

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

Cálculo de volúmenes de sólidos de revolución, mediante Modelación Matemática para mejorar el aprendizaje de estudiantes de la Escuela Superior Politécnica de Chimborazo, Facultad de Mecánica, periodo abril-septiembre 2021

JESSICA PAOLA BONILLA ESCUDERO

Trabajo de Titulación modalidad Proyectos de Investigación y Desarrollo, presentado ante el Instituto de Posgrado y Educación Continua de la ESPOCH, como requisito parcial para la obtención del grado de:

**MAGÍSTER EN MATEMÁTICA, MENCIÓN MODELACIÓN Y
DOCENCIA**

RIOBAMBA-ECUADOR

JULIO 2023

DECLARACIÓN DE AUTENTICIDAD

Yo, JESSICA PAOLA BONILLA ESCUERO, declaro que el presente **Trabajo de Titulación modalidad Proyecto de Investigación y Desarrollo**, es de mi autoría y que los resultados del mismo son auténticos y originales. Los textos constantes en el documento que provienen de otra fuente están debidamente citados y referenciados.

Como autora, asumo la responsabilidad legal y académica de los contenidos de este proyecto de investigación de maestría, el patrimonio intelectual pertenece a la Escuela Superior Politécnica de Chimborazo.

Riobamba, Julio 2023



JESSICA PAOLA BONILLA ESCUDERO

No. Cédula: 0604115188

©2023, Jessica Paola Bonilla Escudero.

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de autor.



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

EL TRIBUNAL DE TRABAJO DE TITULACIÓN CERTIFICA QUE:

El **Trabajo de Titulación modalidad Proyectos de Investigación y Desarrollo**, titulado: **Cálculo de volúmenes de sólidos de revolución, mediante Modelación Matemática para mejorar el aprendizaje de estudiantes de la Escuela Superior Politécnica de Chimborazo, Facultad de Mecánica, periodo abril-septiembre 2021, de responsabilidad de la señorita Jessica Paola Bonilla Escudero**, ha sido minuciosamente revisado por los miembros del Tribunal del trabajo de titulación, el mismo que cumple con los requisitos científicos, técnicos, legales, en tal virtud el Tribunal autoriza su presentación.

Dra. Martha Ximena Dávalos Villegas; M. Sc.
PRESIDENTA



Firmado electrónicamente por:
**MARTHA XIMENA
DAVALOS VILLEGAS**

Ing. José Luis Pérez Rojas; Mtr.
DIRECTOR



Firmado electrónicamente por:
**JOSE LUIS PEREZ
ROJAS**

Dra. Silvia Mariana Haro Rivera; Mgtr.
MIEMBRO



Firmado electrónicamente por:
**SILVIA MARIANA HARO
RIVERA**

Dr. Guillermo Edvin Machado Sotomayor; Ph.D.
MIEMBRO



Firmado electrónicamente por:
**GUILLELMO EDVIN
MACHADO SOTOMAYOR**

Riobamba, julio 2023

DEDICATORIA

El presente trabajo dedico a Dios y a la Virgen María por guiarme y estar siempre a mi lado y haberme dado un hijo que se convirtió en el motor de mi vida, y especialmente a mis padres quienes son el pilar fundamental de mi existencia, que con su cariño, confianza y perseverancia me apoyaron incondicionalmente, y me enseñaron a luchar para alcanzar mis metas propuestas. Y a todas las personas que de una u otra forma me ayudaron en la elaboración de este trabajo.

Jessica

AGRADECIMIENTO

Agradezco a Dios por darme sabiduría y bendecirme en la culminación de este sueño y de manera muy especial al equipo de trabajo de esta investigación, a mi tutor, miembros del tribunal por su paciencia, disposición quienes, mediante su dirección y asesoría, basta experiencia, apoyo y confianza supieron encaminar cada uno de los esfuerzos realizados en este trabajo.

A mi familia por su apoyo y dedicación.

Jessica

TABLA DE CONTENIDO

RESUMEN.....	xiv
SUMMARY.....	xv

CAPÍTULO I

1. INTRODUCCIÓN.....	1
1.1. Situación Problemática.....	2
1.2. Formulación del Problema.....	3
1.3. Preguntas Directrices.....	3
1.4. Justificación de la Investigación.....	3
1.5. Objetivos de la Investigación.....	4
1.5.1. <i>Objetivo General</i>	4
1.5.2. <i>Objetivos Específicos</i>	4
1.6. Hipótesis.	5

CAPÍTULO II

2. MARCO TEÓRICO.....	7
2.1. Antecedentes del Problema.....	7
2.2. Bases Teóricas.....	9
2.2.1. Introducción a los Sólidos de Revolución.....	9
2.2.2. <i>Volumen.</i>	9
2.2.3. <i>Integral o antiderivada.</i>	9
2.2.4. <i>Integral definida e indefinida.</i>	10
2.2.5. <i>Sólidos de Revolución.</i>	12
2.2.6. <i>Métodos de cálculo de volúmenes de Sólidos de Revolución.</i>	13
2.2.7. <i>Integración numérica.</i>	16
2.2.8. <i>Fórmulas de Newton-Cotes.</i>	17
2.2.9. <i>Regla del Trapecio.</i>	17
2.2.10. <i>Reglas de Simpson.</i>	20
2.2.11. <i>Resumen Fórmulas de Newton Cotes</i>	23
2.2.12. <i>Métodos avanzados de integración numérica</i>	25

2.2.13. Integración de Romberg	25
2.2.14. Cuadratura de Gauss	26
2.2.15. Herramientas para el cálculo de volúmenes de Sólidos de Revolución.	29
2.2.15.1. Software educativo.....	29
2.2.16. Modelos Matemáticos.	32
2.2.17. Conceptos clave.	33
2.2.18. Ejemplos.	34
2.2.19. Modelación matemática como método de enseñanza.	37
2.2.20. Implementación de los Modelos Matemáticos en Matlab.	38
2.2.21. Programación en Matlab.	38
2.2.22. Integración en Matlab.	39
2.2.23. Diseño de interfaces gráficas.	42
2.3. Marco conceptual.....	44
2.3.1. Metodologías de la enseñanza de Matemática Avanzada en el campo ingenieril.	44
2.3.2. Métodos de instrucción individual.	45
2.3.3. Métodos de instrucción global.	45
2.4. Identificación de variables.....	46

CAPÍTULO III

3. METODOLOGÍA DE INVESTIGACIÓN	50
3.1. Tipo y diseño de la investigación.....	50
3.2. Método de la investigación.	50
3.3. Estudio de la investigación.	50
3.4. Población de estudio.	50
3.5. Selección de la Muestra.	50
3.6. Técnicas e instrumentos de recolección de datos.	51
3.6.1. Técnica.	51
3.6.2. Instrumento.	51

CAÍTULO IV

4. RESULTADOS Y DISCUSIÓN.....	52
4.1. Situación previa.....	52
4.1.1. Análisis e interpretación de los resultados de la encuesta aplicada en la etapa de diagnóstico de la situación actual.	52

4.1.2.	<i>Resultados de las preguntas específicas.</i>	52
4.1.3.	<i>Conclusión sobre los resultados obtenidos en la encuesta aplicada.</i>	57
4.2.	Situación posterior a la propuesta.	57
4.2.1.	<i>Actitud de los estudiantes.</i>	57
4.2.2.	<i>Conclusión sobre los resultados obtenidos luego de aplicar la propuesta.</i>	59

CAPÍTULO V

5.	PROPUESTA	61
5.1.	Introducción.	61
5.2.	Justificación.	61
5.3.	Descripción.	62
5.4.	Objetivos de la propuesta didáctica.	62
5.5.	Resultados de aprendizaje.	63
5.6.	Recursos.	63
5.6.1.	<i>Recursos Materiales.</i>	63
5.6.2.	<i>Recursos Humanos.</i>	63
5.6.3.	<i>Recursos Tecnológicos.</i>	63
5.7.	Capacitación sobre el programa desarrollado en MATLAB.	64
5.7.1.	<i>Modelo matemático utilizado</i>	64
5.7.2.	<i>Funcionamiento del programa desarrollado en MATLAB.</i>	65
5.8.	Fases de la propuesta.	70
5.8.1.	<i>Fase 1 (Motivación).</i>	70
5.8.2.	<i>Fase 2 (Comprensión).</i>	70
5.8.3.	<i>Fase 3 (Adquisición).</i>	71
5.8.4.	<i>Fase 4 (Retención).</i>	71
5.8.5.	<i>Fase 5 (Recuperación).</i>	71
5.8.6.	<i>Fase 6 (Generalización o Transferencia).</i>	71
5.8.7.	<i>Fase 7 (Desempeño).</i>	74
5.8.8.	<i>Fase 8 (Retroalimentación).</i>	77
	CONCLUSIONES	78
	RECOMENDACIONES	79

GLOSARIO

BIBLIOGRAFÍA

ANEXOS

ÍNDICE DE TABLAS

Tabla 1-2:	Fórmulas de integración cerrada de Newton-Cotes	24
Tabla 2-2:	Operacionalización de variables	48
Tabla 3-2:	Matriz de Consistencia	49
Tabla 1-4:	Nivel de comprensión sobre cálculo del volumen de sólidos de revolución	52
Tabla 2-4:	Uso de herramientas tecnológicas	53
Tabla 3-4:	Uso de herramientas para el cálculo del volumen de sólidos de revolución	54
Tabla 4-4:	Herramientas para el cálculo del volumen de sólidos de revolución	55
Tabla 5-4:	Problemas al calcular de volúmenes de sólidos de revolución.....	56
Tabla 6-4:	Utilidad de la interfaz gráfica del programa en MATLAB	57
Tabla 7-4:	Nivel de comprensión en la materia de Cálculo Integral	58
Tabla 1-5:	Modelos matemáticos implementados en el programa	64

ÍNDICE DE FIGURAS

Figura 1-2:	Volumen de sólidos simples (esfera, cubo y cilindro).....	9
Figura 2-2:	Discretización del área bajo la curva.....	11
Figura 3-2:	Área bajo la curva en una integral indefinida (izquierda) y definida (derecha) .	12
Figura 4-2:	Sólidos de revolución conocidos: cilindro, cono, y esfera	13
Figura 5-2:	Esquema del solido de revolución por discos.....	14
Figura 6-2:	Esquema volumen de revolución por arandelas	15
Figura 7-2:	Esquema de solido de revolución por capas cilíndricas	16
Figura 8-2:	Bosquejo integración numérica, método Newton-Cotes	17
Figura 9-2:	Bosquejo integración numérica, Regla del Trapecio	18
Figura 10-2:	Error de truncamiento en la Regla del Trapecio.....	19
Figura 11-2:	Bosquejo integración numérica, Regla de Simpson 1/3	20
Figura 12-2:	Regla de Simpson 1/3 para aplicación múltiple	21
Figura 13-2:	Bosquejo integración numérica, Regla de Simpson 3/8	22
Figura 14-2:	Regla de Simpson 1/3 y 3/8 combinadas para aplicaciones de intervalos impares	25
Figura 15-2:	Comparación método de Newton-Cotes vs cuadratura de Gauss	26
Figura 16-2:	Esquema gráfico de la ubicación de las variables desconocidas x_0 y x_1 integración por la Regla de Gauss	28
Figura 17-2:	Proceso de modelado matemático	33
Figura 18-2:	Diagrama de flujo y gráfica de la estimación de población regional, en Vensim	34
Figura 19-2:	Bosquejo del Enfriamiento de Newton	35
Figura 20-2:	Bosquejo del ejemplo de velocidad de un caminante.....	36
Figura 21-2:	Modelación matemática en el aprendizaje	37
Figura 22-2:	Ventana al iniciar la interfaz GUIDE.....	43
Figura 23-2:	Ejemplo de interfaz gráfica.....	43

Figura 24-2:	Ejemplo de interfaz gráfica ejecutada y lista para utilizar	44
Figura 1-5:	Diseño del programa en MATLAB.....	66
Figura 2-5:	Interfaz Principal del programa en MATLAB	67
Figura 3-5:	Casos posibles y métodos con que trabaja el programa en MATLAB	68
Figura 4-5:	Interfaz función generatriz $f(x)$	69
Figura 5-5:	Gráfica 2D (izquierda) y 3D (derecha).....	69
Figura 6-5:	Representación gráfica de las funciones	72
Figura 7-5:	Representación gráfica de las funciones planteadas	73
Figura 8-5:	Ventana principal de MATLAB.....	75
Figura 9-5:	Abrir programa creado en MATLAB.....	75
Figura 10-5:	Ejecución del programa creado en MATLAB.....	76
Figura 11-5:	Interfaz Principal del programa desarrollado en MATLAB para el cálculo de volumen de sólidos de revolución.	76

ÍNDICE DE GRÁFICOS

Gráfico 1-4:	Nivel de comprensión sobre cálculo del volumen de sólidos de revolución	53
Gráfico 2-4:	Uso de herramientas tecnológicas	53
Gráfico 3-4:	Uso de herramientas para el cálculo del volumen de sólidos de revolución	54
Gráfico 4-4:	Herramientas para el cálculo del volumen de sólidos de revolución	55
Gráfico 5-4:	Problemas al calcular de volúmenes de sólidos de revolución	56
Gráfico 6-4:	Herramientas para el cálculo del volumen de sólidos de revolución	58
Gráfico 7-4:	Nivel de comprensión en la materia de cálculo integral	58

ÍNDICE DE ANEXOS

ANEXO A: Estructuras básicas de programación.

ANEXO B: Ejemplos de cálculo de volumen de sólido de revolución usando el programa.

ANEXO C: Código implementado en el desarrollo del programa.

RESUMEN

El objetivo del presente trabajo de investigación fue calcular el volumen de sólidos de revolución mediante modelación matemática, para mejorar el aprendizaje de estudiantes de la Escuela Superior Politécnica de Chimborazo, de la Facultad de Mecánica, período abril-septiembre 2021. Teniendo como referencia que el cálculo integral y sus aplicaciones representan uno de los principales desafíos de percepción para los alumnos de ingeniería, se aplicó una encuesta a un grupo de 172 alumnos con el diseño cuasiexperimental, en donde se determinó la dificultad existente de resolver problemas de este tema. Los resultados que se obtuvieron demuestran la situación en la que se encontraban los estudiantes donde: la suma de los porcentajes de aquellos alumnos que consideran que su nivel de comprensión respecto al cálculo del volumen fue satisfactorio y muy satisfactorio es de 47,5%, también a pesar de que el 55,9% del total de encuestados manifestaron hacer uso frecuente de herramientas tecnológicas apenas el 27,1% las emplea, el 72,9 % mencionó que no cuentan con alguna herramienta que facilite dicho cálculo y el 54,2% indicaron que la parte que más problema le causa es la visualización gráfica en 3D. El análisis de resultados de la encuesta permitió elaborar una propuesta de aprendizaje y se diseñó la aplicación en MATLAB para el cálculo de volúmenes de revolución, mediante modelación matemática y después de haber aplicado el software, se evidenció que el 84,7% considera que sí ha mejorado su nivel de comprensión en la materia de cálculo y que la interfaz presentada les pareció satisfactoria, y muy útil, sumando un 95% de aceptación. Se recomienda, tener una base de conocimiento sobre el cálculo de volúmenes de revolución, para que al momento de implementar el programa los estudiantes puedan utilizar adecuadamente y mejorar su comprensión sobre este tema.

Palabras clave: <MATEMÁTICAS>, <MODELACIÓN MATEMÁTICA>, <MATLAB (SOFTWARE)>, <SÓLIDOS DE REVOLUCIÓN>, <CÁLCULO>, <APRENDIZAJE>



Firmado electrónicamente por:
**LUIS ALBERTO
CAMINOS VARGAS**



11-07-2023

0075-DBRA-UPT-IPEC-2023

ABSTRACT

The objective of this research work was to calculate the volume of solids of revolution through mathematical modeling, in order to improve the learning of students at the Escuela Superior Politécnica de Chimborazo, Faculty of Mechanics, from April to September 2021. Considering that integral calculus and its applications represent one of the main perception challenges for engineering students, a survey was conducted with a quasi-experimental design involving a group of 172 students, where the existing difficulty in solving problems related to this topic was determined. The results obtained demonstrated the situation in which the students found themselves: the combined percentage of those students who considered their level of understanding regarding volume calculation as satisfactory and very satisfactory was 47.5%. Despite 55.9% of the total respondents stating that they frequently use technological tools, only 27.1% actually employ them. Furthermore, 72.9% mentioned that they do not have any tools that facilitate this calculation, and 54.2% indicated that the most problematic aspect is the 3D graphical visualization. The analysis of survey results allowed the development of a learning proposal, and an application in MATLAB was designed for calculating volumes of revolution through mathematical modeling. After implementing the software, it was evident that 84.7% of the students considered that their level of understanding in the subject of calculus had improved, and they found the presented interface satisfactory and very useful, resulting in a 95% acceptance rate. It is recommended to have a knowledge base on the calculation of volumes of revolution so that when implementing the program, students can use it properly and enhance their understanding of this topic.

Keywords: <MATHEMATICS>, <MATHEMATICAL MODELING>, <MATLAB (SOFTWARE)>, <SOLIDS OF REVOLUTION>, <CALCULUS>, <LEARNING>

CAPÍTULO I

1. INTRODUCCIÓN

Este trabajo comprende sobre el cálculo de volúmenes de sólidos de revolución mediante modelación matemática para mejorar el aprendizaje de los estudiantes de la Escuela Superior Politécnica de Chimborazo, Facultad de Mecánica, en el periodo abril-septiembre 2021, con el objetivo de subsanar la necesidad de mejorar el aprendizaje de los estudiantes en la materia de Cálculo, especialmente sobre el contenido de Sólidos de Revolución, puesto que se trata de uno de los contenidos que más dificultad representa para los estudiantes, (esto se evidenció en técnicas de recolección de datos, como encuestas realizadas a un grupo de estudiantes).

