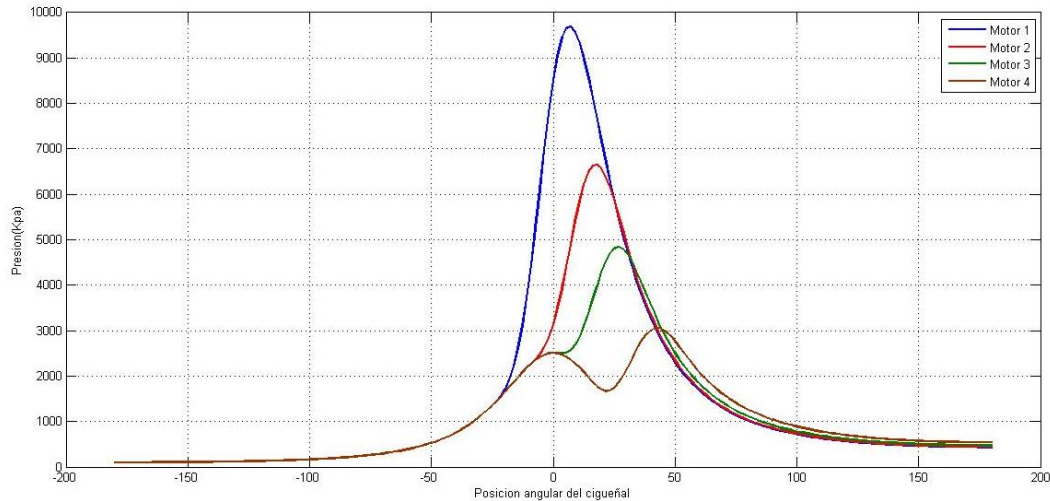
La tabla 3 presenta las condiciones generales de estudio, para las que se seleccionaron tiempo de encendido de -25° , -10° , 0° y 15° respectivamente.

Figura 9: Diagrama Presión vs Volumen obtenido por MATISSE 1.0 para el caso de estudio 1.



Fuente: Los autores

Figura 10: Diagrama Presión vs Posición angular del cigüeñal obtenido por MATISSE 1.0 para el caso de estudio 1.



Fuente: Los autores

Las figuras 9 y 10 muestran los resultados obtenidos con MATISSE 1.0. La presión máxima de las dos graficas se obtiene para el motor 1 con encendido a -25° . Normalmente los motores de encendido por chispa presenta esta configuración en su operación normal. Las curvas de los motores 2 y 3 presentan presión sustancialmente menores como consecuencia del retardo del encendido con respecto a las condiciones normales. Está bien documentado que si el inicio de la combustión se da muy cerca del punto muerto superior, la cantidad de mezcla quemada cuando se alcanza dicho punto será muy pequeña y la mayor parte de la combustión se dará cuando el pistón se aleja hacia el punto muerto inferior. Esta expansión finalmente limita la presión máxima dentro del cilindro. El motor 4 muestra el caso extremo; el primer máximo corresponde a la compresión originada únicamente por el cilindro y el segundo a la combustión tardía de la mezcla.

En cualquier caso el área bajo la curva del diagrama Presión vs Volumen y consecuentemente el trabajo indicado obtenido es mayor cuando la presión del cilindro es máxima.

4.1.2. Variación de relación de compresión.

Para el siguiente caso de estudio se analiza la influencia que tiene la relación de compresión sobre el comportamiento del ciclo de gases de un motor de combustión interna.

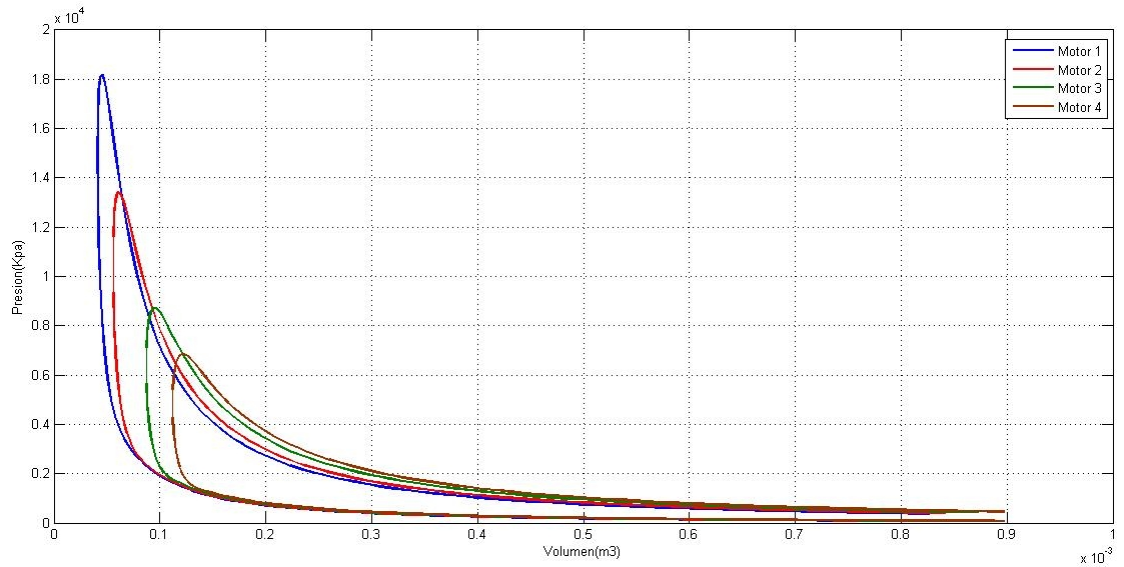
Tabla 4: Parámetros de operación de los motores evaluados para el caso de estudio 1.

PARÁMETROS DE OPERACIÓN	MOTOR 1	MOTOR 2	MOTOR 3	MOTOR 4
Inicio de la combustión	-20	-20	20	20
Duración de la combustión	10	10	10	10
Parámetro de Weibe a	5	5	5	5
Parámetro de Weibe n	3	3	3	3
Temperatura inicial [C]	20	20	20	20
Presión inicial [Kpa]	100	100	100	100
Calor de entrada [Kj]	1.8	1.8	1.8	1.8
Gamma	1.4	1.4	1.4	1.4
Carrera [m]	0.1	0.1	0.1	0.1
Diámetro del pistón [m]	0.1	0.1	0.1	0.1
Longitud de la biela [m]	0.15	0.15	0.15	0.15
Relación de compresión	20	15	10	8

Fuente: Los autores

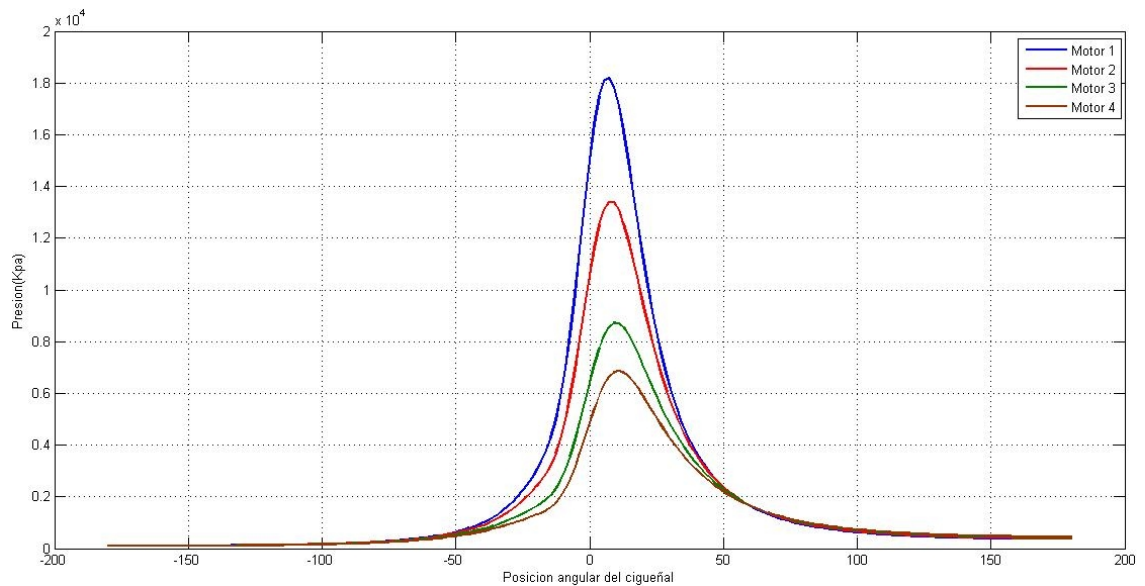
La tabla 4 presenta las condiciones generales de estudio. En este caso se seleccionan cuatro relaciones de compresión diferentes y se mantuvieron los demás parámetros constantes.

Figura 11: Diagrama Presión vs Volumen obtenido por MATISSE 1.0 para el caso de estudio 2.



Fuente: Los autores

Figura 12: Diagrama Presión vs Posición angular del cigüeñal obtenido por MATISSE 1.0 para el caso de estudio 2.



Fuente: Los autores

La relación de compresión puede definirse como un valor adimensional que permite medir la proporción en que se ha comprimido la mezcla aire-combustible. Tanto la figura 11 como la figura 12 presentadas anteriormente para este caso de estudio muestran una tendencia en la disminución de la presión máxima dentro del cilindro para cada motor al disminuir el valor de la relación de compresión. Como consecuencia, el área bajo la curva traducida en trabajo realizado por el motor disminuye. Como se sabe, al aumentar la presión dentro del cilindro aumenta el área bajo la curva lo que permite aumentar la eficiencia térmica al incrementar el valor de la relación de compresión.

5. CONCLUSIONES Y RECOMENDACIONES.

La revisión bibliográfica y la información recopilada en el estado del arte permitieron el desarrollo de la herramienta computacional MATISSE 1.0, diseñada para proporcionar a los estudiantes de la asignatura motores de combustión interna un nuevo objeto de aprendizaje. Mediante su implementación se obtienen los diagramas de Presión vs Volumen y Presión vs Posición angular del cigüeñal apoyados en las herramientas para generación de graficas en 2 dimensiones ofrecidas por Matlab.

El sistema de ecuaciones diferenciales planteado para el ciclo de gases de un motor de combustión interna alternativo se soportó en las ecuaciones de gas ideal, conservación de la energía y en el modelo diferencial de fracción de entrega de calor que simula la adición de energía en el ciclo de potencia. Este modelo dio como resultado una mejor aproximación al comportamiento real de los gases dentro del cilindro en comparación al de los ciclos teóricos de gas ideal.

El desarrollo del algoritmo en Matlab para la solución numérica del sistema diferencial que describe el ciclo de gases del motor de combustión interna alternativo resultó ventajoso. De una parte, constituye una herramienta que facilita la elaboración de los diagramas que describen el comportamiento del motor y la solución del sistema de ecuaciones de las cuales se derivan. Por otra, facilita el trabajo del programador al presentar herramientas para la depuración de los algoritmos y para la elaboración de las interfaces de usuario.

La comparación de los diagramas de Presión vs Volumen del ciclo teórico de gas ideal y de la solución numérica arrojada por MATISSE 1.0, permite evidenciar las mismas diferencias que se obtienen normalmente con los ciclos reales de los motores de combustión interna alternativos. De igual manera la variación de los parámetros operativos de un motor y su efecto en los diagramas de Presión vs Posición Angular del Cigüeñal y Presión vs Volumen dentro del cilindro,

corresponden a los comportamientos que se presentan en los motores reales sometidos a las mismas condiciones de operación y que son reportados normalmente en la literatura.

Mediante el uso de MATISSE 1.0 se pueden evaluar un sinnúmero de configuraciones de motores de combustión interna alternativos y analizar los resultados mediante comparación directa de los diagramas Presión vs Volumen y Presión vs Posición angular del cigüeñal. Esta característica permite al usuario, y particularmente a los estudiantes de la asignatura motores de combustión interna, comprender mejor la influencia de parámetros operativos sobre el comportamiento del ciclo de gases, y en consecuencia, sobre el motor en sí mismo.

La revisión bibliográfica de los trabajos realizados en relación con la solución numérica de los modelos aplicados a los ciclos de los motores de combustión interna muestra que el trabajo realizado aquí puede mejorarse al incluir modelos de combustión más complejos, así como modelos para los procesos de admisión y escape. MATISSE 1.0 fue diseñado y desarrollado como una herramienta computacional abierta que permitirá la incorporación de estos nuevos modelos.

6. BIBLIOGRAFIA

- [1] WILLARD W. PULKRABEK. Engineering fundamentals of the internal combustion Engine, segunda edición. Pearson.
- [2] COLIN R. FERGUSON and ALLAN T. KIRKPATRICK. Internal combustion engines, segunda edición. John wiley & Sons, 2001, p. 1-45.
- [3] JOHN B. HEYWOOD. Internal combustion engines fundamentals. Segunda edición. McGraw-Hill Series in Mechanical Engineering, 2003, p. 1-138.
- [4] ANTONIO NIEVES y FEDERICO C. DOMINGUES. Metodos numéricos aplicados a la ingeniería. Sugunda edición. CECSA, 2004, p. 457-518.
- [5] NORBERTO NIGRO, MARIO A. STORTI and LEONARDO AMBROGGI. Modelización numérica de un motor de combustión interna monocilíndrico encendido por chispa. 15 edicion. Revista internacional de métodos numéricos para calculo y diseño en ingeniería, 1999.
- [6] ARAQUE MALDONADO, FYGUEROA SALGADO and MARTÍN VALERA. Variación de presión y fracción de masa quemada en motores de combustión interna alternativos. Octavo congreso iberoamericano de ingeniería mecánica, 2007.
- [7] RAMACHANDRAN SUNDEEP. Rapid thermodynamic simulation model of an internal combustion engine on alternate fuels. 2 edicion. Proceedings of the international multiconference of engineers and computer scientists, 2009.

- [8] ISMET SEZER and ATILLA BILGIN. Mathematical analysis of spark ignition engine operation via the combination of the first and second laws of thermodynamics. Proceedings of the royal society A, 2010.
- [9] MATHWORKS: MATLAB Descripción del producto, <http://www.mathworks.com/products/matlab/>.
- [10] Simple Heat Release Applet: <http://www.engr.colostate.edu/~allan/thermo/page6/EngineParm1/engine.html>.
- [11] LOGISTIC NUMERICAL METHODS: Runge-Kutta 4th order method, http://numericalmethods.eng.usf.edu/topics/runge_kutta_4th_method.html

ANEXO A
MANUAL DE USUARIO.

1. INTRODUCCION

MATISSE 1.0 es una herramienta computacional diseñada y construida con el objetivo de proporcionar a los estudiantes de la asignatura motores de combustión interna de la Universidad Pontificia Bolivariana seccional Bucaramanga un nuevo objeto de aprendizaje para fortalecer su proceso de formación académica; realizar las tareas de diseño y evaluación en menos tiempo y de forma confiable.

MATISSE 1.0 está construida en base a un algoritmo desarrollado en MATLAB que efectúa la solución numérica del sistema de ecuaciones diferenciales planteado para describir el ciclo de gases de un motor de combustión interna.

Usando MATISSE 1.0, se obtienen los diagramas de Presión vs Posición angular del cigüeñal y Presión vs Volumen de 1 motor de combustión interna alternativo, la comparación mediante graficas sobrepuestas de dos motores permite al usuario comprender y analizar mejor la influencia de ciertos parámetros sobre el comportamiento del ciclo de gases.

MATISSE 1.0 fue validado con resultados obtenidos del ciclo teórico de gas ideal para motores de combustión interna y posteriormente comparado con el software desarrollado por el Departamento de ingeniería mecánica de la universidad de Colorado de los Estados Unidos.

2. PAQUETE COMPUTACIONAL

El paquete computacional para la puesta en marcha de la herramienta MATISSE 1.0 está conformada por:

- CD instalador del software Matlab.
- CD con los archivos de la herramienta computacional MATISSE 1.0 ejecutables en Matlab.
- Manual de usuario.
-

3. INICIO MATISSE 1.0

3.1. Requerimientos del sistema.

Los requisitos mínimos con los que debe cumplir el sistema para ejecutar Matlab y correr los archivos de MATISSE 1.0 sin inconvenientes son:

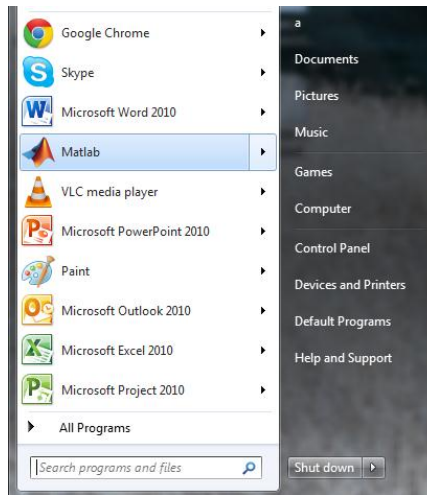
- Procesador de 1 GHz o superior.
- Memoria RAM 256 MB o superior.
- Sistema operativo Windows XP o superior.

3.2. Apertura de MATISSE 1.0 en Matlab.

A continuación se describe el proceso que debe seguir el usuario para la iniciación de MATISSE 1.0.

- Desde el menú de inicio del computador seleccione el software Matlab sobre el cual se va ejecutar el archivo.m de la herramienta computacional MATISSE 1.0. (Ver figura 1).