A partir de ello se determinó la posibilidad de utilizar la Modelación Matemática para corregir dicha situación. Al final se empleó un estudio para validar la propuesta sobre los estudiantes que aprobaron la materia de Análisis Matemático I, además de comprobar la mejora de su actitud respecto al estudio del cálculo del volumen de sólidos de revolución.

La propuesta tiene el objetivo de mitigar la dificultad de los estudiantes al tratar con el cálculo de volúmenes de revolución, que se encontró al realizar la encuesta. Es por ello por lo que se emplean modelos matemáticos pertinentes para dicho cálculo, como el método de discos y el de arandelas, a través de una aplicación en Matlab, con una interfaz sencilla de utilizar para los usuarios (estudiantes o docentes) y que permita facilitar el entendimiento de la materia. Esta propuesta fue validada mediante un estudio a los alumnos que aprobaron el curso de Análisis Matemático I, con lo que se verificó su efectividad al mejorar su actitud y aprendizaje de sólidos de revolución, tal como se evidenció en las encuestas finales aplicadas a los estudiantes.

En el Capítulo I se detalla el estudio de la situación problemática, y se delimita el alcance y justificación del trabajo. En el Cap. II se presenta el estado del arte y la fundamentación teórica de los métodos para el cálculo de volúmenes de sólidos de revolución, así como las herramientas o aplicaciones digitales existentes que permiten adquirir un aprendizaje más rápido, en donde es evidente el potencial de las herramientas digitales sobre el proceso de enseñanza-aprendizaje en instituciones educativas. En el Cap. III se encuentra la metodología aplicada y el diseño de la propuesta, donde se detalla el uso de las encuestas empleadas y el cálculo de la muestra de estudiantes para las mismas. El Cap. IV muestra los resultados de las encuestas previas y la implementación de la propuesta, y en el Cap. VI se detalla la validación de la misma. Por otra parte, en ANEXO C se incluye la programación empleada para desarrollar el programa en Matlab, que servirá como propuesta para mejorar el aprendizaje de los estudiantes del periodo señalado. Se espera que la herramienta desarrollada en este trabajo pueda ser utilizada como estrategia

didáctica para futuras generaciones de estudiantes de ingeniería, no solo de la ESPOCH, sino a nivel nacional.

1.1. Situación Problemática

Una de las principales situaciones adversas con que se enfrenta todo estudiante de Ingeniería en el inicio de su formación es la falta de comprensión en la materia del Cálculo Integral (Molina, 2017). Parte de esta problemática puede referirse al docente, por sus métodos de enseñanza; pero también al alumno, por la falta de aplicación. Sin embargo, cuando ninguna de estas situaciones son la causa del mal rendimiento de los alumnos, la falta o escasez de recursos didácticos, tales como herramientas digitales de Tecnologías de Información y Comunicación (TIC), puede ser la causante de estos problemas, es cuando se emplea la Modelación Matemática para hacer de un problema difícil algo fácil de entender (Aravena, Caamaño, & Joaquín, 2008).

Un estudiante de ingeniería se caracteriza porque, a lo largo de toda su formación, necesita un estudio avanzado de la Matemática Superior. En los primeros semestres, las asignaturas que más les cuesta aprender, y sobre todo aprobar, son Cálculo Diferencial y Vectorial, inclusive para aquellos con las bases necesarias, esta asignatura representa un reto, de acuerdo con Molina (2017). Esto implica que tales materias deben incluir especial atención en el momento de la enseñanza ingenieril.

En el contexto de la ESPOCH, Cálculo 1 y Cálculo 2 representan materias tan demandantes que los estudiantes presentan problemas de aprendizaje, específicamente en la interpretación y entendimiento de sólidos de revolución, a más del cálculo de sus volúmenes, tal como se evidenció en las encuestas realizadas. Tales encuestas mostraron la necesidad de implementar técnicas didácticas que mejoren la comprensión gráfica de los sólidos de revolución propuestos en los ejercicios de la materia. Los estudiantes manifiestan, según las encuestas, que la mayor dificultad para ellos es, el visualizar el sólido de revolución en tres dimensiones, es decir requieren una gráfica que les permita mejorar su entendimiento en la asignatura. Una de las metodologías de enseñanza implica el uso de software educativo (Muelle, 2019), lo cual incluye herramientas tales como Matlab, ideal para plantear ecuaciones de nivel ingenieril y resolverlas fácilmente (Mathworks, nd). Para tales fines, se empleará este software en combinación con la Modelación Matemática para crear un programa que permita resolver problemas de sólidos de revolución, y especialmente, incluya la gráfica tridimensional.

Por otra parte, la formación docente frente a las TIC debe ser caracterizarse por ser constante en una sociedad que exige al profesor universitario, y escolar en general, cierta competencia ante ellas (Belandro-Montoro, 2014). Cuando el docente está bien capacitado en cuanto a las TIC, resulta natural implementar técnicas adecuadas como la que se propone en este trabajo.

El alcance del presente proyecto no contempla la capacitación docente, sino la utilización de Modelación Matemática a través de Matlab para mejorar el aprendizaje de los sólidos de revolución por parte de los estudiantes. Se parte de la premisa de que el cuerpo docente de la Escuela Superior Politécnica de Chimborazo, Facultad de Mecánica, está completamente capacitado en el uso de las TIC y se hace un enfoque más hacia los recursos estudiantiles.

1.2. Formulación del Problema

¿El cálculo de volúmenes de sólidos de revolución, mediante Modelación Matemática, permitirá mejorar el aprendizaje de estudiantes de la Escuela Superior Politécnica de Chimborazo, Facultad de Mecánica, ¿periodo abril-septiembre 2021?

1.3. Preguntas Directrices

Para dar una respuesta objetiva al problema de la investigación se ha definido las siguientes preguntas que definan la línea a seguir para la obtención de resultados:

- ¿Cuál es la metodología de enseñanza de sólidos de revolución y cuál es el nivel de aprendizaje obtenido por los estudiantes?
- ¿Existe alguna herramienta digital que utilice modelación matemática para permitir al docente enseñar sólidos de revolución?

1.4. Justificación de la Investigación

En un mundo en constante crecimiento y evolución es necesario plantear una alternativa didáctica en educación superior para el estudio de la matemática, utilizando las TIC. Es destacable el hecho de que la matemática y el cálculo son bases comunes de todas las ramas de ingeniería, y es preciso una correcta enseñanza de estas (Atllo, 2018; Belando-Montoro, 2014). La modelación matemática permitirá el surgimiento de nuevos programas que faciliten el estudio de dichos contenidos, en un entorno que resulte simple para los estudiantes.

Cuando se habla de la materia de cálculo, esta representa uno de los primeros problemas de percepción e interpretación con que los estudiantes se enfrentan, específicamente en el tema de sólidos de revolución. Esta situación es observable en el plan de estudios 2018, vigente actualmente, de la Facultad de Mecánica de la Escuela Superior Politécnica de Chimborazo.

Al tratar el tema de volumen en sólidos de revolución, las aplicaciones orientadas a mejorar el aprendizaje de esta ciencia son varias. Existen modelos matemáticos que son fácilmente aplicables en códigos de programación a través de Matlab, que permitirán a los estudiantes mejorar su rendimiento a través de la interpretación de situaciones problemáticas del entorno real en términos matemáticos.

En este trabajo se propone el uso de un programa para el cálculo de volúmenes de revolución en Matlab, fundamentado en la modelación matemática, que será bastante útil para el adecuado proceso de enseñanza de sólidos de revolución. Desde una perspectiva personal, si el proceso de enseñanza de matemática en las universidades no es el adecuado, el futuro ingeniero no tendrá las bases suficientes para afrontar futuras materias, ni solventarse en el mundo ingenieril. Es por eso por lo que se plantea un programa sencillo pero robusto, que permita desarrollar el cálculo de volúmenes de revolución, a través de la modelación matemática.

Con el de simplificar el proceso de enseñanza y mejorar el aprendizaje en los estudiantes, el presente trabajo empleará los modelos matemáticos correspondientes a sólidos de revolución para desarrollar una estrategia didáctica integral, que fomente el proceso de enseñanza-aprendizaje en el cálculo de los mismos. La ventaja del programa será que se pueden manipular los gráficos tridimensionales para mejorar la percepción de los estudiantes, además de ofrecer el proceso de resolución del ejercicio y su respuesta correcta.

Por todo lo expuesto se justifica el desarrollo de la investigación: “Cálculo de volúmenes de sólidos de revolución, mediante modelación matemática para mejorar el aprendizaje de los estudiantes de la Escuela Superior Politécnica de Chimborazo, Facultad de Mecánica, periodo abril-septiembre 2021”, por la conveniencia de desarrollar la modelización matemática como proceso didáctico de enseñanza, fortaleciendo la resolución de problemas, mostrando el aporte teórico y práctico que surge como consecuencia de la investigación

Delimitación del problema.

La investigación se desarrollará con los estudiantes de la Escuela Superior Politécnica de Chimborazo, Facultad de Mecánica, durante el periodo abril-septiembre del 2021.

1.5. Objetivos de la Investigación

1.5.1. Objetivo General

Calcular el volumen de sólidos de revolución, mediante modelación matemática para mejorar el aprendizaje de estudiantes de la Escuela Superior Politécnica de Chimborazo, Facultad de Mecánica, periodo abril-septiembre 2021.

1.5.2. Objetivos Específicos

- Realizar un estudio del nivel de aprendizaje alcanzado por los estudiantes en la construcción de volúmenes de sólidos de revolución.
- Diseñar una aplicación en Matlab para el cálculo de volúmenes de revolución, mediante modelación matemática.

- Validar la aplicación del modelo matemático implementado con los estudiantes la Facultad de Mecánica de la Escuela Superior Politécnica de Chimborazo en el período abril- septiembre 2021.

1.6. Hipótesis.

El cálculo de volúmenes de revolución, mediante modelación matemática, ¿mejora el rendimiento de los estudiantes de la Facultad de Mecánica de la Escuela Superior Politécnica de Chimborazo durante el periodo académico abril - septiembre 2021?

CAPÍTULO II

2. MARCO TEÓRICO

2.1. Antecedentes del Problema

Mendezabal y Tindowen (2018), mencionan que en varios casos las Tecnologías de la Información y las Comunicaciones (TIC) han permitido mejorar el rendimiento y actitud de los estudiantes en cálculo y matemáticas. En estos testimonios se utilizan programas ya existentes como Geogebra, Microsoft Mathematics, o simplemente proponen uno a través de la modelación matemática, de modo similar al caso del presente proyecto, para facilitar la enseñanza de materias complicadas para los estudiantes. Algunos de las cuales se describen a continuación.

Mejorar la actitud, la comprensión conceptual y las habilidades procedimentales de los estudiantes en cálculo diferencial a través de Microsoft Mathematics

En este estudio se analizaron los efectos del uso de Microsoft Mathematics en la actitud, comprensión conceptual y habilidades procedimentales de estudiantes de Cálculo Diferencial de la Facultad de Ingeniería Eléctrica de la universidad de San Luis, Filipinas. Se realizó una investigación cuasiexperimental para comparar dos entornos de estudio: uno donde los estudiantes recibían clases tradicionales, y otro en que recibían las mismas lecciones mediante las hojas de actividades de Microsoft Mathematics. Los grupos fueron aleatoriamente seleccionados, de 30 estudiantes cada uno. Al final, se demostró que aquellos estudiantes del segundo grupo obtuvieron una mejora significativa, y que su actitud mejoró, pues fue más dispuesta hacia aprender matemáticas con tecnología (Mendezabal & Tindowen, 2018).

Aplicación del software Geogebra y su influencia en el aprendizaje de áreas y volúmenes de sólidos de revolución en el Cálculo Integral en los estudiantes del primer año de la Facultad de Ingenierías de la Universidad Continental Arequipa – 2017

Este trabajo de investigación tuvo como objetivo determinar de qué manera la aplicación del software GeoGebra influía en el aprendizaje de áreas y volúmenes de sólidos de revolución en el Cálculo Integral en los estudiantes del primer año de la Facultad de Ingenierías de la Universidad Continental Arequipa – 2017 (Vilca, 2019). Se comparó el desempeño de los estudiantes antes de y luego de haber aplicado la enseñanza con Geogebra y tradicional, en dos grupos de 18 alumnos de dicha universidad peruana. Los estudiantes que recibieron enseñanza tradicional mejoraron el promedio de calificaciones de 10.67 a 12.89, mientras que el grupo que recibió clases con Geogebra obtuvo una mejora de 10.39 a 14.33. Por ello, se propuso realizar capacitaciones constantes a los docentes en cuanto al uso de Geogebra como recurso didáctico.

Propuesta metodológica para la enseñanza del Cálculo en Ingeniería, basada en la modelación matemática

En la Universidad de San Buenaventura, Colombia, se propuso una metodología para enseñanza de cálculo basada en la modelación matemática (Morales & Peña, 2013). Esta consistía en plantear cuatro modelos a ser reproducidos mediante funciones matemáticas (polinómicas, logarítmicas, u obtenidas por regresiones, integración, diferenciación, entre otras.) en diferentes plataformas educativas como Geogebra y Matlab. Los casos y temas que involucraban fueron: avión (precálculo), esfero-carro, logo del mundial (Cálculo Diferencial), y reproducción de rostros (Cálculo Integral). La propuesta tuvo que pasar por las fases de “justificación del proceso, elección del tema, desarrollo del contenido pragmático, ejemplos análogos-fijación de conceptos, y evaluación o convalidación de los resultados.” (Morales & Peña, 2013). Los autores mencionan que, con ello, se consiguió aportar de manera eficaz al desarrollo de la Matemática, “no solo porque mantuvieron el interés de los estudiantes durante todo el tiempo que se trabajaron, sino porque varios profesores que no hacen parte de esta investigación reconocieron su valor y han manifestado su interés en vincularse” (p. 548).

Aplicación del software libre Geogebra en el aprendizaje de Sólidos de Revolución en Cálculo Integral para los estudiantes de la Carrera de Pedagogía de las Ciencias Experimentales, Matemática y Física De La Universidad Central del Ecuador

Este trabajo de investigación fue desarrollado en la Carrera de Pedagogía de las Ciencias Experimentales Matemática y Física de la Facultad de Filosofía de la Universidad Central del Ecuador de la ciudad de Quito, en el cual Haro (2020) planteó analizar la influencia del software libre Geogebra Clásico 5.0 en la enseñanza de los sólidos de revolución, en los estudiantes del quinto semestre. Este trabajo reunió información mediante encuestas. Sus resultados dieron suficiente factibilidad para formular la propuesta de utilizar Geogebra como recurso adicional para un aprendizaje más eficaz en los estudiantes. Sin embargo, no se presentan resultados para comprobar la mejoría del aprendizaje de los estudiantes y la efectividad de la herramienta Geogebra.

Es por estos, y muchos más casos a modo de testimonio, que se puede depositar la confianza en la modelación matemática y sus aplicaciones para mejorar el entorno de enseñanza y aprendizaje en el mundo de la educación superior. Con el presente trabajo se pretende crear un aporte a nivel nacional, pues son pocos los estudios de esta naturaleza, en donde la modelación matemática juega un rol principal, además de permitir el acceso a la herramienta desarrollada por la presente autora.

En este capítulo se hace una revisión teórica respecto a los modelos matemáticos y su importancia sobre el cálculo de volúmenes de revolución, de manera que sea sencillo entender el funcionamiento de estos y su rol en la propuesta desarrollada posteriormente. Se parte de la exploración teórica acerca de sólidos de revolución, cómo se forman y qué métodos existen (y cuándo aplicarlos) para calcular su volumen mediante la aplicación de integrales. Además, se muestra una fundamentación teórica sobre modelos matemáticos y su implementación en Matlab, con el fin de comprender mejor las funciones que se utilizan en la propuesta desarrollada en este trabajo.

2.2. Bases Teórica

2.2.1. Introducción a los Sólidos de Revolución.

Para entender el tema de sólidos de revolución (una de las principales aplicaciones de las integrales definidas (CK-12, 2015), que se verán más adelante), primero es preciso revisar ciertos conceptos básicos:

2.2.2. Volumen.

Se conoce a volumen como a aquella magnitud métrica que define la extensión en tres dimensiones derivadas de la longitud, ancho y altura. Un sólido puede representarse por un volumen. Este espacio puede adoptar distintas formas, y la unidad principal de volumen es el metro cúbico (m³) (Sánchez, 2021). En la siguiente figura se muestran ejemplos de sólidos comunes y su volumen.

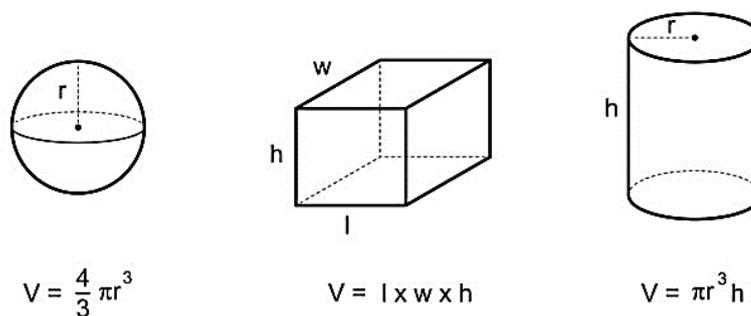


Figura 1-2: Volumen de sólidos simples (esfera, cubo y cilindro)

Fuente: (Editorial Etecé, 2021)

2.2.3. Integral o antiderivada.

Cuando el cuerpo posee una geometría compleja, se recurre a calcular su volumen mediante integrales (Leithold, 1998). Similar a lo que es la suma a la resta, o la multiplicación a la división, una integral o antiderivada, es la operación contraria a una derivada. La derivada permite realizar

el proceso de diferenciación (percibir diferencias en o entre), al determinar la tasa de cambio de una variable dependiente con respecto a una variable independiente y viene dada por la ecuación (Leithold, 1998; Chapra & Canale, 2007):

$$\frac{dy}{dx} = \lim_{\Delta x \rightarrow 0} \frac{f(x_i + \Delta x) - f(x_i)}{\Delta x} \quad (1)$$

Dónde, también puede denotarse como $f'(x)$ o y' . Matemáticamente, la derivada también es la pendiente a la curva en x_i (Chapra & Canale, 2007).

La operación inversa a la diferenciación es la integración. Su símbolo se muestra en la ecuación (2). El símbolo de la integral se parece a una letra S , estilizada antigua, que intenta representar la estrecha relación entre integración y suma (Chapra & Canale, 2007).

$$\int f(x) dx = F(x) + C \quad (2)$$

Dónde, la derivada de $F(x)$ es $f(x)$, como en la ec. (3)

$$\frac{d F(x)}{dx} = f(x) \quad (3)$$

2.2.4. *Integral definida e indefinida.*

La forma de la ec. (2) es conocida como la antiderivada general o indefinida, pues se desconoce la constante C . Existe una familia de antiderivadas con la misma función f que se deriva de ellas (Leithold, 1998).

Las integrales indefinidas o impropias son aquellas en que se desconoce el intervalo de integración y/o la continuidad de la función a integrar (QueGrande ORG, 2010). Existen tres tipos de integrales impropias:

- Primera especie: la función a integrar es acotada, pero se desconocen los límites de integración
- Segunda clase: la función a integrar no es acotada, pero se conocen los límites de integración
- Tercera clase: tanto la acotación de la función como los límites de integrar son desconocidos.

Para distinguir una función de entre una familia de curvas con la misma derivada, es preciso encontrar el valor numérico de la constante C , esto se logra mediante la aplicación condiciones iniciales o de frontera, respecto a F en un punto x conocido. La forma en que la constante C es conocida se denomina antiderivada particular o definida (Leithold, 1998).

2.2.4.1. Aplicaciones.

Las integrales definidas, también conocidas como integrales propias, permiten el cálculo del área bajo la curva, así como el volumen de sólidos de revolución (Leithold, 1998; Chapra & Canale, 2007; QueGrande ORG, 2010).

2.2.4.1.1. Cálculo de Áreas.

Si a una zona delimitada por una función continua dada, se le hacen divisiones pequeñas a modo de zonas rectangulares, el área bajo esta curva puede ser calculada mediante la aplicación de la Suma de Riemann, que representa la suma de todas las pequeñas parcelas realizadas. De este método nació la integral definida (Leithold, 1998; Espinoza, 2002). Este método se describe en la ecuación (4):

$$A \approx \sum_{i=1}^n f(w_i) (x_i - x_{i-1}) \quad (4)$$

La forma en que se realiza la discretización bajo la curva para aplicar la ec. (4), se aprecia en la Figura 2-2. Es decir, se realizan pequeñas divisiones en el área en interés y se suma el valor de cada pequeño rectángulo diferencial. Mientras más pequeña es la división, más exacta la respuesta. La desventaja es que el trabajo resulta muy costoso, pues, como menciona Leithold (1998), la suma de Riemann fue desarrollada en el siglo XIX, cuando las calculadoras no eran tan avanzadas y a medida que se reducía el tamaño de las divisiones, más tiempo se tardaban los cálculos. Luego, este método fue evolucionando hasta convertirse en una integral, cuyo símbolo se asemeja a una S y representa la “suma de los elementos diferenciales bajo la curva” (Leithold, 1998).

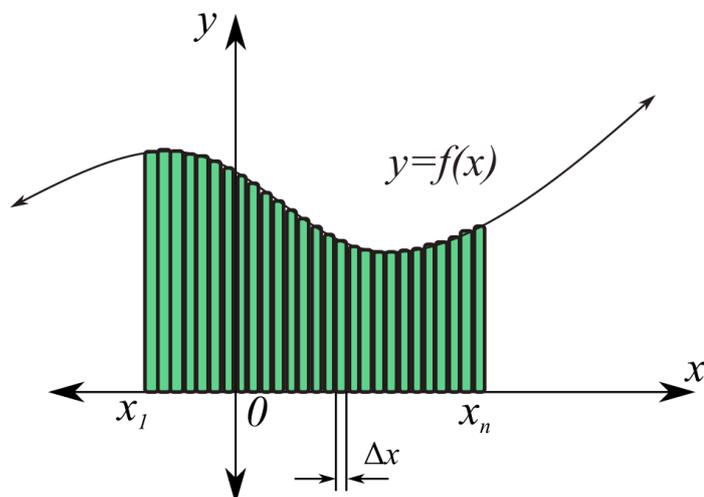


Figura 2-2: Discretización del área bajo la curva

Fuente: (Leithold, 1998, p. 333)

La notación de una integral definida en el intervalo $x = [a, b]$ es la mostrada en la ec. (5), de acuerdo con Leithold (1998), y es igual al límite de la suma de Riemann, cuando el intervalo de aplicación tiende a cero (Espinoza, 2002).

$$A = \int_a^b f(x) dx = \lim_{\Delta x \rightarrow 0} \sum_{i=1}^n f(\bar{x}_i) \Delta x \quad (5)$$

Dónde $\Delta x = (x_i - x_{i-1})$ y $\bar{x}_i = (x_i + x_{i-1})/2$. Entonces, una integral a la que se le ponen límites de integración se llama integral definida, y esta se refiere al área bajo la curva, en la región comprendida entre $x=a$ y $x=b$, como en la Figura 3-2 (Leithold, 1998).

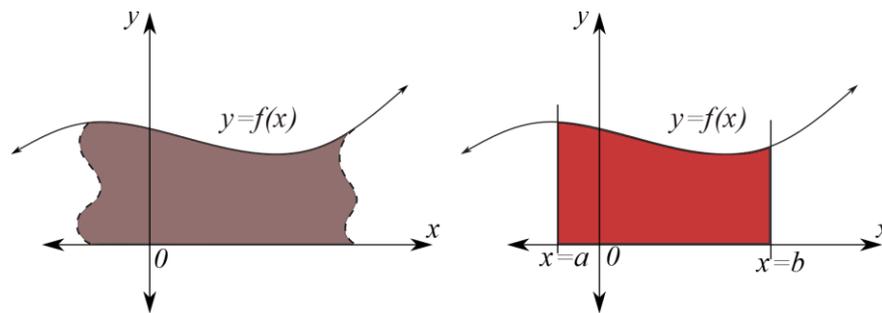


Figura 3-2: Área bajo la curva en una integral indefinida (izquierda) y definida (derecha)

Fuente: (Leithold, 1998, p. 333)

2.2.4.1.2. Cálculo de Volúmenes

Una vez determinados y calculada un área mediante la integral definida, esta puede hacerse girar en torno a un eje conocido para formar un cuerpo llamado sólido de revolución (Leithold, 1998). El eje de revolución puede ser el eje x , el eje y , o cualquier desplazamiento en $x=a$, o también en $y=a$, para lo cual se aplicará la integral por el método de integración adecuado, como se describe más adelante.

2.2.5. Sólidos de Revolución.

Un sólido de revolución es un cuerpo tridimensional que se obtiene al hacer girar una curva o función conocida en torno a un “eje de revolución” (Becerra, 2016; Leithold, 1998). Esta región viene limitada por un intervalo $[a, b]$, que se obtiene de una integral definida. Por ejemplo, al girar funciones elementales de una constante, una recta, y una semiesfera, se obtienen:

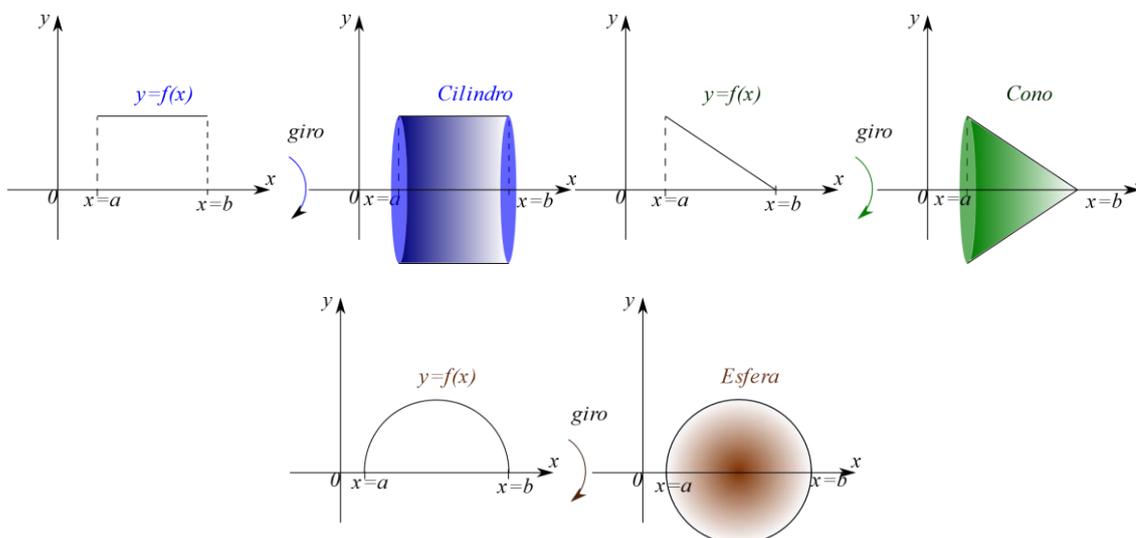


Figura 4-2: Sólidos de revolución conocidos: cilindro, cono, y esfera

Fuente: (Universidad Nacional Autónoma de México, s.f., p. 2)

El volumen del sólido de revolución es proporcional al área bajo la curva. Para encontrar dicho volumen es necesario conocer la integral definida entre $x = a$, y $x = b$, y entonces aplicar uno de los métodos descritos a continuación.

2.2.6. Métodos de cálculo de volúmenes de Sólidos de Revolución.

Cuando el sólido en estudio no es obtenido por una función elemental, como las vistas en la Figura 4-2, se recurre a la técnica de sólidos de revolución, que es una aplicación de las integrales definidas. La base para calcular el volumen de un sólido de revolución consiste en hallar el área perpendicular al eje de revolución, bajo una curva continua en $[a, b]$, y puede obtenerse a través de la sumatoria de la ec.(6) o mediante la integral (7) (Leithold, 1998):

$$V = \lim_{\|\Delta x\| \rightarrow 0} \sum_{i=1}^n A(\bar{x}_i) \Delta_i x \quad (6)$$

$$V = \int_a^b A(x) dx \quad (7)$$

Existen tres métodos para el cálculo de estos volúmenes de revolución:

- Método de Discos
- Método de Arandelas
- Método de Capas Cilíndricas

Los primeros dos métodos se conocen también como métodos de los anillos para encontrar el volumen de sólidos de revolución, en donde el volumen del sólido representa la suma de todas

las secciones perpendiculares al eje de revolución mientras que, en el último método, estas secciones son paralelas al eje (Leithold, 1998). Por ejemplo, la Figura 5-2, a continuación, representa un rectángulo infinitesimal que al ser rotado en torno al eje x , genera un disco perpendicular al eje dado.

2.2.6.1. Método de Discos.

Leithold (1998) menciona que el volumen V de un sólido obtenido al hacer girar alrededor del eje x una función f continua en el intervalo cerrado $[a, b]$, se obtiene a través de las ecuaciones (8)-(9):

$$V = \lim_{\|\Delta x\| \rightarrow 0} \sum_{i=1}^n \pi * [f(x)]^2 \Delta_i x \quad (8)$$

$$V = \pi \int_a^b [f(x)]^2 dx \quad (9)$$

Esto se aprecia en la Figura 5-2, donde el diferencial de área es perpendicular al eje de revolución. Cuando se aplica el método de discos y el eje de revolución es el eje x , la integral se hace respecto a un diferencial dx .

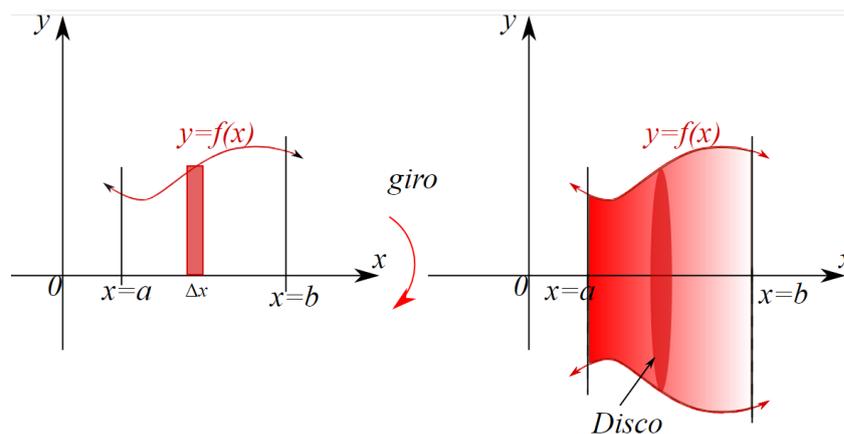


Figura 5-2: Esquema del sólido de revolución por discos

Fuente: (Leithold, 1998, p. 384)

Análogamente, si la rotación es respecto al eje y se sigue la ecuación (10):

$$V = \pi \int_a^b [f(y)]^2 dy \quad (10)$$

2.2.6.2. Método de Arandelas.

Dado el caso de que se requiera encontrar el área entre dos curvas, una de ellas puede ser constante, se aplica el método de arandelas. La región limitada por el intervalo $[a, b]$ entre las curvas $f(x)$ y $g(x)$, como en la siguiente figura, puede obtenerse al aplicar las ecuaciones (11)-(12), de acuerdo con Leithold (1998):

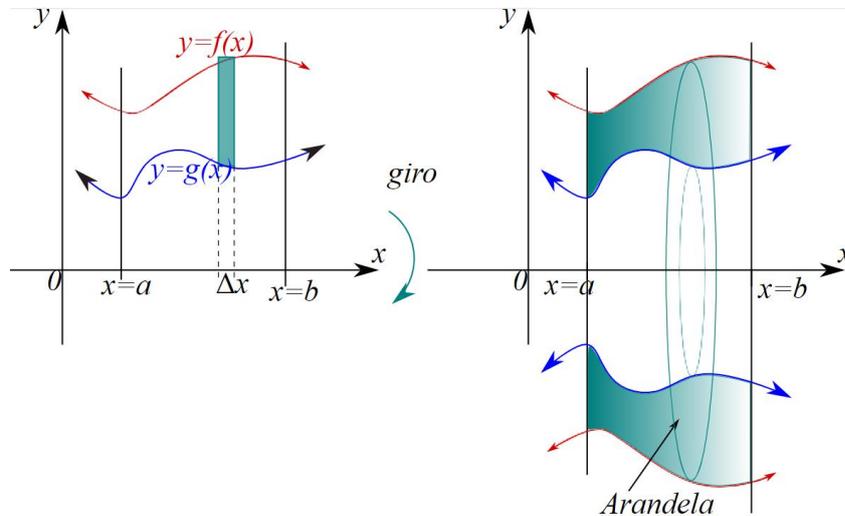


Figura 6-2: Esquema volumen de revolución por arandelas

Fuente: (Leithold, 1998, p. 386)

$$V = \lim_{\|\Delta x\| \rightarrow 0} \sum_{i=1}^n \pi * ([f(x)]^2 - [g(x)]^2) \Delta_i x \quad (11)$$

$$V = \pi \int_a^b ([f(x)]^2 - [g(x)]^2) dx \quad (12)$$

Este método es similar al de discos, pero la diferencia está en que existe un espacio vacío entre eje de revolución y la curva más cercana a él, por lo que el método de arandelas se puede aplicar cuando se tienen dos curvas conocidas, una de ellas puede ser constante, es decir, el área sobre la curva. Es decir, con este método el área en cuestión nunca toca el eje de revolución, sino se aplicaría el método de discos, así mismo el elemento diferencial de área es perpendicular al eje, de modo que se integra respecto a dx si el eje de revolución es el eje x .