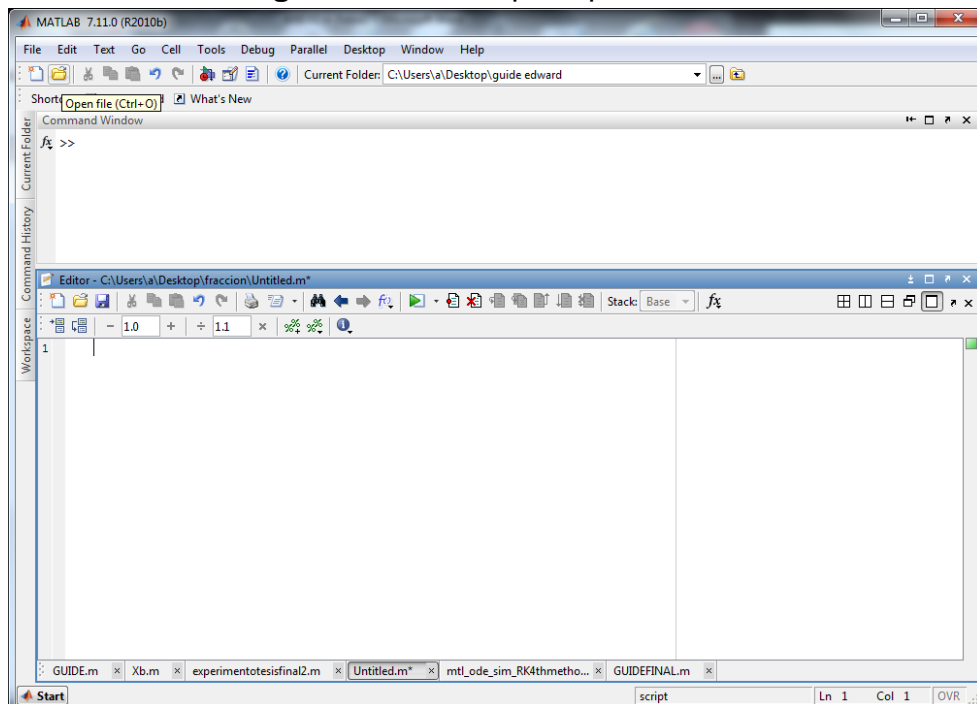
Figura 1: Inicio de Matlab.



Fuente: Matlab.

- A continuación se muestra la pantalla principal de Matlab, en la que se puede ver el área de trabajo y las barras de herramientas que posee Matlab.

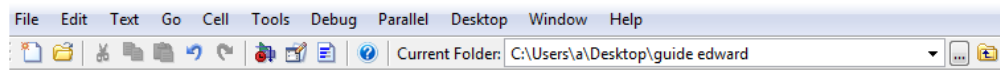
Figura 2: Pantalla principal Matlab.



Fuente: Matlab

- Para abrir MATISSE 1.0 EN Matlab, el usuario debe dirigirse al icono de abrir y posteriormente buscar el sitio en donde se guardo el archivo.m que contiene la herramienta computacional. (ver figuras 3 y 4).

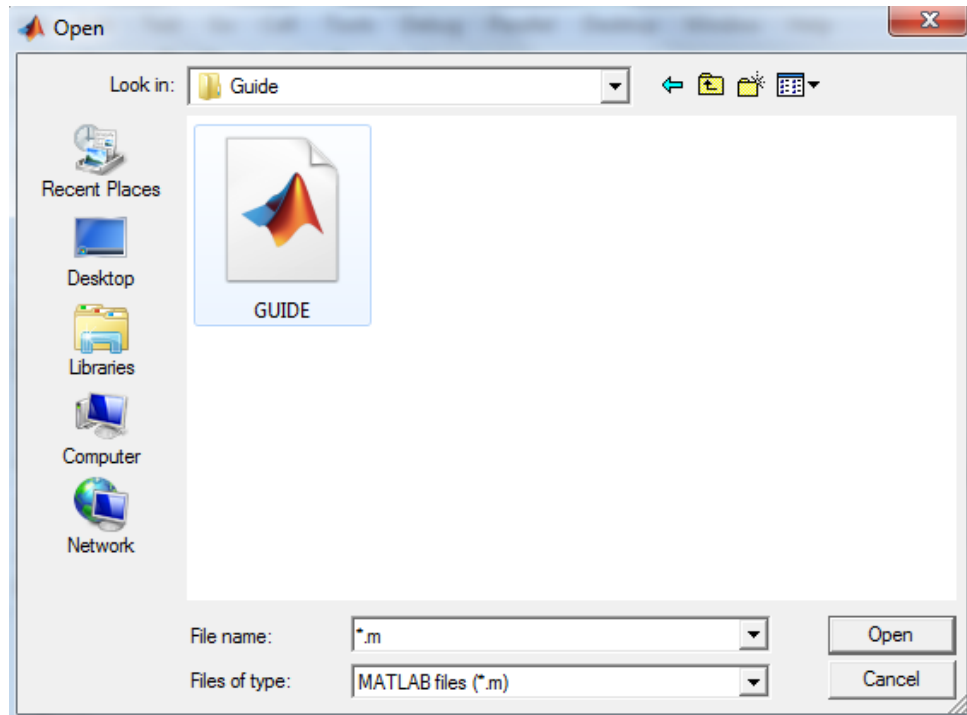
Figura 3: Selección del icono “open” en la pantalla principal de Matlab.



↑
Abrir

Fuente: Matlab.

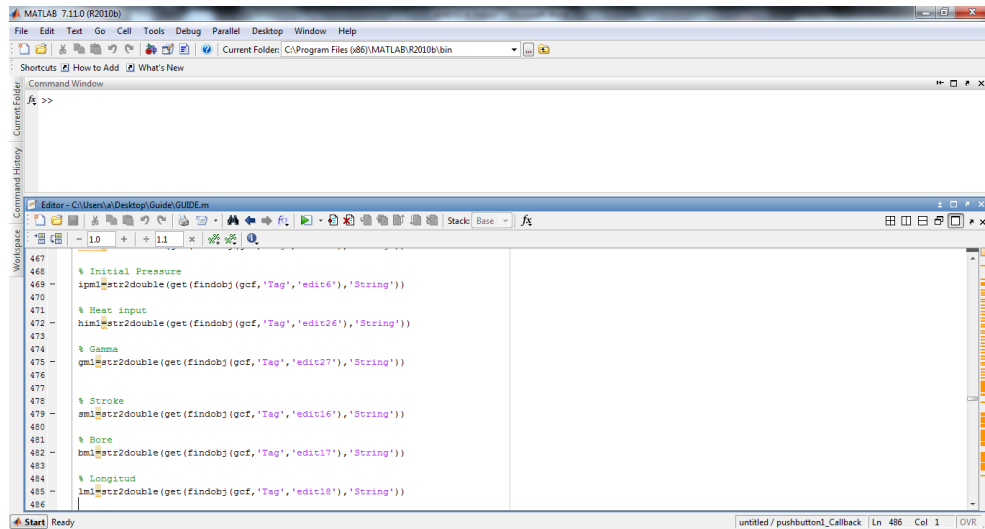
Figura 4: Selección del archivo.



Fuente: Matlab.

- Luego de haber seleccionado el archivo.m, automáticamente aparece en el editor de Matlab el algoritmo desarrollado que sigue el proceso de solución para la herramienta computacional MATISSE 1.0. (ver figura 5).

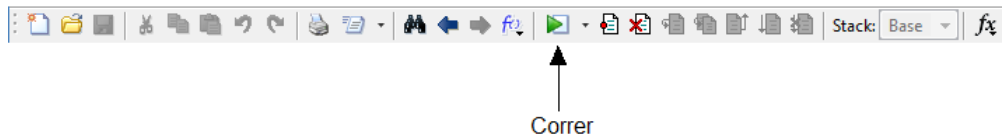
Figura 5: Visualización del código de la herramienta.



Fuente: Matlab.

- Para hacer uso de las propiedades de Matisse 1.0, Matlab debe reconocer el archivo.m de este por lo cual se da click en el icono mostrado como se muestra en la siguiente figura. (ver figura 6).

Figura 6: Selección del icono “run” de la herramienta computacional.



Fuente: Matlab.

- Al correr el archivo.m en Matlab, automáticamente se despliega la ventana principal del entorno de trabajo que permite la interacción entre el usuario y el MATISSE 1.0. (ver figura 7).

Figura 7: Ventana Principal de MATISSE 1.0

The screenshot displays the main interface of the MATISSE 1.0 software. It is organized into three main sections for parameter input, each with a title and a list of parameters in a table format. A 'Calcular' button is located in the bottom right corner. At the bottom of the window, there is a footer containing the software name 'MATISSE 1.0', the university name 'UNIVERSIDAD PONTIFICIA BOLIVARIANA FACULTAD DE INGENIERIA MECANICA SECCIONAL BUCARAMANGA', and the university's logo.