Análogamente, para una rotación respecto del eje y , se aplica la ec. (13):

$$V = \pi \int_a^b ([f(y)]^2 - [g(y)]^2) dy \quad (13)$$

2.2.6.3. Método de Capas Cilíndricas.

Este método es una alternativa cuando la integral a resolver por los métodos de discos o arandelas resulta muy complicado, por ejemplo, la integral de una raíz cúbica (Leithold, 1998). La clave de este método radica en que el eje de revolución y el diferencial de área son paralelos entre sí, constituyendo un sólido de revolución en forma de capas cilíndricas (similar a una cebolla), como en la Figura 7-2:

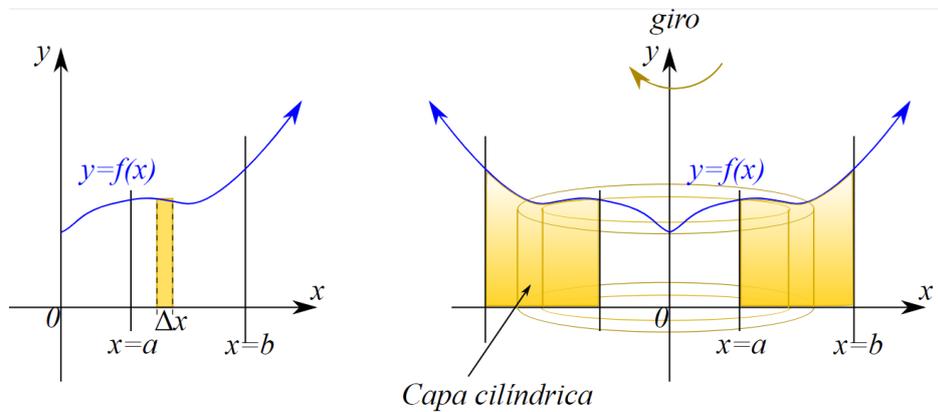


Figura 7-2: Esquema de sólido de revolución por capas cilíndricas

Fuente: (Leithold, 1998, p. 395)

Para encontrar el volumen de revolución por este método, se aplican las ec.(14)-(15):

$$V = \lim_{\|\Delta x\| \rightarrow 0} \sum_{i=1}^n 2\pi \bar{x}_i * f(\bar{x}_i) \Delta_i x \quad (14)$$

$$V = 2\pi \int_a^b x f(x) dx \quad (15)$$

Recordando que $\bar{x}_i = (x_i + x_{i-1})/2$. De acuerdo con Leithold (1998), en términos discretos, la ec. (15) representa $V= 2\pi$ (radio medio)(altura)(espesor). La ecuación anterior se utiliza para un sólido de revolución en torno al eje y. Si el sólido gira en torno al eje x, la integral se realiza respecto a un elemento diferencial dy siguiendo la ec. (16):

$$V = 2\pi \int_a^b y f(y) dy \quad (16)$$

2.2.7. Integración numérica.

La integración numérica es un método para resolver integrales definidas cuando estas no se pueden obtener por métodos analíticos (Chapra & Canale, 2007; Peña, 2015; QueGrande ORG,

2010). Este método tiene algunas fórmulas comunes: regla del trapecio, regla de Simpson 1/3 y 3/8, que son casos particulares de las Fórmulas de Newton-Cotes (Chapra & Canale, 2007).

2.2.8. Fórmulas de Newton-Cotes.

Estas fórmulas forman una estrategia que se basa en el uso de polinomios para aproximar datos tabulados o una función complicada de integrar. La ec. (17) muestra la fórmula asociada a ellas (Chapra & Canale, 2007):

$$I = \int_a^b f(x) dx \cong \int_a^b p_n(x) dx \quad (17)$$

Dónde el polinomio tiene la forma de la ec. (18):

$$p_n(x) = a_0 + a_1x + \dots + a_{n-1}x^{n-1} + a_nx^n \quad (18)$$

La aproximación puede venir dada por polinomios de cualquier grado n . El grado del polinomio puede aumentar, de manera que cada vez sea más exacta la aproximación respecto a los puntos a integrar. En la siguiente figura se muestra una aproximación de puntos de una curva con tres casos, primero un polinomio de grado cero, luego un polinomio de primer grado o función constante, y finalmente un polinomio de segundo grado o función cuadrática, donde el más exacto es el tercero, por ser el polinomio del grado superior entre los restantes (Chapra & Canale, 2007).

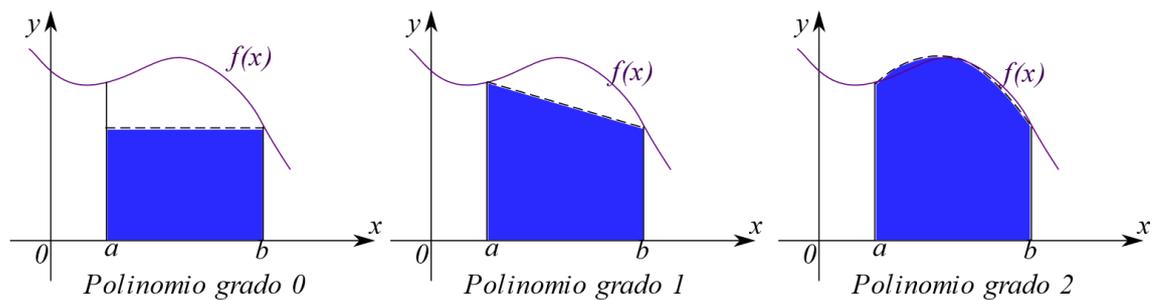


Figura 8-2: Bosquejo integración numérica, método Newton-Cotes

Fuente: (Peña, 2015, p. 35)

2.2.9. Regla del Trapecio.

Esta es una de las fórmulas (cerradas) de Newton-Cotes (Peña, 2015). Consiste en hacer divisiones en el área a integrar bajo la curva, con polinomios consecutivos de primer grado. Como en la figura:

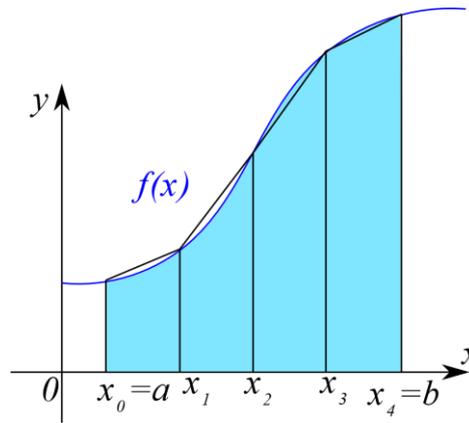


Figura 9-2: Bosquejo integración numérica, Regla del Trapecio

Fuente: (Chapra & Canale, 2007, p. 620)

En este método se realizan n divisiones, de forma que la longitud de cada intervalo es el de la ec. (19):

$$h = \frac{b - a}{n} \quad (19)$$

Al aplicar la integral en una línea recta, se obtiene:

$$I \cong \int_a^b f_1(x) dx \quad (20)$$

Dónde la función lineal puede escribirse de la forma:

$$f_1(x) = f(a) + \frac{f(b) - f(a)}{b - a}(x - a) \quad (21)$$

Entonces, al reemplazar (21) en (20) resulta:

$$I \cong \int_a^b f_1(x) dx = h * \frac{f(a) + f(b)}{2} \quad (22)$$

Que se denomina regla del trapecio, y es equivalente a decir que el área trapecio (I) es igual a su altura $h = (b - a)$ multiplicada por el promedio de las bases $(f(a) + f(b))/2$ (Chapra & Canale, 2007).

En la Figura 10-2 se muestra la diferencia que existe entre la aproximación de trapecios y la curva, que se conoce como error de truncamiento, y que viene dado por la ec. (23):

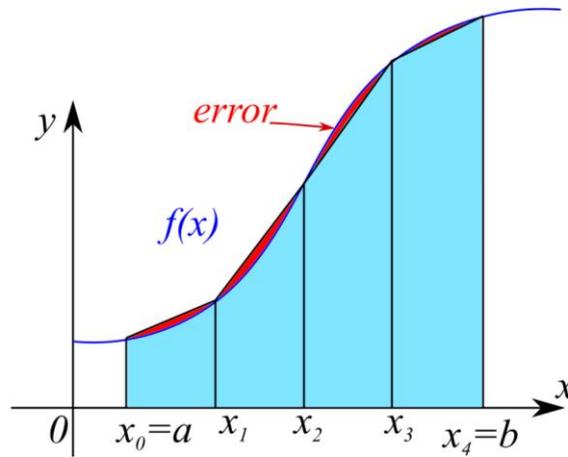


Figura 10-2: Error de truncamiento en la Regla del Trapecio

Fuente: (Chapra & Canale, 2007, p. 620)

$$E = -\frac{1}{12} f''(\xi)(h^3) \quad (23)$$

Dónde ξ representa la función entre a y b , de tal manera que, si la función a integrar es lineal, el error será nulo (Chapra & Canale, 2007). De manera que el área bajo la curva es igual a la sumatoria entre regla del trapecio y el error de truncamiento:

$$I = h * \frac{f(a) + f(b)}{2} - \frac{1}{12} f''(\xi)(h^3) \quad (24)$$

Cuando se aplica la Regla del Trapecio en n divisiones dentro de una curva (como en la Figura 10-2), se obtiene una expresión que sintetiza la suma del área de todas las divisiones trapezoidales realizadas (Chapra & Canale, 2007):

$$\begin{aligned} I &= \int_{x_0}^{x_1} f(x) dx + \int_{x_1}^{x_2} f(x) dx + \dots + \int_{x_{n-1}}^{x_n} f(x) dx \\ &\cong h \frac{f(x_0) + f(x_1)}{2} + h \frac{f(x_1) + f(x_2)}{2} + \dots + h \frac{f(x_{n-1}) + f(x_n)}{2} \end{aligned} \quad (25)$$

Qué, de forma simplificada, se expresa como:

$$I \cong \underbrace{(b-a)}_{\text{ancho}} \underbrace{\frac{f(x_0) + 2 \sum_{i=1}^{n-1} f(x_i) + f(x_n)}{2n}}_{\text{altura promedio}} \quad (26)$$

Y el error de truncamiento para este caso es:

$$E_t = -\frac{(b-a)^3}{12n^3} \sum_{i=1}^n f''(\xi_i) \quad (27)$$

$$\overline{f''} = \frac{\sum_{i=1}^n f''(\xi_i)}{n} \quad (28)$$

Dónde $\overline{f''}$ es el promedio de la segunda derivada de los polinomios de cada región trapezoidal:

$$E_t = -\frac{(b-a)^3}{12n^2} \overline{f''} \quad (29)$$

2.2.10. Reglas de Simpson.

Estas representan otra forma cerrada de integración numérica de Newton-Cotes (Peña, 2015). Trabaja con aproximaciones basadas en polinomios de segundo grado (Regla de Simpson 1/3), y de tercer grado (Regla de Simpson 3/8), en un número n de divisiones consecutivas del intervalo a integrar (Chapra & Canale, 2007).

2.2.10.1. Regla de Simpson 1/3.

Este método se basa en un polinomio de interpolación de segundo grado, donde se toma tres puntos por cada subintervalo, como en la Figura 11-2.

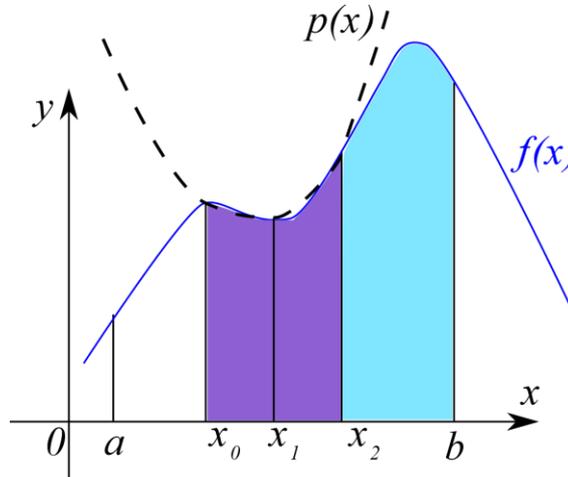


Figura 11-2: Bosquejo integración numérica, Regla de Simpson 1/3

Fuente: (Peña, 2015, p. 43)

De modo que la integral asociada a este método es el de la ec. (30) y $f_2(x)$ representa un polinomio de Lagrange de segundo grado:

$$I \cong \int_a^b f_2(x) dx \quad (30)$$

$$f_2(x) = \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)}f(x_0) + \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)}f(x_1) + \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}f(x_2) \quad (31)$$

$$I \cong \int_{x_0}^{x_2} f_2(x) dx = \frac{h}{3n} [f(x_0) + 4f(x_1) + f(x_2)] \quad (32)$$

Si el número de divisiones fuera $n=2$, habría un solo subintervalo, entonces $h = (b-a)/2$ y esta regla se podría usar mediante la ecuación (33):

$$I \cong \underbrace{(b-a)}_{\text{ancho}} * \underbrace{\frac{f(x_0) + 4f(x_1) + f(x_2)}{6}}_{\text{altura promedio}} \quad (33)$$

Dónde $a = x_0$, $b = x_2$ y $x_1 =$ punto medio entre a y b , es decir $x_1 = (a+b)/2$. El error de truncamiento relacionado con la aplicación de este método viene dado por la ec (34):

$$E = -\frac{1}{90} f^{(4)}(\xi)(h^5) \quad (34)$$

Es decir, la expresión completa se obtiene añadiendo a la regla del trapecio el error de truncamiento:

$$I = (b-a) * \frac{f(x_0) + 4f(x_1) + f(x_2)}{6} - \frac{1}{90} f^{(4)}(\xi)(h^5) \quad (35)$$

Sin embargo, la regla del trapecio de 1/3 mejora al aplicar n divisiones consecutivas en una curva, como en la Figura 12-2:

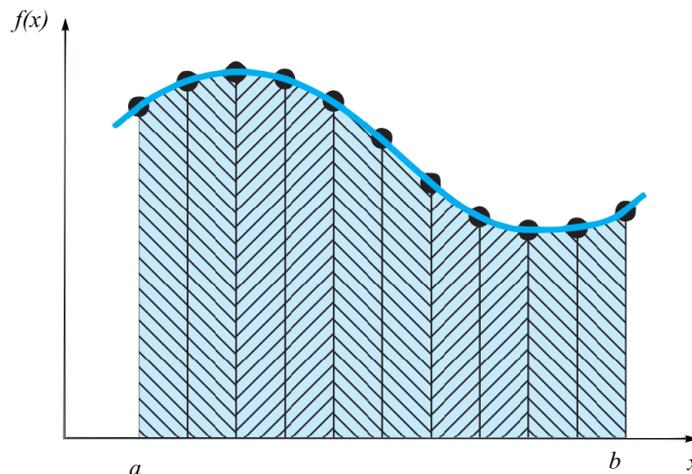


Figura 12-2: Regla de Simpson 1/3 para aplicación múltiple

Fuente: (Chapra & Canale, 2007, p. 634)

Dónde la integral total se puede expresar como:

$$\begin{aligned}
I &= \int_{x_0}^{x_2} f(x) dx + \int_{x_2}^{x_4} f(x) dx + \dots + \int_{x_{n-2}}^{x_n} f(x) dx \\
&\approx 2h \frac{f(x_0) + 4f(x_1) + f(x_2)}{6} + 2h \frac{f(x_2) + 4f(x_3) + f(x_4)}{6} + \dots \\
&\quad + 2h \frac{f(x_{n-2}) + 4f(x_{n-1}) + f(x_n)}{6}
\end{aligned} \tag{36}$$

Dónde $h = \frac{b-a}{n}$. La ec. (36) se puede simplificar hasta conseguir la ec. (37) (Chapra & Canale, 2007):

$$I \cong \underbrace{(b-a)}_{\text{ancho}} * \underbrace{\frac{f(x_0) + 4 \sum_{i=1,3,5\dots}^{n-1} f(x_i) + 2 \sum_{j=2,4,6\dots}^{n-2} f(x_j) + f(x_n)}{3n}}_{\text{peso promedio}} \tag{37}$$

De manera similar el error de truncamiento para n divisiones viene dado por la ec. (38), de acuerdo con Chapra y Canale (2007, p. 635). Donde $\bar{f}^{(4)}$ representa el promedio de la cuarta derivada de los polinomios de cada sub intervalo.

$$E_a = -\frac{(b-a)^5}{180n^4} \bar{f}^{(4)} \tag{38}$$

$$\bar{f}^{(4)} = \frac{\sum_{i=1}^n f^{(4)}(\xi_i)}{n} \tag{39}$$

2.2.10.2. Regla de Simpson 3/8.

Esta regla se distingue porque la aproximación de la integral se hace con un polinomio de Lagrange de tercer grado (Chapra & Canale, 2007). En esta aproximación, se toman cuatro puntos por cada subintervalo, como en la Figura 13-2, y se encuentra mediante la ec. (40):

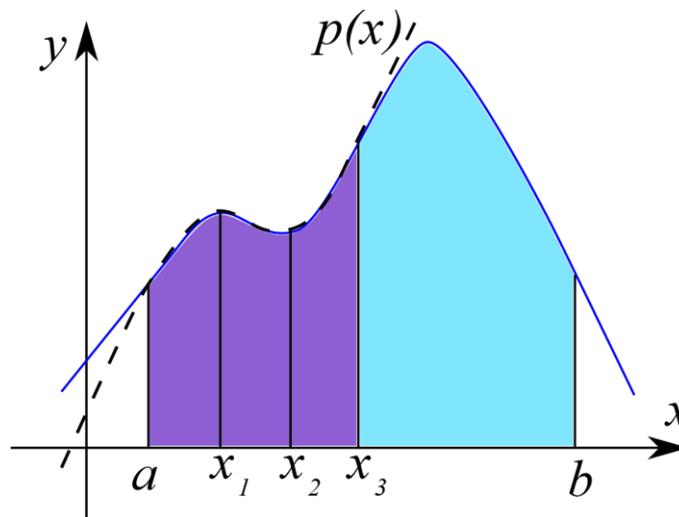


Figura 13-2: Bosquejo integración numérica, Regla de Simpson 3/8

Fuente: (Peña, 2015, p. 45)

$$I \cong \int_a^b f_3(x) dx \quad (40)$$

$$f_3(x) = \frac{(x-x_1)(x-x_2)(x-x_3)}{(x_0-x_1)(x_0-x_2)(x_0-x_3)}f(x_0) + \frac{(x-x_0)(x-x_2)(x-x_3)}{(x_1-x_0)(x_1-x_2)(x_1-x_3)}f(x_1) + \frac{(x-x_0)(x-x_1)(x-x_3)}{(x_2-x_0)(x_2-x_1)(x_2-x_3)}f(x_2) + \frac{(x-x_0)(x-x_1)(x-x_2)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)}f(x_3) \quad (41)$$

$$I \cong \frac{3h}{8} [f(x_0) + 3f(x_1) + 3f(x_2) + f(x_3)] \quad (42)$$

Debido al factor 3/8, esta regla se denomina Regla de Simpson 3/8 (Chapra & Canale, 2007, p. 636). Como $h = \frac{b-a}{3}$, la ec. (42) puede escribirse como:

$$I \cong \underbrace{(b-a)}_{\text{ancho}} * \underbrace{\frac{[f(x_0) + 3f(x_1) + 3f(x_2) + f(x_3)]}{8}}_{\text{altura promedio}} \quad (43)$$

El error asociado a este método está dado por (Chapra & Canale, 2007):

$$E = -\frac{3}{80} f^{(4)}(\xi)(h^5) \quad (44)$$

2.2.11. Resumen Fórmulas de Newton Cotes

En la Tabla 1-2 se muestra un resumen de las fórmulas de integración cerrada de Newton Cotes: Regla del Trapecio, Reglas de Simpson 1/3 y 3/8, además otra conocida como Regla de Boole, que equivale a la integración bajo un polinomio de Lagrange de cuarto grado (Chapra & Canale, 2007). Puede haber más métodos derivados de polinomios de orden superior, pero no son muy prácticos debido a que el grado de exactitud de las Reglas de Simpson basta, de acuerdo con los autores Chapra y Canale (2007, p. 639).

Tabla 1-2: Fórmulas de integración cerrada de Newton-Cotes

Método	Grado	Paso (h)	Error de truncamiento
Regla del Trapecio	1	$h = b - a$	$E = -\frac{1}{12}f''(\xi)(h^3)$ $= -\frac{(b-a)^3}{12}f''(\xi)$
Regla de Simpson 1/3	2	$h = \frac{b-a}{2}$	$E = -\frac{1}{90}f^{(4)}(\xi)(h^5)$ $= -\frac{(b-a)^5}{2880}f^{(4)}(\xi)$
Regla de Simpson 3/8	3	$h = \frac{b-a}{3}$	$E = -\frac{3}{80}f^{(4)}(\xi)(h^5)$ $= -\frac{(b-a)^5}{6480}f^{(4)}(\xi)$
Regla de Boole	4	$h = \frac{b-a}{4}$	$E = -\frac{8}{945}f^{(6)}(\xi)(h^7)$ $= -\frac{(b-a)^7}{483840}f^{(6)}(\xi)$

Realizado por: Bonilla, Jessica (2022)

Entre las Reglas de Simpson de 1/3 y la de 3/8, el mayor grado de exactitud lo tiene la de 3/8, pues el denominador del error de truncamiento es mayor (ver Tabla 1-2). Sin embargo, se prefiere trabajar con la Regla de Simpson de 1/3 por simplicidad debido a que “alcanza una exactitud de tercer orden con tres puntos en lugar de los cuatro requeridos con la versión 3/8” (Chapra & Canale, 2007, p. 636). Por otra parte, la Regla de Simpson de 3/8 es utilizada cuando el número de segmentos es impar, y puede aplicarse conjuntamente con su homólogo de 1/3 en un mismo ejercicio, como en la Figura 14-2:

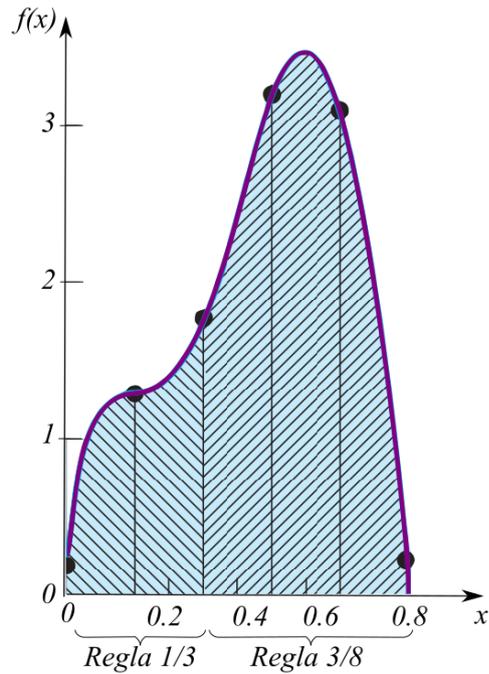


Figura 14-2: Regla de Simpson 1/3 y 3/8 combinadas para aplicaciones de intervalos impares

Fuente: (Chapra & Canale, 2007, p. 637)

2.2.12. Métodos avanzados de integración numérica

Cuando se requiere de mucha mayor exactitud que las Fórmulas de Newton Cotes, se recurre a métodos tales como la integración de Romberg o a la cuadratura de Gauss (Chapra & Canale, 2007, p. 640). Estos métodos se diferencian de los anteriores porque aquí sí se cuenta con la función y no hay que aproximarla mediante un polinomio, además, se puede hacer tantas divisiones como sea necesario hasta alcanzar un grado de exactitud aceptable (Chapra & Canale, 2007, p. 648).

2.2.13. Integración de Romberg

Este método combina dos estimaciones numéricas en la integral para obtener un valor más exacto, basándose en la regla del trapecio, y en la Extrapolación de Richardson, alcanzando mejores resultados. El nivel de tolerancia es especificado previamente a la aplicación, lo que vuelve a este método una técnica recursiva (Chapra & Canale, 2007).

Recordando que la regla del trapecio de aplicación múltiple (n divisiones) se encuentra sumando el valor de la integral y el error $I = I(h) + E(h)$, si se realizan por separado dos estimaciones de integración con dos tamaños de paso (h_1 y h_2), se tiene:

$$I(h_1) + E(h_1) = I(h_2) + E(h_2) \quad (45)$$

Al analizar la relación ente los errores por ambos pasos:

$$\frac{E(h_1)}{E(h_2)} \cong \frac{h_1^2}{h_2^2} \quad (46)$$

Dónde se eliminó el término \bar{f}'' y se puede utilizar el error “sin conocimiento previo de la segunda derivada de la función” (Chapra & Canale, 2007, p. 651). De esta última ecuación se despeja $E(h_1)$ y se reemplaza en la ec. (45), para obtener:

$$I(h_1) + E(h_2) \frac{h_1^2}{h_2^2} = I(h_2) + E(h_2) \quad (47)$$

De dónde se obtiene:

$$I \cong I(h_2) + \left(\frac{h_1^2}{h_2^2} - 1 \right)^{-1} * (I(h_2) - I(h_1)) \quad (48)$$

Si la relación ente los pozos es $h_2 = h_1/2$ se obtiene:

$$I \cong \frac{4}{3}I(h_2) - \frac{1}{3}I(h_1) \quad (49)$$

2.2.14. Cuadratura de Gauss

La Cuadratura de Gauss es un método que emplea valores en un intervalo $[a, b]$ sin restricción alguna (en las Reglas del Trapecio y de Simpson los valores de x son fijos, pues se ubican en el promedio de los subintervalos entre a y b). De esta forma se obtiene una estimación aún más exacta de la integral al existir la posibilidad de equilibrar los errores negativos y positivos del polinomio en la curva (Chapra & Canale, 2007, p. 456). Esto se aprecia en la figura:

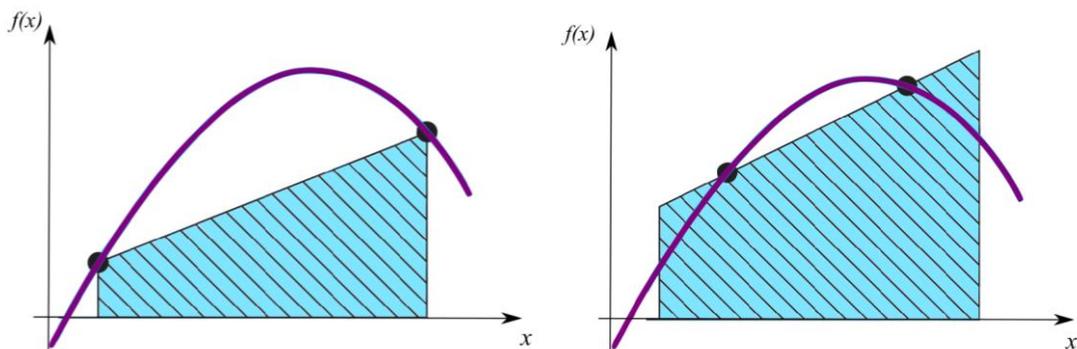


Figura 15-2: Comparación método de Newton-Cotes vs cuadratura de Gauss

Fuente: (Chapra & Canale, 2007, p. 656)

2.2.14.1. Método de coeficientes indeterminados

Tomando como referencia la regla del trapecio (ec. (22)), el área bajo una recta también puede darse por la ecuación:

$$I \cong c_0 f(a) + c_1 f(b) \quad (50)$$

Dónde las c son constantes. Además, para que la solución sea exacta, la ecuación a integrar debe ser una constante o una recta, en cualquiera de los casos se tendrá la ec. (68) o (69):

$$c_0 + c_1 = \int_{-(b-a)/2}^{(b-a)/2} 1 \, dx \quad (51)$$

$$-c_0 \frac{(b-a)}{2} + c_1 \frac{(b-a)}{2} = \int_{-(b-a)/2}^{(b-a)/2} x \, dx \quad (52)$$

Que, al resolver las integrales, y despejar las constantes, se obtiene, para ambos casos:

$$c_0 = c_1 = \frac{b-a}{2} \quad (53)$$

Y al reemplazar en la ec. (67):

$$I = \frac{b-a}{2} f(a) + \frac{b-a}{2} f(b) \quad (54)$$

Que es equivalente a la regla del trapecio.

2.2.14.2. Fórmula de Gauss-Legendre de dos puntos

Similar al caso anterior, se tiene:

$$I \cong c_0 f(x_0) + c_1 f(x_1) \quad (55)$$

Igualmente, los puntos x_0 y x_1 no son los extremos del trapecio, sino puntos intermedios, de modo que ahora se tienen cuatro incógnitas. En la siguiente figura se aprecia un esquema de la ubicación de x_0 y x_1 para este método.

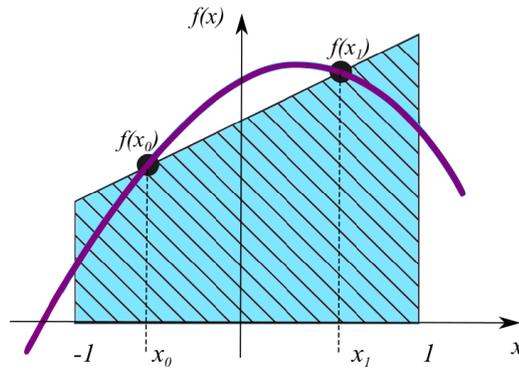


Figura 16-2: Esquema gráfico de la ubicación de las variables desconocidas x_0 y x_1 integración por la Regla de Gauss

Fuente: (Chapra & Canale, 2007, p. 658)

Para obtener las incógnitas, se asume cuatro funciones: constante, lineal, cuadrática y cúbica, que se justan a la integral, obteniendo:

$$c_0 f(x_0) + c_1 f(x_1) = \int_{-1}^1 1 \, dx = 2 \quad (56)$$

$$c_0 f(x_0) + c_1 f(x_1) = \int_{-1}^1 x \, dx = 0 \quad (57)$$

$$c_0 f(x_0) + c_1 f(x_1) = \int_{-1}^1 x^2 \, dx = \frac{2}{3} \quad (58)$$

$$c_0 f(x_0) + c_1 f(x_1) = \int_{-1}^1 x^3 \, dx = 0 \quad (59)$$

Estas ecuaciones se resuelven para encontrar:

$$c_0 = c_1 = 1$$

$$x_0 = -\frac{1}{\sqrt{3}} \approx -0.57735$$

$$x_1 = \frac{1}{\sqrt{3}} \approx 0.57735$$

Que se sustituyen en (50) y resulta:

$$I \cong f\left(-\frac{1}{\sqrt{3}}\right) + f\left(\frac{1}{\sqrt{3}}\right) \quad (60)$$

Al haber integrado entre $x = [-1,1]$ se consigue simplificar la matemática y que la formulación sea lo más general posible, aunque es posible hacer un cambio de variable para transformar otros límites de integración a esta forma (Chapra & Canale, 2007, p. 659).

2.2.14.3. Error de la cuadratura de Gauss

El error para este método se encuentra mediante la ec. (61):

$$E_t = \frac{2^{2n+3} * [(n + 1)!]^4}{(2n + 3)[(2n + 2)!]^3} * f^{(2n+2)}(\xi) \quad (61)$$

Dónde:

- n = número de puntos menos 1
- $f^{(2n+1)}(\xi)$ = derivada $(2n+2)$ -ésima de la función
- ξ = función con cambio de variable localizada entre -1 y 1

La cuadratura de Gauss es un método que tiene mucha mayor exactitud que las fórmulas de Newton-Cotes, siempre y cuando no se trabaje con polinomios de grado superior (Chapra & Canale, 2007).

2.2.15. Herramientas para el cálculo de volúmenes de Sólidos de Revolución.

Existen varias herramientas para calcular el volumen de sólidos de revolución, que hacen uso de las TIC, como los descritos a continuación.

2.2.15.1. Software educativo.

Este término designa todo componente no físico o intangible que forma parte de los dispositivos electrónicos, permite su funcionamiento y está compuesto por un conjunto de programas y aplicaciones diseñadas para cumplir con determinadas funciones dentro de un sistema. Además, está conformado por información perteneciente al usuario y datos de procesamiento (Agreda, 2021).

Para Gabriela Muenta (2019), el *software* educativo es un medio, el cual tiene el objetivo de facilitar tanto el proceso de enseñanza como el de aprendizaje, y a través de una plataforma digital facilita la adquisición de conocimientos en una colaboración con el estudiante. Gracias a su variedad, el *software* educativo permite capturar la atención de los alumnos y mantenerlos motivados e interesados por un período continuo.

El software educativo también es un conjunto de programas de enseñanza que utilizan plataformas digitales, como apoyo entre alumnos y docentes, en el que se aplica un proceso de

enseñanza/aprendizaje. Estos diseñan, potencian, y facilitan la adquisición de conocimientos exclusivamente académicos, lo que da paso al reforzamiento educativo de las diferentes materias curriculares, incluyendo prácticas en matemáticas o idiomas, que posibilitan el uso de cualquier materia o ámbito educativo, siendo una herramienta interactiva con capacidad de acople de acuerdo con la necesidad del alumno, incluyendo edad, curso escolar o capacidad individual (Rovira, 2019).

Algunos de los *softwares* educativos se diseñan para apoyar al docente. De manera que el profesor recurra a él para ofrecer sus lecciones o reforzarlas, otros tipos se orientan estrictamente al estudiante, dándole un entorno en el cual pueda aprender a su propio ritmo y manera. Este aspecto final es justo lo que se busca con el software educativo, ya que, en lugar de ofrecer lecciones sobre temas, pretende que el alumno refuerce sus conocimientos dentro de un entorno de aprendizaje, de modo que complemente o evalúe su educación (Pérez & Gardey, 2016).

2.2.15.2. *Categorías de software educativos.*

Cada software educativo tiene diferentes objetivos los cuales los plantea el educador para el alumno, que permite la interacción y aprendizaje, y pueden ser las siguientes:

a) Resolución de problemas.

El alumno mediante esta característica de software resuelve los problemas mediante hipótesis planteadas previamente, que cumple varios pasos, que supera las diferentes etapas que se acerca a los objetivos finales que parte desde un estigma, ejercitando la capacidad de memoria, atención y análisis (Farias, 2018).

De acuerdo con Muent (2019), el objetivo principal de este tipo de software es desarrollar la capacidad analítica del alumno, y para ello, el alumno deberá ser capaz de analizar las posibilidades, plantear hipótesis y seguir una serie de secuencias con el fin de resolver un problema en específico.

Algunos de los softwares para resolución de problemas son: MatlabSimulink, SciLab, donde se pretende resolver problemas “desde una perspectiva de teoría de control” (Rodríguez R., 2017, p. 75).

b) Ejercicio y práctica.

Es también llamado ejercicio interactivo, de posibilidades que emplea el desarrollo de conocimiento mediante el intercambio de información, el cual expone el tema para que el lector participe, adquiriendo conocimientos que se transfieren, respondiendo preguntas que eligen varias ofertas de opciones correspondientes que utiliza el software como una base de ejecución de

conocimiento previo de materia tratada que desarrolla el conocimiento de fases del tutorial, aplicación y retroalimentación (Rodríguez & Gairín, 2009).

Está compuesto de distintas lecciones, cada una de ellas con una parte teórica y otra práctica. En ese sentido, hay mayor interacción entre profesor y estudiantes, pues están estructurados para brindar lecciones, y luego evaluar el contenido aprendido. En el mundo digital, este método es conocido como de enseñanza tradicional didáctica. Se trata de uno de los softwares educativos más utilizados por profesores, ya que facilita y garantiza una constante evaluación y monitoreo de las acciones realizadas por los alumnos para obtener respuestas esperadas (Ramirez, 2019).

Ejemplos de software de esta categoría son las calculadoras gráficas, o programas específicos, como el software libre Tracker, que permite modelación de problemas de Física a través de videos interactivos (Rodríguez R., 2017).

c) Simulación.

Dispone de un modelo o entorno dinámico que contiene gráficos o animaciones interactivas que facilitan la exploración, y permite que los alumnos puedan modificar para poder aprender, de forma inductiva o deductiva, mediante la observación y manipulación de las estructuras presentes en el simulador. Contiene elementos de modelo interactivos que permiten tomar decisiones y adquirir experiencias directas delante de situaciones, ya sea de forma correcta o incorrecta en la toma de decisiones, esto posee un control central de tiempo pilotaje que va determinado de acuerdo al docente y muchos de los simuladores proponen videojuegos que le gustan a las personas y ayudan al logro de un aprendizaje por medio del entretenimiento (Pere, 2017).