PARAMETROS GEOMETRICOS	
s-Carrera [m]	0.10
b-Diametro del piston [m]	0.10
l-Longitud de la biela [m]	0.15
r-Radio de compresion	10.00

ADICION DE CALOR	
Calor de entrada [kJ]	1.80
Gamma	1.40

PARAMETROS DE COMBUSTION	
Inicio de la combustion	-20.00
Duracion de la combustion	40.00
Parametro de Weibe a	5.00
Parametro de Weibe n	3.00
Temperatura Inicial [C]	20.00
Presion inicial [Kpa]	100.00

Calcular

MATISSE 1.0

UNIVERSIDAD PONTIFICIA BOLIVARIANA
FACULTAD DE INGENIERIA MECANICA
SECCIONAL BUCARAMANGA

Fuente: Matlab

4. Descripción del entorno de MATISSE 1.0

MATISSE 1.0 es una herramienta computacional que permite evaluar el ciclo de gases de motores de combustión interna alternativo, sus resultados se basan en los diagramas Presión vs Volumen y Presión vs Posición angular del cigüeñal que describen el comportamiento del ciclo de gases dentro del cilindro.

La ventana principal de MATISSE 1.0 muestra los parámetros iniciales que el usuario debe introducir, dichos parámetros se dividen en tres grupos, parámetros geométricos, parámetros de combustión y parámetros de adición de calor. (ver figuras 7, 8, 9 y 10).

Como recomendación para obtener resultados coherentes con la realidad, el usuario debe tener un conocimiento previo de teoría en materia de motores de combustión interna para introducir los datos solicitados por MATISSE 1.0.

Los parámetros geométricos, de combustión y de adición de calor permiten definir las características por las que se regirá la simulación en la cámara de combustión del motor.

Figura 8: Parámetros Geométricos.

PARAMETROS GEOMETRICOS	
s-Carrera [m]	0.10
b-Diametro del piston [m]	0.10
l-Longitud de la biela [m]	0.15
r-Radio de compresion	10.00

Fuente: Matlab.

Figura 9: Parámetros de combustión.

PARAMETROS DE COMBUSTION	
Inicio de la combustion	-20.00
Duracion de la combustion	40.00
Parametro de Weibe a	5.00
Parametro de Weibe n	3.00
Temperatura Inicial [C]	20.00
Presion inicial [Kpa]	100.00

Fuente: Matlab

Figura 10: Adición de calor

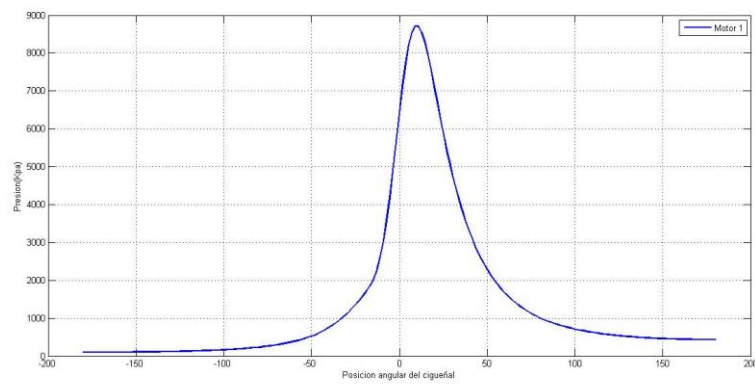
ADICION DE CALOR	
Calor de entrada [kj]	1.80
Gamma	1.40

Fuente: Matlab

5. Resultados obtenidos.

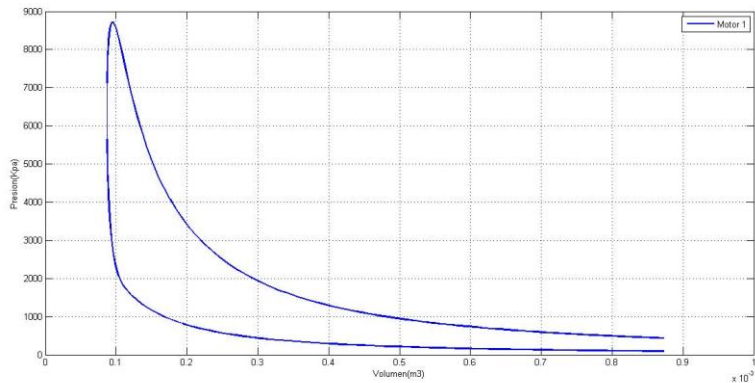
Utilizando MATISSE 1.0 se obtienen como resultados los diagramas de Presión vs Volumen y Presión vs Posición angular que describen el comportamiento del ciclo de gases dentro del cilindro del motor. A continuación se muestran las figuras 11 y 12 que dejan ver dichos diagramas.

Figura 11: Presión vs Posición angular del cigüeñal.



Fuente: Matlab.

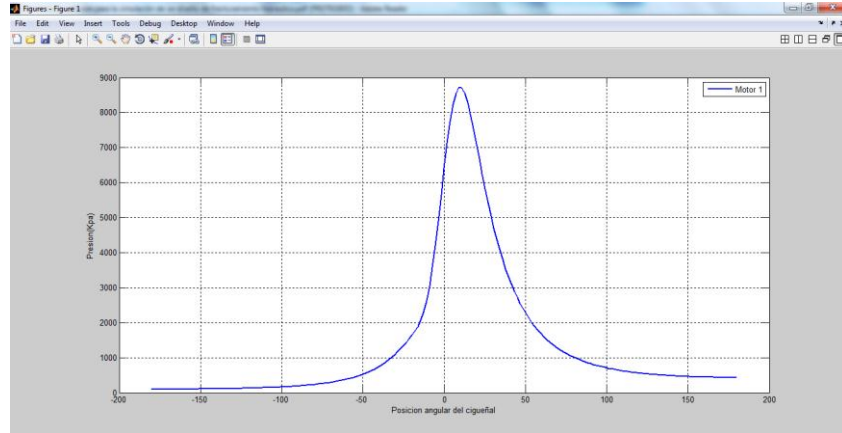
Figura 12: Presión vs Volumen.



Fuente: Matlab.

Los diagramas obtenidos mediante la herramienta computacional MATISSE 1.0 pueden ser modificados mediante las herramientas que posee Matlab para la graficas en 2D. (Ver figuras 13 y 14).

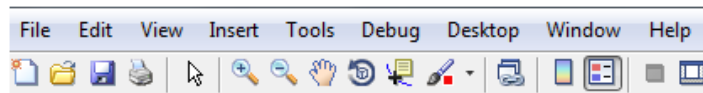
Figura 13: Ventana Principal de la gráfica.



Fuente: Matlab

Figura 14: Barras de herramientas de la gráfica

Barra de herramientas principal



















Barra de herramientas de la figura

Fuente: Matlab

La siguiente tabla muestra la descripción de los iconos principales para las graficas en 2D que posee Matlab.

Tabla1. Descripción de iconos.

Icono	Descripción
	Nueva figura.
	Abrir archivo.
	Guardar archivo.
	Imprimir figura.
	Editar figura.
	Aumentar zoom.
	Disminuir zoom.
	Cursor.
	Rotar 3-D.
	Permite observar el valor de los ejes (x,y) en la figura.
	Cepillado de datos.
	Enlace de datos.
	Insertar barras de color.
	Insertar leyenda.
	Ocultar herramientas de dibujo.
	Mostrar herramientas de dibujo.

Fuente: Los autores

ANEXO B.
CODIGO COMPUTACIONAL PARA EL DESARROLLO DE MATTISE 1.0 EN
MATLAB.

% UNIVERSIDAD PONTIFICIA BOLIVARIANA
% FACULTAD DE INGENIERIA MECANICA
% SECCIONAL BUCARAMANGA

% TRABAJO PRESENTADO POR:
% EDWARD FABIAN OLARTE SILVA
% FABIO ANDRES RIOS QUIJANO

% Director:
%JAVIER MAURICIO CASTELLANOS. Ingeniero mecánico

%Codigo Base tomado de:
% Nathan Collier, Autar Kaw
% Department of Mechanical Engineering
% University of South Florida

%Asesorado por:
%EDWIN JESÚS CÓRDOBA TUTA. Ingeniero Mecánico
%Docente de la Universidad Pontificia Bolivariana seccional Bucaramanga
% Facultad de ingeniería mecánica