Permite la transmisión del conocimiento por medio del descubrimiento, el cual depende del uso de quien lo emplea y tiene la sensación de acción o juego, que se aplica en caso de los niños. Es una buena manera de aprendizaje, atención y concentración, pues esto tiene una simulación aplicada con mucho ahínco en la formación de pilotos de aviones, o de conductores de vehículos de carga (Contreras & García, 2010).

Algunos ejemplos son, Vensim, y Stella, que permiten simulación dinámica de ecuaciones diferenciales (Rodríguez R., 2017).

d) Tutorial.

Su función es explicar un determinado tema sobre una base que tiene una cuestión en particular, el cual respeta para progresar la información de recursos complejos (Muyente, 2019). Define diferentes pasos como señala Fandos (2003):

- Introducción (que genera el primer contacto del tema e interés para abordarlo).

- Orientación.
- Aplicación en que se encuentran los datos del lector.
- Retroalimentación, donde se desarrolla los diferentes temas con resumen general del abordaje de conocimientos transferidos.

Esto se trata de que en mayor o menor medida supervisan el trabajo de los alumnos el cual tiene información mediante la realización de ciertas actividades previstas de antemano, pues los estudiantes ponen en juego determinadas capacidades aprendidas en la clase para un reforzamiento cognitivo. Además, desarrolla la capacidad por medio de ejercicios de refuerzo, proporcionando explicaciones conceptuales previas, que se denominan tutoriales de ejercitación, las cuáles desarrollarán la coordinación neuro-motriz en las actividades como el dibujo y la escritura, entre otras (Pere, 2017).

Dentro de esta categoría se encuentra el software Geogebra, que cuenta con un sinnúmero de tutoriales tanto en Cálculo, Probabilidad y Estadística, Geometría, entre otras (Geogebra, s.f.).

2.2.16. Modelos Matemáticos.

Un modelo matemático es una representación de la realidad, una expresión simplificada y generalizada de las características de un fenómeno, situación, objeto y sistema del mundo real. Representa una abstracción de la realidad, la cual se expresa con palabras, números, iconos, símbolos especiales, y se emplea para obtener “una imagen o representación de las diversas partes de la realidad, (...) se debe a que los seres humanos no apuntan a un conocimiento en cuanto a totalidad, lo cual resulta en una labor difícil de cumplir” (Carvajal, 2002, p. 2).

Al proceso mediante el cual se construye, resuelve y valida un modelo matemático se le conoce como modelización matemática. Este término es una contracción de modelización y educación, es decir, la suma de ambos. Por lo que, “convenimos en llamar modelación matemática al método de enseñanza- aprendizaje que utiliza el proceso de modelización en cursos regulares” (Herrera, 2012, p. 50).

En la modelación matemática se pretende “enfocarse en la realización y desarrollo de ciertas habilidades, en lugar de la transmisión de conocimientos puramente conceptuales” (Rodríguez R., 2017, p. 71). Esta modelación es considerada tanto como un dispositivo como un proceso académico y muestra ciertas ventajas (Plaza, 2016):

- Permite al estudiante comprender mejor el ambiente en que se desarrolla.
- Motivación: refuerza el aprendizaje de matemáticas.
- Estimula el desarrollo de determinadas actividades de carácter matemático.
- Contribuye a mejorar el punto de vista de las matemáticas.

La modelación matemática permite la aplicación de un proceso dinámico, que permite entender los problemas o situaciones de interés matemático, físico, químico, que están en una propuesta de un modelo. Este facilita comprender las situaciones reales que tiene relación con los objetos estudiados, que hacen uso de conceptos, técnicas, esencialmente matemáticas, convirtiéndolo en un método de enseñanza (Bassanezi & Biembengut, 1997). Esto constituye la representación o abstracción de la realidad. De acuerdo con Brito, Alemán, Fraguerra, Para, y Arias (2011) mencionan que los modelos pueden ser analógicos, físicos, gráficos, esquemáticos y matemáticos, y sus fases se representan en la Figura 17-2:

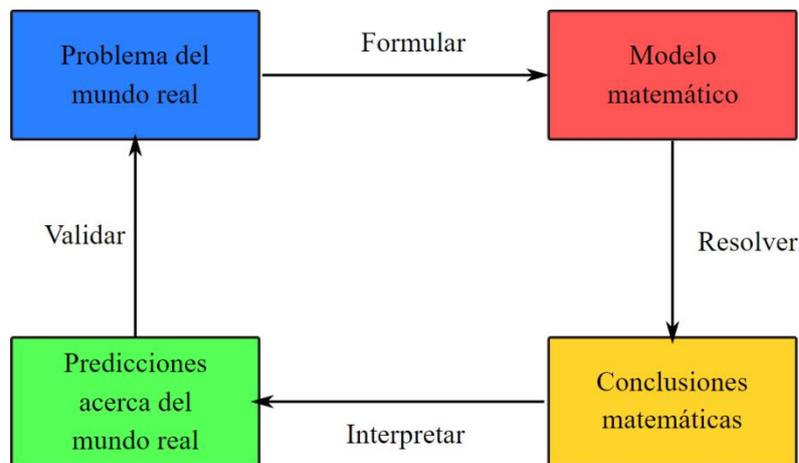


Figura 17-2: Proceso de modelado matemático

Fuente: (Brito et al., (2011, p. 130)

2.2.17. Conceptos clave.

Es necesario revisar ciertos conceptos clave para el desarrollo de un programa:

2.2.17.1. Cálculo.

Se refiere a los pasos mediante los cuales se obtiene el resultado de una operación, la misma que se halla mediante datos cuyo valor numérico puede ser o no conocido. El cálculo estudia la forma en que se resuelven problemas matemáticos luego de determinar las variables de una ecuación, en forma progresiva, aumentando cada uno de sus valores. Sirve para determinar curvas, pendientes, los valores mínimo y máximo de una función (Martínez A., 2021). Desde un enfoque en sólidos, permite la estimación de la magnitud mediante la longitud, área, volumen, permitiendo establecer el perímetro y volumen (Westreicher, 2021).

2.2.17.2. Algoritmo.

Se conoce como algoritmo a la secuencia de pasos a seguir para resolver un problema. Está formado por un conjunto finito y ordenado de operaciones para encontrar la solución al problema,

el mismo debe contener precisión, definitividad y finitud (Aranda, 2005). Este conjunto de reglas o instrucciones bien definidas, ordenadas y finitas, permite efectuar una actividad mediante pasos sucesivos, de modo que no se generen dudas a quien deba hacer dicha actividad, y funcionan sobre una vasta cantidad de estructuras de datos (SECUMATE, 2011).

2.2.18. Ejemplos.

Los modelos matemáticos hacen uso de funciones elementales para resolver problemas, como funciones lineales, cuadráticas, exponenciales, logarítmicas y trigonométricas. Además, puede valerse del cálculo diferencial (Bocco, 2010). Algunas aplicaciones de los modelos matemáticos permiten, por ejemplo, estimar la población mundial, determinar la transferencia de calor de un objeto al enfriarse al ambiente, o analizar una situación arbitraria, como se ven a continuación:

2.2.18.1. Ejemplo 1. Estimación de la población.

Un ejemplo clásico de la dinámica de sistemas es la estimación de la población en muchos niveles: mundial, regional, de consejos, y así sucesivamente, basándose en la tasa de natalidad, así como la de mortalidad, dada una población inicial. Esto puede modelarse fácilmente a través del software libre Vensim (Ventana Systems, Inc, 2007), con un diagrama de flujo similar al de la figura, en donde, luego de introducir los parámetros de natalidad, mortalidad y al darle clic en simular, el programa muestra la gráfica de estimación poblacional:

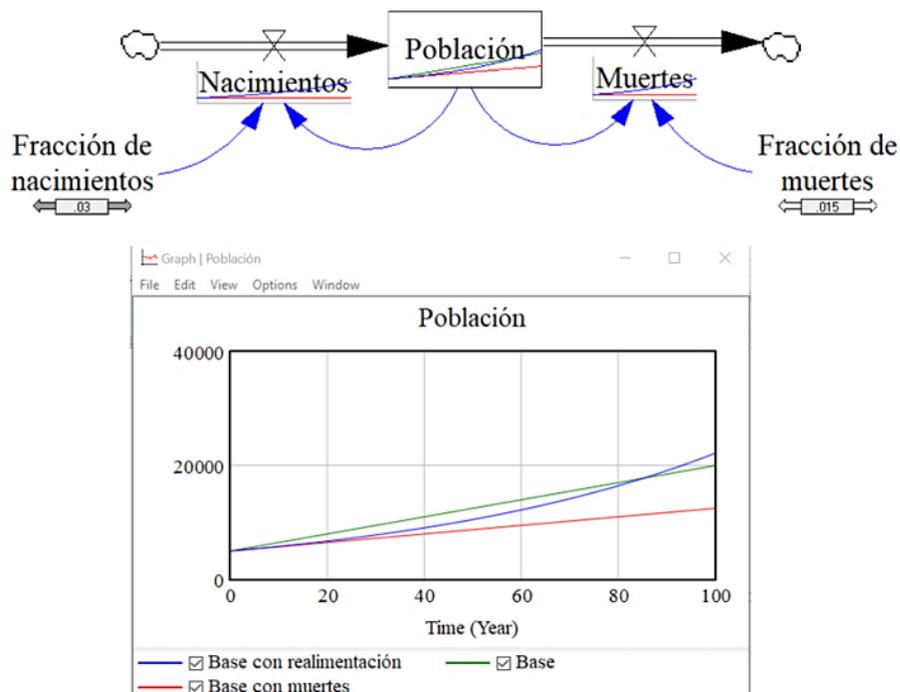


Figura 18-2: Diagrama de flujo y gráfica de la estimación de población regional, en Vensim

Fuente: (Universidad de los Andes (UniAndes), s.f., p. 8)

2.2.18.2. Ejemplo 2. Enfriamiento al ambiente de una taza de café.

Este modelo se conoce como Ley de Enfriamiento de Newton, y establece que la tasa o razón de enfriamiento es directamente proporcional a la diferencia de temperaturas entre la tasa y el ambiente. La ecuación (62) modela este fenómeno (Incropera & DeWitt, 1999):

$$\frac{dT}{dt} = -k(T - T_a) \quad (62)$$

Dónde:

T : temperatura de la tasa de café, K

t : tiempo, s

k : constante de transferencia de calor por convección entre la tasa y el ambiente, W/m²K

T_a : temperatura ambiente, K

Que, al ser integrada entre un tiempo inicial y un tiempo t , queda expresada como la ec. (63) (Incropera & DeWitt, 1999):

$$\int_{T_0}^T \frac{dT}{T - T_a} = -k \int_{t=0}^t dt \quad (63)$$

$$T = T_a + (T_0 - T_a)\exp(-kt)$$

En la figura, se aprecia un bosquejo de la Ley de Enfriamiento de Newton

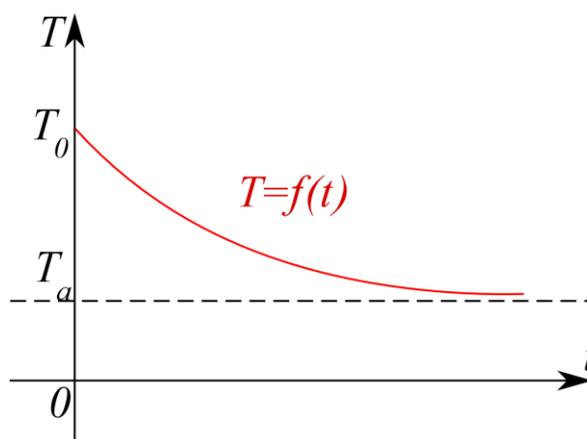


Figura 19-2: Bosquejo del Enfriamiento de Newton

Fuente: (Incropera & DeWitt, 1999, p. 231)

2.2.18.3. Ejemplo 3. Velocidad de un caminante.

En esta situación se plantea la velocidad con que se mueve una persona al caminar, por ejemplo: un sujeto se encontraba esperando un bus, cuando decide caminar hacia la parada de otra línea a causa de la espera muy alargada, entonces ve que el nuevo bus está por llegar y decide acelerar su paso. Es así cómo se plantea el modo en que esta persona varía su velocidad en función del largo de sus piernas, y la velocidad radial de su paso, hasta que alcanza un límite, y sin necesidad de correr alcanza al bus (Bocco, 2010; McNeill, 2021). La representación de este problema se vale de ecuaciones cuadráticas y trigonométricas, como en las ec. (64)-(65):

$$\frac{1}{2}m(v_{\theta}^2 - v_0^2) = mgl(1 - \cos\theta) \quad (64)$$

$$v_{\theta} = v_0^2 \left[1 + \left(\frac{2gl}{v_0^2} \right) (1 - \cos\theta) \right] \quad (65)$$

Dónde:

m : masa del caminante

v_{θ} : velocidad del caminante, m/s

g : aceleración de la gravedad, m/s²

l : largo de la pierna del caminante, m

θ : ángulo de las piernas respecto a un eje vertical imaginario, al caminar

En la siguiente figura, se muestra un bosquejo del caminante y los elementos de la ecuación anterior:

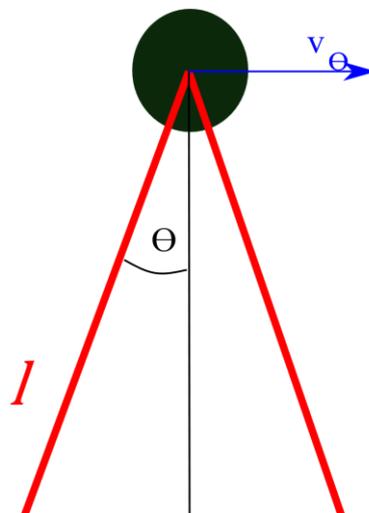


Figura 20-2: Bosquejo del ejemplo de velocidad de un caminante

Fuente: (McNeill, 2021, p. 1)

hacia tareas investigativas. Estas competencias de los alumnos pueden ser trabajadas mediante el uso de herramientas computacionales, en las cuales, la intervención del razonamiento es progresiva, aumentando la participación estudiantil. Así se permite al estudiante solventar con la tensión cognitiva que le produce el desconocimiento de cierto tema y lo soluciona al investigar, entender y aplicar su nuevo conocimiento aprendido.

2.2.20. Implementación de los Modelos Matemáticos en Matlab.

MATLAB, desarrollado por MathWorks, es un entorno de programación para el tratamiento de algoritmos, análisis de datos, visualización y cálculo numérico, tiene un alto rendimiento informático. Incluye matemáticas y computación, desarrollo de algoritmos, modelado, simulación, creación de prototipos, análisis, exploración, entre otras. Cuenta con un alto nivel de visualización de datos bidimensionales y tridimensionales, que procesa imágenes, animación y gráficos de presentación (PUCP, 2019).

Matlab cuenta con diferentes complementos, entornos visuales que facilitan la modelación matemática de sistemas de control, mecatrónicos, algoritmos de procesamiento de señales, entre otros. Algunos de ellos son Simulink, Stakeflow y Simscape y se encuentra disponible en sistemas operativos Unix, Windows, MacOS y GNU/Linux (Mathworks, nd).

Matlab tiene modelos de aplicaciones para desarrollo de problemas ingenieriles, llamados Modelos Determinísticos (Alfonso, De Armas, & Macías, s.f.). Por ejemplo, permite modelar el crecimiento de la población humana, de especies de animales, Sistemas Eléctricos, Mecánicos, con amortiguamiento, caída libre, y mucho más. También existen Modelos Estocásticos, relacionados con la generación de números aleatorios y el campo de la estadística, por ejemplo, la simulación del número de piezas en una línea de ensamble. Para implementar nuevos modelos matemáticos en Matlab, será necesario atravesar por las etapas de un proceso de modelación Matemática, formular, resolver, interpretar y validar (Brito et al., 2011) vistas anteriormente.

2.2.21. Programación en Matlab.

Matlab se caracteriza por trabajar con matrices, en especial, su significado es Laboratorio de Matrices, según su etimología (Alfonso, De Armas, & Macías, s.f., p. 56). Matlab basa su programación en librerías conocidas como *toolbox.*, que no solo se utilizan en el desarrollo de algoritmos, sino en campos de aeronáutica, automovilismo, ente otros. Para programar en Matlab, no es necesario declarar las variables, lo cual es una ventaja frente al uso de otros lenguajes de programación.

Los archivos en Matlab llevan la extensión *.m* y pueden ser escritos en cualquier editor de texto como Microsoft Word, Bloc de Notas, etc. Pero deben ser guardados con dicha extensión para

que Matlab pueda leerlos. Existen dos tipos de archivos que se pueden crear en Matlab (MathWorks, s.f.):

- *Script*, es un texto en el que se escribe un programa, contiene una serie de comandos y no requiere datos de entrada a menos que se llame a una función.
- *Function* (función), se emplea para reducir el número de líneas de programación en un *script*, y pueden ser usadas varias veces con solo llamarlas. Son similares a un *script*, pero se guardan por separado. El formato de una función es;

$function [y1, \dots, yN] = myfun(x1, \dots, xM)$

Dónde $y1, y2, \dots, yN$ son los argumentos de salida (resultados) al aplicar la función llamada *myfun* sobre las variables de entrada $x1, x2, \dots, xM$

Para llamar una función, se debe escribir su nombre y sus argumentos de entrada, en el mismo formato $myfun(x1, \dots, xM)$ con el que se creó. La función puede llamarse desde el *script*, o desde el *Command Window*, que es la ventana de trabajo de Matlab.

2.2.22. Integración en Matlab.

Para calcular integrales, Matlab dispone de algunas opciones, las mismas se basan en métodos numéricos y simbólicos, de donde es posible obtener la solución en forma analítica o tanto para integrales definidas como indefinidas. Los métodos de Matlab se basan en: cuadratura de Simpson, cuadratura de Lobatto, cuadratura de Gauss-Kronrod (MathWorks, s.f.).

2.2.22.1. Integración Numérica en Matlab.

Este método se utiliza para integrales definidas y se obtiene un resultado numérico, algunas de las funciones disponibles son (MathWorks, s.f.):

2.2.22.1.1. Método de Trapecios.

Para utilizar este método, se emplea la ec. (66):

$$Q = trapz(Y) \quad (66)$$

Dónde Y puede ser un vector, un a matriz o un arreglo multidimensional. Si Y es un vector, la solución será la integral de Y (representa los puntos que conforman una curva), entre el primer y último valor. Por ejemplo (MathWorks, s.f.):

$$Y = [1 \ 4 \ 9 \ 16 \ 25]$$

$$Q = trapz(Y)$$

$$Q = 42$$

Esto representa la integral de la función x^2 entre un intervalo de $[1, 5]$, con un espaciado de 1. Para cambiar este espaciado y obtener una integral mucho más exacta, se define un vector x que represente el dominio de integración dividido en las zonas necesarias. Por ejemplo:

$$X = 1:0.01:5$$

$$Y = X.^2$$

$$Q = \text{trapz}(X, Y)$$

$$Q = 41.33$$

Que es un resultado más exacto que el anterior, debido a que, en este último ejemplo, el vector x se encuentra dividido en 4000 partes, con un espaciado de 0.01.

2.2.22.1.2. Función Integral.

Este es un formato simple en donde se requiere definir solamente la función, y el intervalo de integración. Trabaja bajo el esquema de la ec. (67), de acuerdo con MathWorks (s.f.):

$$q = \text{integral}(\text{fun}, \text{xmin}, \text{xmax}) \quad (67)$$

Por ejemplo (MathWorks, s.f.):

$$\text{fun} = @(x) \exp(-x.^2) .* \log(x).^2;$$

$$q = \text{integral}(\text{fun}, 0, \text{Inf})$$

$$q = 1.9475$$

En dónde se definió la función y luego se integró entre $[0, \infty]$.

2.2.22.2. Integración Simbólica en Matlab.

Matlab cuenta con la ventaja de calcular la integral analíticamente, mediante variables simbólicas o *sym*. Esto permite obtener la respuesta como si se tratase de un ejercicio resuelto a mano, es decir, muestra el resultado en función de la variable x . La función que se utiliza para este cálculo es *int* y se puede utilizar tanto en integrales definidas como indefinidas (MathWorks, s.f.).

Para utilizar esta función, primero se define la variable mediante el comando *syms*, luego se ingresa la función, y finalmente se aplica mediante la ec. (68) como indica la guía de MathWorks (s.f.):

$$F = \mathit{int}(expr) \tag{68}$$

Dónde *expr* representa la función a integrar, por ejemplo:

$$\begin{aligned} & \mathit{syms} \ x \\ f &= 2 * \frac{x}{(1 + x^2)^2} \end{aligned}$$

$$F = \mathit{int} (f)$$

$$F = -\frac{1}{x^2 + 1}$$

Una de las ventajas que tiene este método es que no solo permite integrar funciones con una variable, sino, con varias, sin embargo, es preciso indicar respecto a qué variable se requiere integrar, como en la ec. (69) según MathWorks (s.f.).

$$F = \mathit{int}(expr, var) \tag{69}$$

Dónde *var* representa la variable sobre la cual integrar, por ejemplo:

$$\begin{aligned} & \mathit{syms} \ x \ z \\ f &= \frac{x}{1 + z^2} \\ F &= \mathit{int} (f, x) \\ F &= \frac{x^2}{2(z^2 + 1)} \end{aligned}$$

El mismo ejemplo, si se aplica la integración respecto a la variable *z* (MathWorks, s.f.):

$$F = \mathit{int} (f, z)$$

$$F = x \operatorname{atan} z$$

Para encontrar resultados de integración definida, la ec. (70) muestra el siguiente esquema en que se aplica (MathWorks, s.f.):

$$F = \mathit{int}(expr, [a, b]) \tag{70}$$

Dónde *a* y *b* representan los límites de integración, por ejemplo:

$$\mathit{syms} \ x$$

$$expr = x * \log(1 + x);$$

$$F = \text{int}(expr, [0 1])$$

$$F = \frac{1}{4}$$

De igual manera, es posible aplicar una integral definida a una función con más de una variable, pero especificando respecto a cuál de ellas se quiere integrar, como en la ec. (71) de acuerdo con MathWorks (s.f.):

$$F = \text{int}(expr, var, [a, b]) \quad (71)$$

Por ejemplo:

$$\text{syms } x \ t$$

$$fexp = \exp(x * t);$$

$$F = \text{int}(fexp, t, [0 2])$$

$$F = \frac{\exp(2x) - 1}{x}$$

Que resulta en una función respecto a la variable sobre la cual no se realizó la integral, pues ya se integró respecto a t .

2.2.23. Diseño de interfaces gráficas.

Una interfaz gráfica es una ventana que permite la interacción del usuario con el programa (Alfonso, De Armas, & Macías, s.f.). Es decir, a modo de videojuego. Están formadas de botones y gráficas en donde se hace clic y se realiza determinada operación para la que fue programada la interfaz. En Matlab, reciben el nombre de GUI (*Graphical User Interface*) y se crean en el ambiente de desarrollo de interfaces graficas GUIDE (*GUI Development Environment*) en las cuales primero “se escogen los controles, y luego se definen las acciones que cada uno va a realizar” (Alfonso, De Armas, & Macías, p. 80).

Más instrucciones sobre la interfaz GUIDE se encuentra en el manual de Barragán (2008). En resumen, los pasos para crear una interfaz gráfica son:

- Escribir *guide* en la ventana de comandos
- Elegir una de las siguientes opciones, la primera es una interfaz en blanco, las demás son prediseñadas para ahorrar tiempo al diseñarlas:

fig (contiene la interfaz gráfica de los botones creados); cada vez que se añada un nuevo elemento a la interfaz, va a generarse automáticamente un archivo *.m* (Barragán, 2008).

Para asignar una función a cada botón, se deberá editar el archivo *.m* como si se tratase de un *script* común y corriente. Para ejecutar el archivo el programa creado en la interfaz GUIDE, se debe abrir Matlab, y a continuación buscar el directorio donde se guardó el archivo. fig y abrirlo con un doble clic, con lo que el programa estará listo para usar en su interfaz, como en la figura.

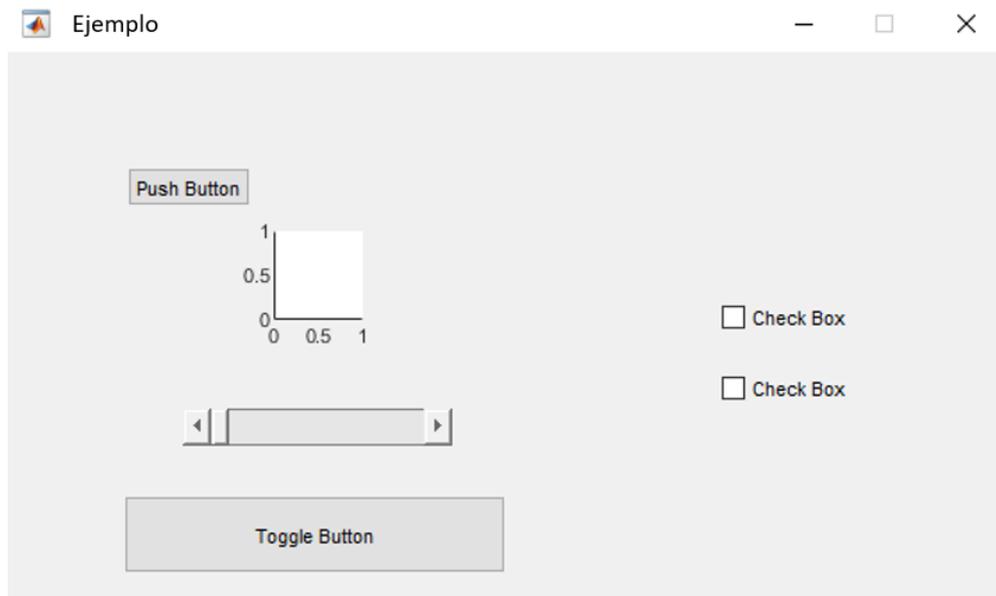


Figura 24-2: Ejemplo de interfaz gráfica ejecutada y lista para utilizar

Fuente: captura Matlab, elaboración propia

En esta figura se muestra una interfaz gráfica creada aleatoriamente para fines de ejemplificación, es decir, solo muestra el modo en que debería seleccionarse cada uno de los elementos para ser mostrados en la ventana. Para darle una función a cada botón, será necesario editar el archivo *.m* como se indicó previamente.

2.3. Marco conceptual

2.3.1. Metodologías de la enseñanza de Matemática Avanzada en el campo ingenieril.

Si bien, la tecnología continúa en constante avance, la presente autora considera que los métodos descritos a continuación siguen siendo la base primordial de cualquier metodología de enseñanza, incluyendo las TIC. Además, Martínez (1983) fórmula que “no somos partidarios de que se impongan métodos rígidos de enseñanza, ya que esto limitaría la libertad de cátedra. Sin embargo, algunas directrices generales deberán trazarse a fin de lograr cierta uniformidad en la enseñanza.”

En un entorno de Matemáticas Avanzadas para ingeniería, dos fuertes preguntas se formulan al programar actividades de enseñanza (Hernández, 1983):

- ¿Qué enseñar?
- ¿Cómo enseñar?

En cuanto a la primera interrogante, la respuesta es enseñar temas tales como Jacobianos, Series de Fourier, Transformada de Laplace, etc. es decir, temas propios y de gran peso para ingeniería (Hernández, 1983). Además, es importante proponer problemas de aplicación, cuyas fases son: traducción de determinada información física a una forma matemática, tratamiento del modelo mediante métodos matemáticos, e interpretación del resultado matemático en términos físicos.

Con respecto a la segunda pregunta, se tiene que emplear técnicas de enseñanza sistémicas, como los siguientes:

2.3.2. Métodos de instrucción individual.

2.3.2.1. Proyecto.

En este método cada estudiante trabaja en una tarea particular asignada y luego muestra un reporte con los resultados alcanzados. “El trabajo de proyectos se ha convertido en una vía prometedora para elevar los aprendizajes matemáticos de los estudiantes” (Aravena, Caamaño, & Joaquín, 2008). La planificación de proyectos consta de tres etapas primordiales:

- Diseño de la propuesta global.
- Métodos e instrumentos de análisis.
- Presentación de resultados.

2.3.2.2. Laboratorio.

Es una modificación del método anterior, implica el uso de instrumentos o dispositivos mecánicos para estudiar Matemáticas Aplicadas.

2.3.2.3. Heurístico.

El estudiante es continuamente dirigido (a diferencia de los métodos de proyecto y laboratorio). A través de preguntas cuidadosamente formuladas se busca entusiasmar a los estudiantes a pensar independiente y lógicamente (Hernández, 1983).

2.3.3. Métodos de instrucción global.

2.3.3.1. Genético.

Especial para desarrollo de material nuevo. La clase participa activamente en el desarrollo de preguntas formuladas por el profesor. Este método puede usarse antes o después de que los estudiantes hayan leído el texto correspondiente al tema nuevo, en su libro, según lo considere el docente.

2.3.3.2. Conferencia.

Exposición de ciertas fases de la materia en un discurso organizado, con el fin de ayudar a los estudiantes a comprender dicho material. Esta técnica requiere de un conocimiento a fondo de la materia, un desarrollo organizado, interpretación y aplicaciones (Hernández, 1983). En la cátedra se recomienda tomar notas para la mejor asimilación del contenido por parte del estudiante.

2.3.3.3. Ejercicio.

Consiste en la aplicación práctica de un proceso particular o concepto hasta que la reacción mental del estudiante sea inmediata y segura.

2.3.3.4. Pregunta-respuesta.

Similar al método Genético, este difiere en su propósito y ejecución. Consiste en la dinámica de hacer preguntas y dejar a un estudiante a la vez responderlas, si no se obtiene una respuesta correcta, se debe formular otra pregunta relacionada con la anterior que permita establecer la información necesaria como propiedad común de la clase (Hernández, 1983).

2.3.3.5. Estudio dirigido o supervisado.

El profesor dirige o supervisa el trabajo asignado a sus estudiantes. Este método sirve para que el estudiante progrese tanto como pueda y el profesor lo pueda ayudar cuando lo necesite.

En este capítulo se ha realizado una exhaustiva revisión teórica acerca de los modelos matemáticos necesarios para el cálculo de volúmenes de sólidos de revolución y su implementación en el entorno de Matlab, de modo que sea sencillo entender cómo este programa funciona, y se pueda proceder con el uso de este. De igual manera se vio qué es la modelación matemática y cómo permite resolver problemas, facilitando el aprendizaje de los estudiantes. A continuación, se muestra el capítulo de metodología, donde se explica el método utilizado para la recolección de datos, que serán de vital importancia para el diseño de la propuesta.

2.4. Identificación de variables

2.4.1. Variable Independiente.

Modelación matemática como estrategia didáctica en el cálculo del volumen de sólidos de revolución.

2.4.2. Variable Dependiente.

Rendimiento en los estudiantes de la Facultad de Mecánica

2.5. Operacionalización de variables

Tabla 2-2: Operacionalización de variables

VARIABLES	CONCEPTUALIZACIÓN	DIMENSIONES	INDICADORES	DEFINICIÓN DE LOS INDICADORES	CRITERIO DE MEDICIÓN	TÉCNICA	INSTRUMENTO	ESCALA
INDEPENDIENTE MODELACIÓN MATEMÁTICA COMO ESTRATEGIA DIDÁCTICA EN EL CÁLCULO DEL VOLUMEN EN SÓLIDOS DE REVOLUCIÓN.	Permite simular e interpretar diferentes problemas y situaciones de la vida real o académica, teniendo como evidencia diferentes condiciones de aplicación de los contenidos de los cursos de Matemática universitaria.	Didáctico	Genera material elaborado de aprendizaje	Comprender mejor el escenario en el que se desarrolla.	Número de estudiantes utilizando el software	Observación	Cuestionario	Nominal
		Interactivo	Intercambio de información entre los estudiantes y el sistema informático	Responden de forma inmediata las acciones de los estudiantes.	Número de estudiantes interactuando	Observación	Cuestionario	Nominal
		Evaluación	Cumplimiento de tareas	Rendimiento del estudiante.	Nivel de conocimiento en la ejecución de los exámenes	Observación	Cuestionario	Nominal
DEPENDIENTE RENDIMIENTO EN LOS ESTUDIANTES DE LA FACULTAD DE MECÁNICA	Expresión de capacidades y de características psicológicas del estudiante, desarrolladas y actualizadas a través del proceso de enseñanza-aprendizaje que le posibilita obtener un nivel de funcionamiento y logros académicos a través de un periodo lectivo, año o semestre.	Calidad	Resultados del aprendizaje previsto	Declaración de lo que el estudiante se espera que conozca, comprenda y sea capaz de hacer al finalizar un periodo de aprendizaje	Aprendizaje destacado	Prueba de evaluación	Cuestionario	Nominal
		Cumplimiento	Evidencia de las metas alcanzadas de aprendizaje	Capacidad de evaluar su propio proceso de aprendizaje con el propósito de hacerlo más consciente y eficiente, lo que hoy se resume con el lema de “aprender a aprender”.	Aprendizaje en el desarrollo	Prueba de evaluación	Cuestionario	Nominal
		Valores y eficiencia	Resultado del buen manejo de las tareas previstas	Demostraciones de los conocimientos valores y eficiencia en el proceso educativo y personal.	Aprendizaje logrado	Prueba de evaluación	Cuestionario	Nominal

Realizado por: Bonilla, J. 2021

2.6. Matriz de consistencia

Tabla 3-2: Matriz de Consistencia

FORMULACIÓN DEL PROBLEMA	OBJETIVO GENERAL	HIPÓTESIS	VARIABLES	INDICADORES	TÉCNICAS	INSTRUMENTOS
¿Se podrá mejorar el rendimiento de los estudiantes de primer semestre en el cálculo de volumen en sólidos de revolución mediante la modelación matemática como estrategia didáctica?	Realizar el cálculo de volúmenes de sólidos de revolución, mediante modelación matemática para mejorar el rendimiento de estudiantes de la Escuela Superior Politécnica de Chimborazo, Facultad de Mecánica, periodo abril-septiembre 2021.	El cálculo de volúmenes de sólidos de revolución, mediante modelación matemática mejorara el rendimiento de estudiantes de la Escuela Superior Politécnica de Chimborazo, Facultad de Mecánica, periodo abril-septiembre 2021.	V. Ind. Modelación matemática como estrategia didáctica en el cálculo del volumen en sólidos de revolución	Genera material elaborado de aprendizaje	Encuesta	Cuestionario
				Intercambio de información entre los estudiantes y el sistema informático		
				Cumplimiento de tareas		
			V. Dep. Rendimiento en los estudiantes de primer semestre de mecánica	Resultados del aprendizaje previsto	Prueba de Evaluación	Cuestionario
				Evidencia de las metas alcanzadas de aprendizaje		
				Resultado del buen manejo de las tareas previstas		

Realizado por: Bonilla, J. 2021

CAPÍTULO III

3. METODOLOGÍA DE INVESTIGACIÓN

3.1. Tipo y diseño de la investigación.

El tipo de investigación fue del tipo aplicada, lo que relaciona el estudio con la indagación teórica, porque depende de los resultados y de principalmente contener un marco teórico que dé solución a problemas prácticos. Para este caso, resultó imperativo implementar este tipo de investigación en consideración al objetivo general, que se refiere a la aplicación de un modelo matemático, de esta forma se garantiza el óptimo análisis e interpretación de los resultados para la aplicación del modelo.