%Codigo de generacion de la GUIDE
function varargout = untitled(varargin)

```
gui_Singleton = 1;  
gui_State = struct('gui_Name',    mfilename, ...  
                  'gui_Singleton', gui_Singleton, ...  
                  'gui_OpeningFcn', @untitled_OpeningFcn, ...  
                  'gui_OutputFcn', @untitled_OutputFcn, ...  
                  'gui_LayoutFcn', [] , ...  
                  'gui_Callback', []);  
if nargin && ischar(varargin{1})  
    gui_State.gui_Callback = str2func(varargin{1});  
end  
  
if nargout  
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});  
else  
    gui_mainfcn(gui_State, varargin{:});  
end
```



```
% No editar
```

```
function untitled_OpeningFcn(hObject, eventdata, handles, varargin)
```

```
handles.output = hObject;
```

```
guidata(hObject, handles);
```

```
%Codigo Para insertar imagenes dentro de la GUIDE
```

```
axes(handles.axes1)
```

```
background = imread('upb.jpg');
```

```
axis off;
```

```
imshow(background);
```

```
axes(handles.axes2)
```

```
background = imread('nombre.jpg');
```

```
axis off;
```

```
imshow(background);
```

```
axes(handles.axes3)
```

```
background = imread('portada.jpg');
```

```
axis off;
```

```
imshow(background);
```

```
function varargout = untitled_OutputFcn(hObject, eventdata, handles)
```

```
varargout{1} = handles.output;
```

```
function edit1_Callback(hObject, eventdata, handles)
```

```
function edit1_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),
```

```
get(0,'defaultUicontrolBackgroundColor'))
```

```
    set(hObject,'BackgroundColor','white');
```

```
end
set(hObject,'String','-20.00')%Valor de Inicio de la combustion.
```

```
function edit2_Callback(hObject, eventdata, handles)
```

```
function edit2_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
set(hObject,'String','40.00')%Valor duracion de la Combustion
```

```
function edit3_Callback(hObject, eventdata, handles)
```

```
function edit3_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
set(hObject,'String','5.00')%Valor Parametro de Weibe a
```

```
function edit4_Callback(hObject, eventdata, handles)
```

```
function edit4_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
set(hObject,'String','3.00')%Valor Parametro de Weibe n
```

```
function edit5_Callback(hObject, eventdata, handles)
```

```
function edit5_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),  
get(0,'defaultUicontrolBackgroundColor'))  
    set(hObject,'BackgroundColor','white');  
end  
set(hObject,'String','20.00')%Valor Temperatura inicial.
```

```
function edit6_Callback(hObject, eventdata, handles)
```

```
function edit6_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),  
get(0,'defaultUicontrolBackgroundColor'))  
    set(hObject,'BackgroundColor','white');  
end  
set(hObject,'String','100.00')%Valor Presion Inicial
```

```
function edit16_Callback(hObject, eventdata, handles)
```

```
function edit16_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),  
get(0,'defaultUicontrolBackgroundColor'))  
    set(hObject,'BackgroundColor','white');  
end  
set(hObject,'String','0.10')%Valor de carrera
```

```
function edit17_Callback(hObject, eventdata, handles)
```

```
function edit17_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),  
get(0,'defaultUicontrolBackgroundColor'))  
    set(hObject,'BackgroundColor','white');  
end  
set(hObject,'String','0.10')%Valor Diametro de piston.
```

```
function edit18_Callback(hObject, eventdata, handles)
```

```
function edit18_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),  
get(0,'defaultUicontrolBackgroundColor'))  
    set(hObject,'BackgroundColor','white');  
end  
set(hObject,'String','0.15')%Valor longitud Biela
```

```
function edit19_Callback(hObject, eventdata, handles)
```

```
function edit19_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),  
get(0,'defaultUicontrolBackgroundColor'))  
    set(hObject,'BackgroundColor','white');  
end  
set(hObject,'String','10.00')%Valor Relacion de compresion
```

```
function edit26_Callback(hObject, eventdata, handles)
```

```
function edit26_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),  
get(0,'defaultUicontrolBackgroundColor'))  
    set(hObject,'BackgroundColor','white');  
end  
set(hObject,'String','1.80')%Valor Calor de entrada
```

```
function edit27_Callback(hObject, eventdata, handles)
```

```
function edit27_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),  
get(0,'defaultUicontrolBackgroundColor'))  
    set(hObject,'BackgroundColor','white');  
end  
set(hObject,'String','1.40')%Valor Gamma
```

```

% --- Funcion que ejecuta las variables de entrada
function pushbutton1_Callback(hObject, eventdata, handles)
% Inicio combustion
stm1=str2double(get(findobj(gcf,'Tag','edit1'),'String'))

% Duracion combustion
dcm1=str2double(get(findobj(gcf,'Tag','edit2'),'String'))

% Weibe parameter a
wam1=str2double(get(findobj(gcf,'Tag','edit3'),'String'))

% Weibe parameter n
wnm1=str2double(get(findobj(gcf,'Tag','edit4'),'String'))

% Temperatura inicial
tim1=str2double(get(findobj(gcf,'Tag','edit5'),'String'))

% Presion inicial
ipm1=str2double(get(findobj(gcf,'Tag','edit6'),'String'))

% Calor de entrada
him1=str2double(get(findobj(gcf,'Tag','edit26'),'String'))

% Gamma
gm1=str2double(get(findobj(gcf,'Tag','edit27'),'String'))

% Carrera
sm1=str2double(get(findobj(gcf,'Tag','edit16'),'String'))

% Diametro Piston
bm1=str2double(get(findobj(gcf,'Tag','edit17'),'String'))

% Longitud Biela
lm1=str2double(get(findobj(gcf,'Tag','edit18'),'String'))

% Radio de compresion
rcm1=str2double(get(findobj(gcf,'Tag','edit19'),'String'))
rcm2=str2double(get(findobj(gcf,'Tag','edit25'),'String'))

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Entradas de las variables y constantes de las ecuaciones

```

```

gama=gm1;%gamma
b1=bm1;%diametro piston
s=sm1;%carrera
l=lm1;%longitud viela
r=rcm1;%radio de compresion

a_gama=num2str(gama);
a_b=num2str(b1);
a_s=num2str(s);
a_l=num2str(l);
a_r=num2str(r);

vd=(pi/4)*(b1^2)*s;%Volumen desplazado
a_vd=num2str(vd);

cst=(-gama*vd)/2;
a_cst=num2str(cst);

vc=(vd)/(r-1);%Volumen muerto superior
a_vc=num2str(vc);

R=(2*l)/s;
a_R=num2str(R);

a1=l/R;
a_a1=num2str(a1);

cst2=l+a1;
a_cst2=num2str(cst2);

n=wnm1;%Parametro de Weibe n
a_n=num2str(n);

A=wam1;%Parametro de Weibe a
a_A=num2str(A);

qin=him1;%Calor de entrada
a_qin=num2str(qin);

tetaD=(dcm1*pi)/180;%Duracion de la combustion
a_tetaD=num2str(tetaD);

tetaS=(stm1*pi)/180;%Inicio de la combustion
a_tetaS=num2str(tetaS);

```

```

cst3=(qin/tetaD)*n*A;
a_cst3=num2str(cst3);

cst4=(gama-1);
a_cst4=num2str(cst4);

% ECUACIONES DIFERENCIALES DE COMPRESION Y COMBUSTION

fcnstr=strcat('(','a_cst,')/(','a_vc,')+((pi/4)*(','a_b, '^2,')*['a_cst2,'-(','a_l, '^2, '-
(sin(x))^2*(1, '*'a_a1, '^2,')^0.5)+',a_a1, '*cos(x)))])*y*sin(x)*[1+cos(x)*(','a_R,')^2-
(sin(x))^2]^(-0.5)'];
f=inline(fcnstr) ; %ecuacion para la compresion

gcnstr=strcat(['(','a_cst4,')/(','a_vc,')+((pi/4)*(','a_b, '^2,')*['a_cst2,'-(','a_l, '^2, '-
(sin(x))^2*(1, '*'a_a1, '^2,')^0.5)+',a_a1, '*cos(x)))])*(','a_cst3,')*(((x-
'a_tetaS,')/(','a_tetaD,'))^('a_n, '-1)]*(1-(1-exp(-',a_A, '(((x-
'a_tetaS,')/(','a_tetaD,'))^('a_n, ')))))+0*y');
g=inline(gcnstr) ; %ecuacion de la combustion

%x0, valor inicial de la posicion angular del cigueñal.
x0=-(180*pi)/180 ;

% y0, valor inicial de la presion

y0=ipm1 ;

% xf, valor final de la posicion angular del cigueñal
xf=(180*pi)/180 ;

% n, numero de pasos

n=200 ;

% *****

% Entradas del metodo que seran usadas
disp(sprintf('\n\n *****Datos de
entrada *****'))
disp('Parametros para que el codigo inicie')
disp('changed in the m-file input section')
disp(sprintf('\n f = dy/dx '))
disp(sprintf(' x0 = inicial x '))

```

```

disp(sprintf(' y0 = inicial y '))
disp(sprintf(' xf = final x '))
disp(sprintf(' n = numero de pasos'))
format short g
disp(sprintf('\n-----\n'))
disp(sprintf([' f(x,y) = dy/dx = ' fcnstr]))
disp(sprintf(' x0 = %g',x0))
disp(sprintf(' y0 = %g',ipm1))
disp(sprintf(' xf = %g',xf))
disp(sprintf(' n = %g',n))
disp(sprintf('\n-----'))
disp(sprintf('para esta simulacion, los siguientes parametros son constantes.\n'))
h=(xf-x0)/n ;

disp(sprintf(' h = ( xf - x0 ) / n '))
disp(sprintf(' = ( %g - %g ) / %g ',xf,x0,n))
disp(sprintf(' = %g',h))
% programacion para sacar la admision

xa(1)=x0 ;
ya(1)=ipm1 ;

% La simulacion del metodo Comienza aqui.
disp(sprintf('\n\n*****Simulation*****'))

for i=1:n %
    disp(sprintf('\nStep %g',i))
    disp(sprintf('-----'))

% Agregando el tamaño de paso.
    xa(i+1)=xa(i)+h ;
    if(xa(i)>=tetaS && xa(i)<=tetaS+tetaD)
% Calcular k1, k2, k3, y k4
        k1 = f(xa(i),ya(i))+g(xa(i),ya(i)) ;
        k2 = f(xa(i)+0.5*h,ya(i)+0.5*k1*h)+g(xa(i)+0.5*h,ya(i)+0.5*k1*h) ;
        k3 = f(xa(i)+0.5*h,ya(i)+0.5*k2*h)+g(xa(i)+0.5*h,ya(i)+0.5*k2*h) ;
        k4 = f(xa(i)+h,ya(i)+k3*h)+g(xa(i)+h,ya(i)+k3*h) ;
    else
        % Calcular k1, k2, k3, y k4
        k1 = f(xa(i),ya(i)) ;
        k2 = f(xa(i)+0.5*h,ya(i)+0.5*k1*h) ;
        k3 = f(xa(i)+0.5*h,ya(i)+0.5*k2*h) ;
        k4 = f(xa(i)+h,ya(i)+k3*h) ;
    end
end

```



```
% Formula Runge-Kutta de Cuarto orden.
ya(i+1)=ya(i)+1/6*(k1+2*k2+2*k3+k4)*h ;
```

```
disp('1) Encontrar k1 y k2 usando la informacion previa.')
```

```
disp(sprintf(' k1 = f( x%g , y%g )',i-1,i-1))
disp(sprintf(' = f( %g , %g )',xa(i),ya(i)))
disp(sprintf(' = %g\n',k1))
```

```
disp(sprintf(' k2 = f( x%g + 0.5 * h , y%g + 0.5 * k1 * h )',i-1,i-1))
disp(sprintf(' = f( %g + 0.5 * %g , %g + 0.5 * %g * %g)',xa(i),h,ya(i),k1,h))
disp(sprintf(' = f( %g , %g )',xa(i)+0.5*h,ya(i)+0.5*k1*h))
disp(sprintf(' = %g\n',k2))
```

```
disp(sprintf(' k3 = f( x%g + 0.5 * h , y%g + 0.5 * k2 * h )',i-1,i-1))
disp(sprintf(' = f( %g + 0.5 * %g , %g + 0.5 * %g * %g)',xa(i),h,ya(i),k2,h))
disp(sprintf(' = f( %g , %g )',xa(i)+0.5*h,ya(i)+0.5*k2*h))
disp(sprintf(' = %g\n',k3))
```

```
disp(sprintf(' k4 = f( x%g + h , y%g + k3*h)',i-1,i-1))
disp(sprintf(' = f( %g , %g )',xa(i)+h,ya(i)+k3*h))
disp(sprintf(' = %g\n',k4))
```

```
disp(sprintf('2) Aplicacion del metodo Runge-Kutta de Cuarto orden para calcular
y%g',i))
```

```
disp(sprintf(' y%g = y%g + 1/6*( k1 + 2*k2 + 2*k3 +k4) * h',i,i-1))
disp(sprintf(' = %g + %g * %g * %g',ya(i),1/6,(k1+2*k2+2*k3+k4),h))
disp(sprintf(' = %g\n',ya(i+1)))
disp(sprintf(' at x%g = %g',i,xa(i+1)))
```

```
end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%-----
```

%MODELO TEORICO P-V DEI MOTOR

```
vbdc=vd+vc
gk=273;
T1=26;
T1a=T1+gk;%temperatura dada en Kelvin
cv=0.718;
cp=1.005;
densidad=1.2;
masa=vbdc*densidad;
```

```

vd1a=vc:(vbdc-vc)/10000:vbdc
vd2a=vbdc:(vc-vbdc)/10000:vc
rc1=vbdc.*(vd2a.^-1)
rc2=[vd1a/vc]
p1=y0%presion inicial
p2=y0.*(rc1.^gama)
T2=T1a.*(r.^(gama-1));

T3=(qin/(cv*masa))+T2;
p3=p2(10001)*(T3/T2)

p4=p3.*[(rc2.^-1).^gama]
p5=y0

P=[p1,p2,p3,p4,p5];
V12=[vbdc,vd2a,vc,vd1a,vbdc];

%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% METODO NUMERICO PARA LA SOLUCION DE LA ECUACION
DIFERENCIAL DEL VOLUMEN

fcnstrv=strcat('(((,a_vd,')/(2))*sin(x)*[1+cos(x)*((,a_R,')^2-(sin(x))^2)^(-0.5)]+0*y')
;
fv=inline(fcnstrv) ;%ecuacion del volumen
%valor inicial de la posicion angular del cigueñal.
x0v= -pi ;
%valor inicial del volumen
y0v=vbdc ;
%valor final de la posicion angular del cigueñal.
xfv= pi ;
% Numero de pasos
nv=200 ;

% *****

% Entradas del metodo que seran usadas

```

```

disp(sprintf('\n\n*****Datos de
entrada*****'))
disp('Parametros para que el codigo inicie')
disp('changed in the m-file input section')
disp(sprintf('\n  f = dy/dx '))
disp(sprintf('  x0 = inicial x '))
disp(sprintf('  y0 = inicial y '))
disp(sprintf('  xf = final x '))
disp(sprintf('  n = numero de pasos'))
format short g
disp(sprintf('\n-----\n'))
disp(sprintf(['  f(x,y) = dy/dx = ' fcnstrv]))
disp(sprintf('  x0 = %g',x0v))
disp(sprintf('  y0 = %g',y0v))
disp(sprintf('  xf = %g',xfv))
disp(sprintf('  n = %g',nv))
disp(sprintf('\n-----'))
disp(sprintf('para esta simulacion, los siguientes parametros son constantes.\n'))
hv=(xfv-x0v)/nv ;
disp(sprintf('  h = ( xf - x0 ) / n '))
disp(sprintf('    = ( %g - %g ) / %g ',xfv,x0v,nv))
disp(sprintf('    = %g',hv))

xav(1)=x0v ;
yav(1)=y0v ;

% La simulacion del metodo Comienza aqui..
disp(sprintf('\n\n*****Simulacion*****'))

for i=1:nv
  disp(sprintf('\nStep %g',i))
  disp(sprintf('-----'))

% Agregando el tamaño de paso.
  xav(i+1)=xav(i)+hv ;

% Calcular k1, k2, k3, y k4
  k1 = fv(xav(i),yav(i)) ;
  k2 = fv(xav(i)+0.5*hv,yav(i)+0.5*k1*hv) ;
  k3 = fv(xav(i)+0.5*hv,yav(i)+0.5*k2*hv) ;
  k4 = fv(xav(i)+hv,yav(i)+k3*hv) ;

% Formula Runge-Kutta de Cuarto orden.
  yav(i+1)=yav(i)+1/6*(k1+2*k2+2*k3+k4)*hv ;

```

```

disp('1) Encontrar k1 y k2 usando la informacion previa.')
disp(sprintf(' k1 = f( x%g , y%g )',i-1,i-1))
disp(sprintf(' = f( %g , %g )',xav(i),yav(i)))
disp(sprintf(' = %g\n',k1))

disp(sprintf(' k2 = f( x%g + 0.5 * h , y%g + 0.5 * k1 * h )',i-1,i-1))
disp(sprintf(' = f( %g + 0.5 * %g , %g + 0.5 * %g * %g)',xav(i),hv,yav(i),k1,hv))
disp(sprintf(' = f( %g , %g )',xav(i)+0.5*hv,yav(i)+0.5*k1*hv))
disp(sprintf(' = %g\n',k2))

disp(sprintf(' k3 = f( x%g + 0.5 * h , y%g + 0.5 * k2 * h )',i-1,i-1))
disp(sprintf(' = f( %g + 0.5 * %g , %g + 0.5 * %g * %g)',xav(i),hv,yav(i),k2,hv))
disp(sprintf(' = f( %g , %g )',xav(i)+0.5*hv,yav(i)+0.5*k2*hv))
disp(sprintf(' = %g\n',k3))

disp(sprintf(' k4 = f( x%g + h , y%g + k3*h)',i-1,i-1))
disp(sprintf(' = f( %g , %g )',xav(i)+hv,yav(i)+k3*hv))
disp(sprintf(' = %g\n',k1))

disp(sprintf('2) Aplicacion del metodo Runge-Kutta de Cuarto orden para calcular
y%g',i))
disp(sprintf(' y%g = y%g + 1/6*( k1 + 2*k2 + 2*k3 +k4) * h',i,i-1))
disp(sprintf(' = %g + %g * %g * %g',yav(i),1/6,(k1+2*k2+2*k3+k4),hv))
disp(sprintf(' = %g\n',yav(i+1)))
disp(sprintf(' at x%g = %g',i,xav(i+1)))
end

%-----

%Codigo para la generacion de las graficas.

%Figura presion contra posicion angular del cigueñal
figure(1)
subplot(1,1,1);plot((xa*180)/pi,ya,'-', 'LineWidth',2,'Color',[0 0 1]);
hold on
grid on
xlabel('Posicion angular del cigueñal');ylabel('Presion(Kpa)');
title('Presion vs Posicion angular del cigueñal');
legend('Motor 1','Motor 2');

%Figura presion contra volumen en la camara de combustion
figure(2)
subplot(1,1,1);plot(yav,ya,'-', 'LineWidth',2,'Color',[0 0 1]);

```

```

hold on
grid on
xlabel('Volumen(m3)');ylabel('Presion(Kpa)');
title('Presion vs Volumen del Cilindro');
legend('Motor 1','Motor 2');

%Comparacion de la simulacion numerica contra el metodo teorico.
figure(3)
subplot(1,1,1); plot(V12,P,'--','LineWidth',2,'Color',[1 0 0]);
hold on
grid on
xlabel('Volumen(m3)');ylabel('Presion(Kpa)');
title('Presion vs Volumen');
plot(yav,ya,'-','LineWidth',2,'Color',[0 0 1]);
legend('Resultado Ideal','Resultado Numerico');

%%% Codigo para guardar en excel.
Entradas = {'Inicio combus.', 'Duracion combus.', 'Weibe a','Weibe n','Temp.
inicial','Presion inicial','Calor entrada','Gamma','Carrera','Diame. piston','Longitud
biela','Radio comp' ; stm1 dcm1 wam1 wnm1 tim1 ipm1 him1 gm1 sm1 bm1 lm1
rcm1};

xlswrite('datos_motor',Entradas , 'hoja1', 'A2')

```

ANEXO C.
**CASOS DE ESTUDIO PARA LA SOLUCIÓN DE ECUACIONES
DIFERENCIALES DE PRIMER ORDEN MEDIANTE LA APLICACIÓN DEL
MÉTODO NUMÉRICO DE RUNGE KUTTA DE CUARTO ORDEN.**

A continuación se presentan el proceso de solución para problemas de ecuaciones diferenciales de primer orden mediante la aplicación del método numérico de Runge-Kutta de cuarto orden.

- Caso de estudio numero 1:

$$\frac{dy}{dx} = 3 * e^{-x} - 0.3 * y \quad y_0=3, x_0=0, h=1 \text{ encontrar } y_3$$

$$y(i+1)=y(i)+1/6*(k_1+2*k_2+2*k_3+k_4)*h$$

$$k_1 = f(x(i) , y(i))$$

$$k_2 = f(x(i)+0.5*h , y(i)+0.5*k_1*h)$$

$$k_3 = f(x(i)+0.5*h , y(i)+0.5*k_2*h)$$

$$k_4 = f(x(i)+h , y(i)+k_3*h)$$

$$\text{➤ } i=0, x_0=0, y_0=3, h=1$$

$$k_1 = f(x_0 , y_0)$$

$$=f(0,3)$$

$$= 3e^{-0} - 0.3 * 3$$

$$=2.1$$

$$k_2 = f(x_0+0.5*h , y_0+0.5*k_1*h)$$

$$=f(0+0.5*1 , 3+0.5*2.1*1)$$

$$= f(0.5 , 4.05)$$

$$=3e^{-0.5} - 0.3 * 4.05$$

$$=0.60459$$

$$\begin{aligned}
k_3 &= f(x_0+0.5*h, y_0+0.5*k_2*h) \\
&= f(0+0.5*1, 3+0.5*0.60459*1) \\
&= f(0.5, 3.302295) \\
&= 3e^{-0.5} - 0.3 * 3.302295 \\
&= 0.8289
\end{aligned}$$

$$\begin{aligned}
k_4 &= f(x_0+h, y_0+k_3*h) \\
&= f(0+1, 3+0.8289*1) \\
&= f(1, 3.8289) \\
&= 3e^{-1} - 0.3 * 3.8289 \\
&= 2.1
\end{aligned}$$

$$\begin{aligned}
y_1 &= y_0+1/6*(k_1+2*k_2+2*k_3+k_4)*h \\
&= 3+1/6*(2.1+2*0.60459+2*0.8289+-0.04503)*1 \\
&= 3.82033
\end{aligned}$$

$$x_1=1, y_1= 3.82033$$

$$\triangleright \quad i=1, x_1=1, y_1= 3.82033, h=1$$

$$\begin{aligned}
k_1 &= f(x_1, y_1) \\
&= f(1, 3.82033) \\
&= 3e^{-1} - 0.3 * 3.82033 \\
&= -0.0424596
\end{aligned}$$

$$\begin{aligned}
k_2 &= f(x_1+0.5*h, y_1+0.5*k_1*h) \\
&= f(1+0.5*1, 3.820325+0.5*-0.0424596*1) \\
&= f(1.5, 3.7991) \\
&= 3e^{-1.5} - 0.3 * 3.7991
\end{aligned}$$

$$=-0.470338$$

$$\begin{aligned}k_3 &= f(x_1 + 0.5 * h, y_1 + 0.5 * k_2 * h) \\ &= f(1 + 0.5 * 1, 3.82033 + 0.5 * -0.470338 * 1) \\ &= f(1.5, 3.58516) \\ &= 3e^{-1.5} - 0.3 * 3.58516 \\ &=-0.406157\end{aligned}$$

$$\begin{aligned}k_4 &= f(x_1 + h, y_1 + k_3 * h) \\ &= f(2, 3.41417) \\ &= 3e^{-2} - 0.3 * 3.41417 \\ &= -0.0424596\end{aligned}$$

$$\begin{aligned}y_2 &= y_1 + 1/6 * (k_1 + 2 * k_2 + 2 * k_3 + k_4) * h \\ &= 3.82033 + 0.166667 * -2.41369 * 1 \\ &= 3.41804\end{aligned}$$

$$x_2=2, y_1= 3.41804$$

$$\triangleright \quad i=2, x_2=2, y_2= 3.41804, h=1$$

$$\begin{aligned}k_1 &= f(x_2, y_2) \\ &= f(2, 3.41804) \\ &= 3e^{-2} - 0.3 * 3.41804 \\ &= -0.619407\end{aligned}$$

$$\begin{aligned}k_2 &= f(x_2 + 0.5 * h, y_2 + 0.5 * k_1 * h) \\ &= f(2 + 0.5 * 1, 3.41804 + 0.5 * -0.619407 * 1)\end{aligned}$$

$$\begin{aligned}
&= f(2.5, 3.10834) \\
&= 3e^{-2.5} - 0.3 * 3.10834 \\
&= -0.686247
\end{aligned}$$

$$\begin{aligned}
k_3 &= f(x_2 + 0.5 * h, y_2 + 0.5 * k_2 * h) \\
&= f(2 + 0.5 * 1, 3.41804 + 0.5 * -0.686247 * 1) \\
&= f(2.5, 3.07492) \\
&= 3e^{-2.5} - 0.3 * 3.07492 \\
&= -0.676221
\end{aligned}$$

$$\begin{aligned}
k_4 &= f(x_2 + h, y_2 + k_3 * h) \\
&= f(3, 2.74182) \\
&= 3e^{-3} - 0.3 * 2.74182 \\
&= -0.619407
\end{aligned}$$

$$\begin{aligned}
y_3 &= y_2 + 1/6 * (k_1 + 2 * k_2 + 2 * k_3 + k_4) * h \\
&= 3.41804 + 0.166667 * -4.01753 * 1 \\
&= 2.74846
\end{aligned}$$

$$\mathbf{x_3=3, y_3= 2.74846}$$

- Caso de estudio numero 2:

$$\frac{dy}{dx} = 5 * x + x^2 + 5 * y \quad y_0=4, x_0=0, h=0.5 \text{ encontrar } y_3$$

$$\text{➤ } i=0, x_0=0, y_0=4, h=0.5$$

$$k_1 = f(x_0, y_0)$$

$$\begin{aligned}
&= f(0, 4) \\
&= 5 * 0 + 0^2 + 5 * 4 \\
&= 20
\end{aligned}$$

$$\begin{aligned}
k_2 &= f(x_0 + 0.5 * h, y_0 + 0.5 * k_1 * h) \\
&= f(0 + 0.5 * 0.5, 4 + 0.5 * 20 * 0.5) \\
&= f(0.25, 9) \\
&= 5 * 0.25 + 0.25^2 + 5 * 9 \\
&= 46.3125
\end{aligned}$$

$$\begin{aligned}
k_3 &= f(x_0 + 0.5 * h, y_0 + 0.5 * k_2 * h) \\
&= f(0 + 0.5 * 0.5, 4 + 0.5 * 46.3125 * 0.5) \\
&= f(0.25, 15.5781) \\
&= 5 * 0.25 + 0.25^2 + 5 * 15.5781 \\
&= 79.2031
\end{aligned}$$

$$\begin{aligned}
k_4 &= f(x_0 + h, y_0 + k_3 * h) \\
&= f(0.5, 43.6016) \\
&= 5 * 0.5 + 0.5^2 + 5 * 43.6016 \\
&= 20
\end{aligned}$$

$$\begin{aligned}
y_1 &= y_0 + 1/6 * (k_1 + 2 * k_2 + 2 * k_3 + k_4) * h \\
&= 4 + 0.166667 * 491.789 * 0.5 \\
&= 44.9824
\end{aligned}$$

$$x_1 = 0.5, y_1 = 44.9824$$

$$\triangleright \quad i=1, x_1=0.5, y_1=44.9824, h=0.5$$

$$\begin{aligned}
k_1 &= f(x_1, y_1) \\
&= f(0.5, 44.9824) \\
&= 5 * 0.5 + 0.5^2 + 5 * 44.9824 \\
&= 227.662
\end{aligned}$$

$$\begin{aligned}
k_2 &= f(x_1 + 0.5 * h, y_1 + 0.5 * k_1 * h) \\
&= f(0.5 + 0.5 * 0.5, 44.9824 + 0.5 * 227.662 * 0.5) \\
&= f(0.75, 101.898) \\
&= 5 * 0.75 + 0.75^2 + 5 * 101.898 \\
&= 513.802
\end{aligned}$$

$$\begin{aligned}
k_3 &= f(x_1 + 0.5 * h, y_1 + 0.5 * k_2 * h) \\
&= f(0.5 + 0.5 * 0.5, 44.9824 + 0.5 * 513.802 * 0.5) \\
&= f(0.75, 173.433) \\
&= 5 * 0.75 + 0.75^2 + 5 * 173.433 \\
&= 871.477
\end{aligned}$$

$$\begin{aligned}
k_4 &= f(x_1 + h, y_1 + k_3 * h) \\
&= f(1, 480.721) \\
&= 5 * 1 + 1^2 + 5 * 480.721 \\
&= 227.662
\end{aligned}$$

$$\begin{aligned}
y_2 &= y_1 + 1/6 * (k_1 + 2 * k_2 + 2 * k_3 + k_4) * h \\
&= 44.9824 + 0.166667 * 5407.83 * 0.5 \\
&= 495.635
\end{aligned}$$

$$x_2 = 1, y_2 = 495.635$$

➤ $i=2, x_2=1, y_2=495.635, h=0.5$

$$\begin{aligned}
k_1 &= f(x_2, y_2) \\
&= f(1, 495.635) \\
&= 5 * 1 + 1^2 + 5 * 495.635 \\
&= 2484.17
\end{aligned}$$

$$\begin{aligned}
k_2 &= f(x_2 + 0.5 * h, y_2 + 0.5 * k_1 * h) \\
&= f(1 + 0.5 * 0.5, 495.635 + 0.5 * 2484.17 * 0.5) \\
&= f(1.25, 1116.68) \\
&= 5 * 1.25 + 1.25^2 + 5 * 1116.68 \\
&= 5591.2
\end{aligned}$$

$$\begin{aligned}
k_3 &= f(x_2 + 0.5 * h, y_2 + 0.5 * k_2 * h) \\
&= f(1 + 0.5 * 0.5, 495.635 + 0.5 * 5591.2 * 0.5) \\
&= f(1.25, 1893.44) \\
&= 5 * 1.25 + 1.25^2 + 5 * 1893.44 \\
&= 9474.99
\end{aligned}$$

$$\begin{aligned}
k_4 &= f(x_2 + h, y_2 + k_3 * h) \\
&= f(1.5, 5233.13) \\
&= 5 * 1.5 + 1.5^2 + 5 * 5233.13 \\
&= 2484.17
\end{aligned}$$

$$\begin{aligned}
y_3 &= y_2 + 1/6 * (k_1 + 2 * k_2 + 2 * k_3 + k_4) * h \\
&= 495.635 + 0.166667 * 58792 * 0.5 \\
&= 5394.96
\end{aligned}$$

$$\mathbf{X_3=1.5, y_3= 5394.96}$$

- Caso de estudio numero 3:

$$\frac{dy}{dx} = 2 * e^{-y} + y^x \quad y_0=1, x_0=0.5, h=1/3 \text{ encontrar } y_3$$

➤ $i=0, x_0=0.5, y_0=1, h=1/3$

$$\begin{aligned} k_1 &= f(x_0, y_0) \\ &= f(0.5, 1) \\ &= 2 * e^{-1} + 1^{0.5} \\ &= 1.73576 \end{aligned}$$

$$\begin{aligned} k_2 &= f(x_0 + 0.5 * h, y_0 + 0.5 * k_1 * h) \\ &= f(0.5 + 0.5 * 0.333333, 1 + 0.5 * 1.73576 * 0.333333) \\ &= f(0.666667, 1.28929) \\ &= 2 * e^{-1.28929} + 1.28929^{0.666667} \\ &= 1.73552 \end{aligned}$$

$$\begin{aligned} k_3 &= f(x_0 + 0.5 * h, y_0 + 0.5 * k_2 * h) \\ &= f(0.5 + 0.5 * 0.333333, 1 + 0.5 * 1.73552 * 0.333333) \\ &= f(0.666667, 1.28925) \\ &= 2 * e^{-1.28925} + 1.28925^{0.666667} \\ &= 1.73552 \end{aligned}$$

$$\begin{aligned} k_4 &= f(x_0 + h, y_0 + k_3 * h) \\ &= f(0.833333, 1.57851) \\ &= 2 * e^{-1.57851} + 1.57851^{0.833333} \end{aligned}$$

$$= 1.73576$$

$$\begin{aligned}y_1 &= y_0 + 1/6 * (k_1 + 2*k_2 + 2*k_3 + k_4) * h \\ &= 1 + 0.166667 * 10.5533 * 0.333333 \\ &= 1.58629\end{aligned}$$

$$X_1=0.833333 \quad Y_1=1.58629$$

➤ $i=1, x_1=0.833333, y_1=1.58629, h=1/3$

$$\begin{aligned}k_1 &= f(x_1, y_1) \\ &= f(0.833333, 1.58629) \\ &= 2 * e^{-1.58629} + 1.58629^{0.833333} \\ &= 1.87825\end{aligned}$$

$$\begin{aligned}k_2 &= f(x_1 + 0.5 * h, y_1 + 0.5 * k_1 * h) \\ &= f(0.833333 + 0.5 * 0.333333, 1.58629 + 0.5 * 1.87825 * 0.333333) \\ &= f(1, 1.89933) \\ &= 2 * e^{-1.89933} + 1.89933^1 \\ &= 2.19867\end{aligned}$$

$$\begin{aligned}k_3 &= f(x_1 + 0.5 * h, y_1 + 0.5 * k_2 * h) \\ &= f(0.833333 + 0.5 * 0.333333, 1.58629 + 0.5 * 2.19867 * 0.333333) \\ &= f(1, 1.95274) \\ &= 2 * e^{-1.95274} + 1.95274^1 \\ &= 2.23651\end{aligned}$$

$$\begin{aligned}k_4 &= f(x_1 + h, y_1 + k_3 * h) \\ &= f(1.16667, 2.3318) \\ &= 2 * e^{-2.3318} + 2.3318^{1.16667}\end{aligned}$$

$$= 1.87825$$

$$\begin{aligned}y_2 &= y_1 + 1/6 * (k_1 + 2*k_2 + 2*k_3 + k_4) * h \\ &= 1.58629 + 0.166667 * 13.628 * 0.333333 \\ &= 2.3434\end{aligned}$$

$$x_2 = 1.16667 \quad Y_2 = 2.3434$$

➤ $i=2, x_2=1.16667, y_2=2.3434, h=1/3$

$$\begin{aligned}k_1 &= f(x_2, y_2) \\ &= f(1.16667, 2.3434) \\ &= 2 * e^{-2.3434} + 2.3434^{1.16667} \\ &= 2.89278\end{aligned}$$

$$\begin{aligned}k_2 &= f(x_2 + 0.5 * h, y_2 + 0.5 * k_1 * h) \\ &= f(1.16667 + 0.5 * 0.333333, 2.3434 + 0.5 * 2.89278 * 0.333333) \\ &= f(1.33333, 2.82553) \\ &= 2 * e^{-2.82553} + 2.82553^{1.33333} \\ &= 4.1131\end{aligned}$$

$$\begin{aligned}k_3 &= f(x_2 + 0.5 * h, y_2 + 0.5 * k_2 * h) \\ &= f(1.16667 + 0.5 * 0.333333, 2.3434 + 0.5 * 4.1131 * 0.333333) \\ &= f(1.33333, 3.02892) \\ &= 2 * e^{-3.02892} + 3.02892^{1.33333} \\ &= 4.47919\end{aligned}$$

$$\begin{aligned}
k_4 &= f(x_2 + h, y_2 + k_3 \cdot h) \\
&= f(1.5, 3.83647) \\
&= 2 * e^{-3.83647} + 3.83647^{1.5} \\
&= 2.89278
\end{aligned}$$

$$\begin{aligned}
y_3 &= y_2 + 1/6 * (k_1 + 2 * k_2 + 2 * k_3 + k_4) * h \\
&= 2.3434 + 0.166667 * 27.6349 * 0.333333 \\
&= 3.87868
\end{aligned}$$

$$\mathbf{X_3 = 1.5 \quad Y_3 = 3.87868}$$