Además, el diseño fue cuasi experimental, debido a que la asignación de los participantes de la investigación no fue aleatoria, sin embargo, existió una manipulación de la variable independiente para dar pleno uso del conocimiento adquirido en el área para dar solución al problema planteado.

3.2. Método de la investigación.

Se implementó el método cuantitativo, ya que durante el estudio se hizo uso principal de datos numéricos y el sistema estadístico de los resultados para comprobar las teorías de las variables: modelación matemática como estrategia didáctica y rendimiento en los estudiantes de primer semestre.

3.3. Estudio de la investigación.

Este estudio fue definido con corte transversal, ya que los datos e informaciones relevantes se recolectaron y analizaron en un periodo de tiempo específico para su respectiva medición, específicamente en el periodo del semestre abril-septiembre 2021 en la Escuela Superior Politécnica de Chimborazo.

3.4. Población de estudio.

La población estuvo conformada por 480 estudiantes del primer semestre de la Facultad de Mecánica de la Escuela Superior Politécnica de Chimborazo de la ciudad de Riobamba.

3.5. Selección de la Muestra.

Se utilizó un muestreo aleatorio para la selección de la muestra y se aplicó la encuesta a cada uno de ellos.

La población es finita por lo cual se procedió a obtener una muestra aplicando la siguiente fórmula:

$$n = \frac{N\sigma^2Z^2}{(N-1)e^2 + \sigma^2Z^2}$$

Dónde:

1. Población (N): 480 estudiantes.
2. Desviación estándar (σ): 0,5 (valor más confiable).
3. Nivel de confianza (Z): 95% probable correspondiente a 1,96.
4. Error muestral (e): 6% equivalente a 0,06.

$$n = \frac{480 * (0,5)^2 * (1,96)^2}{(479) * (0,06)^2 + (0,5)^2 * (1,96)^2}$$

$$n = \frac{460,99}{1,72 + 0,96} = \frac{460,99}{2,68}$$

$$n = 172,01 = 172$$

Es decir, la muestra de individuos seleccionados para aplicar los instrumentos de recolección de datos quedó conformada por un total de 172 alumnos.

3.6. Técnicas e instrumentos de recolección de datos.

Para obtener la información sobre el problema a investigar se usaron las técnicas e instrumentos de investigación siguientes:

3.6.1. Técnica.

Encuesta: Comprende un listado de preguntas previamente realizadas, con el objeto de recopilar información. Técnica que se aplicó a 172 estudiantes de primer semestre de la Facultad de Mecánica en el área de ciencias exactas.

3.6.2. Instrumento.

Cuestionario: Se redactó a partir de un conjunto de interrogantes orientadas a responder el contenido de las variables que ayudaron en la obtención de resultados, o en consiguiente, a direccionarlos correctamente.

CAPÍTULO IV

4. RESULTADOS Y DISCUSIÓN

En esta sección se contempla dos partes, una antes de la propuesta, para diagnosticar la situación inicial de los estudiantes, de donde se obtuvieron datos a través de encuestas; y otra después de la propuesta, para evaluar el nivel de aceptación obtenido por los alumnos, como se describe a continuación.

4.1. Situación previa.

4.1.1. *Análisis e interpretación de los resultados de la encuesta aplicada en la etapa de diagnóstico de la situación actual.*

A continuación, se efectúa el análisis y la interpretación de los resultados de la encuesta aplicada a la muestra establecida (172 estudiantes de la Facultad de Mecánica de la ESPOCH, en el período abril- septiembre 2021). La encuesta fue aplicada con el fin de analizar la situación previa a la aplicación de la propuesta, con el fin de evaluar el aprendizaje en el cálculo del volumen de sólidos de revolución. Los resultados son útiles para la elaboración de la propuesta de utilización de MATLAB como estrategia de aprendizaje en el cálculo de volúmenes de sólidos de revolución, mediante modelación matemática.

4.1.2. *Resultados de las preguntas específicas.*

En las siguientes tablas se muestran los resultados de la encuesta realizada a la muestra estudiantil, de donde se recabaron los datos de frecuencia absoluta (n_i), frecuencia absoluta acumulada (N_i), frecuencia relativa (f_i) y frecuencia relativa acumulada (F_i). Los estudiantes marcaron la opción que más se ajustaba a sus perspectivas personales.

1. ¿Cuál es su nivel de comprensión sobre cálculo de volúmenes de revolución?

Tabla 1-4: Nivel de comprensión sobre cálculo del volumen de sólidos de revolución

	n_i	N_i	f_i	F_i
Muy satisfactorio	12	12	6,8%	6,8%
Satisfactorio	70	82	40,7%	47,5%
Regular	70	152	40,7%	88,1%
Insuficiente	20	172	11,9%	100,0%
Total	172		100%	

Realizado por: Bonilla, J. 2022

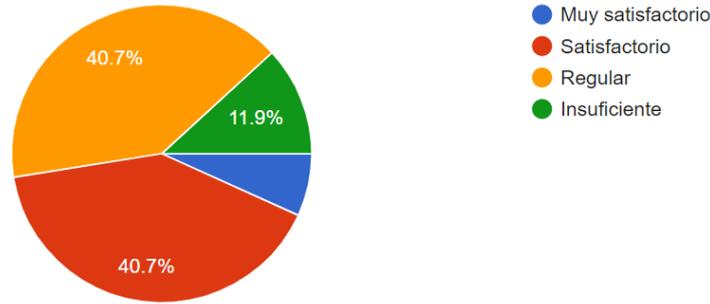


Gráfico 1-4: Nivel de comprensión sobre cálculo del volumen de sólidos de revolución

Realizado por: Bonilla, J. 2022

El 40,7% de estudiantes consideraron que su nivel de comprensión sobre cálculo de volúmenes de revolución les resulta satisfactorio, seguido por un 40,7% que consideran que su nivel de comprensión es regular, mientras que un pequeño porcentaje 6,8% manifiesta que es muy satisfactorio, lo que quiere decir que para la mayoría de los estudiantes el nivel de comprensión sobre cálculo de volúmenes de revolución les resulta bueno.

- ¿Qué tan frecuentemente hace uso de Herramientas Tecnológicas para comprender materia de cálculo?

Tabla 2-4: Uso de herramientas tecnológicas

	Ni	Ni	Fi	Fi
Frecuentemente	32	32	18,6%	18,6%
Casi siempre	64	96	37,3%	55,9%
A veces	76	172	44,1%	100,0%
Nunca	0	172	0,0%	100,0%
Total	172		100%	

Realizado por: Bonilla, J. 2022

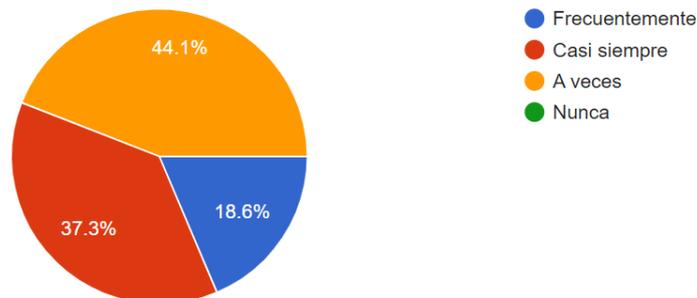


Gráfico 2-4: Uso de herramientas tecnológicas

Realizado por: Bonilla, J. 2022

El 44,1 % de los estudiantes encuestados manifestó hacer uso medianamente frecuente de herramientas tecnológicas para comprender la materia de cálculo, seguido del 37,3% con una frecuencia mayor de casi siempre, y a continuación un 18,6% que las utiliza con la mayor frecuencia. Esto significa que en torno a las dos terceras partes de los encuestados utilizan con alta frecuencia las herramientas tecnológicas, resultando en un buen estado respecto al uso de ellas.

3. ¿Posee alguna herramienta que le facilite el cálculo de volúmenes de revolución?

Tabla 3-4: Uso de herramientas para el cálculo del volumen de sólidos de revolución

	Ni	Ni	Fi	Fi
Sí		47	47	27,1%
No	125		172	72,9%
Total		172		100,0%

Realizado por: Bonilla, J. 2022

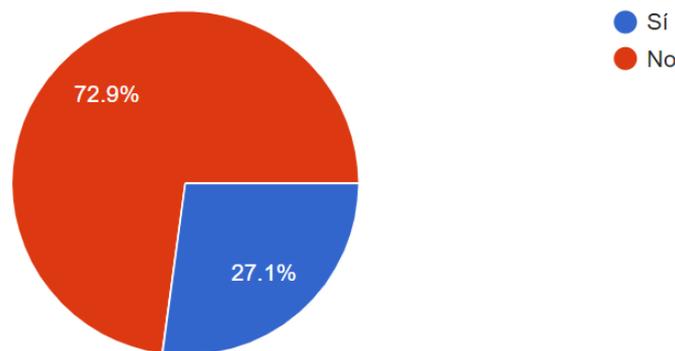


Gráfico 3-4: Uso de herramientas para el cálculo del volumen de sólidos de revolución

Realizado por: Bonilla, J. 2022

De acuerdo con los resultados, se apreció que el 72,9 % de los encuestados no cuentan con alguna herramienta que facilite el cálculo de volúmenes de revolución, mientras que el 27,1% manifiesta que sí posee una herramienta. Esto demostró un menor uso de herramientas de estudio para el cálculo de volúmenes lo cual pone en manifiesto, el bajo rendimiento de los estudiantes en la Facultad de Mecánica, a pesar de que en la pregunta 2 decían que sí usaban herramientas tecnológicas para aprender Cálculo.

4. De ser el caso, especifique cuál:¹

¹ En esta pregunta fue abierta, el total de encuestados fue 47 (quienes respondieron “sí” a la pregunta 3)

Tabla 4-4: Herramientas para el cálculo del volumen de sólidos de revolución

	Ni	Ni	Fi	Fi
Análisis matemático	6	6	12,5%	12,5%
Geogebra	26	32	56,3%	68,8%
Matlab	9	41	18,8%	87,5%
Solidworks	6	47	12,5%	100,0%
No aplica	0	47	0,0%	100,0%
Total	47		100,0%	

Realizado por: Bonilla, J. 2022

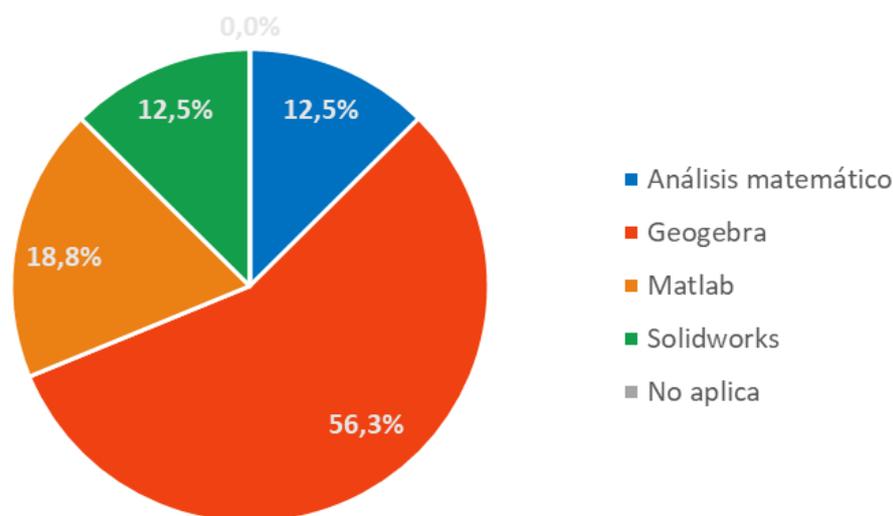


Gráfico 4-4: Herramientas para el cálculo del volumen de sólidos de revolución

Realizado por: Bonilla, J. 2022

Del grupo de 47 estudiantes que sí cuenta con alguna herramienta para el cálculo de volúmenes de revolución (ver pregunta anterior), el 56,3% dispone de Geogebra como soporte para resolución de ejercicios, mientras que el 18,8% está relacionado con MATLAB. Esto muestra que sí hay disponibilidad de programas para facilitar el cálculo de sólidos de revolución, siendo más popular Geogebra. De este pequeño grupo, según las encuestas, solo 32 estudiantes de entre los 47 que utilizan herramientas tecnológicas para sólidos de revolución, conocen sobre las facilidades que presentan Geogebra y Matlab para la resolución de problemas matemáticos y de Cálculo. Entonces se ve la necesidad de dar a conocer abiertamente las herramientas que favorezcan a los estudiantes el cálculo de volumen de revolución. En este caso, fomentar el uso y presentar el programa propuesto a todos los estudiantes posibles de la ESPOCH.

5. ¿Cuál considera que es la parte que más problema le causa al calcular volúmenes de revolución? ²

Tabla 5-4: Problemas al calcular de volúmenes de sólidos de revolución

	Ni	Ni	Fi	Fi
Definir límites	82	82	47,5%	47,5%
Visualizar la revolución en 3D	93	175	54,2%	101,7%
El proceso en general	47	222	27,1%	128,8%
La parte algebraica	35	257	20,3%	149,2%
Total	257		149%	

Realizado por: Bonilla, J. 2022

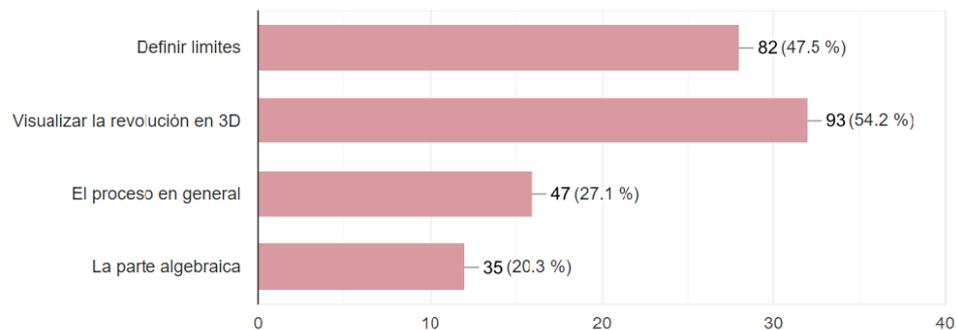


Gráfico 5-4: Problemas al calcular de volúmenes de sólidos de revolución

Realizado por: Bonilla, J. 2022

Respecto a cuál considera que es la parte que más problema le causa al calcular volúmenes de revolución, se encontró que el 54,2% menciona que visualizar la revolución en 3D es el lado que trae inconvenientes a la hora de calcular volúmenes, seguido del 47,5% definir límites, el 27,1% el proceso en general y un 20,3% de los encuestados piensa que la parte algebraica es la más difícil de dominar. Estos resultados indicaron que en su mayoría los estudiantes tienen inconvenientes para resolver los cálculos por problemas de percepción tridimensional del ejercicio. Cabe mencionar que esta pregunta fue de opción múltiple, razón por la cual la suma de los porcentajes supera el 100%.

² Esta pregunta fue de opción múltiple, razón por la cual las frecuencias acumuladas superan el 100%

4.1.3. Conclusión sobre los resultados obtenidos en la encuesta aplicada.

Respecto a la situación previa a la implementación de la propuesta y, en base a los resultados obtenidos y al análisis realizado en cada una de las preguntas, se concluye que es necesario estructurar propuesta didáctica para permitir al estudiante una participación frecuente y activa, que utilice una teoría del aprendizaje y se centre en el alumno. Esto debe conducir al desarrollo del análisis y la comprensión del cálculo de volúmenes de sólidos de revolución mediante modelación matemática. Así mismo, se analizó la existencia de una fuerte disposición por parte de los estudiantes para utilizar un programa con frecuencia, el cual debe cumplir con las condiciones de facilidad de uso, y aplicable en computadoras o celulares. Es necesario el acceso a internet, puesto que la propuesta debe emplear un programa gratuito y de libre acceso. Por las razones expuestas, la propuesta didáctica utilizará la teoría conocida como constructivista y el programa que a continuación se detalla.

4.2. Situación posterior a la propuesta.

En esta parte se muestran los resultados obtenidos luego de aplicar la propuesta como metodología de enseñanza a la muestra de estudiantes antes señalada. Con esto se pretende evidenciar las ventajas que tiene usar el programa propuesto, así como validarlo, pues va a permitir a los estudiantes mejorar su rendimiento significativamente, así como su actitud hacia la materia de Cálculo, en el tema de sólidos de revolución.

4.2.1. Actitud de los estudiantes.

1. ¿Qué tan útil considera la interfaz gráfica del programa en MATLAB presentada con usted?

Tabla 6-4: Utilidad de la interfaz gráfica del programa en MATLAB

	Ni	Ni	fi	Fi
Muy satisfactorio	50	50	28,8%	28,8%
Satisfactorio	113	163	66,1%	94,9%
Poco satisfactorio	9	172	5,1%	100,0%
Irrelevante	0	172	0,0%	100,0%
Total	172		100%	

Realizado por: Bonilla, J. 2022

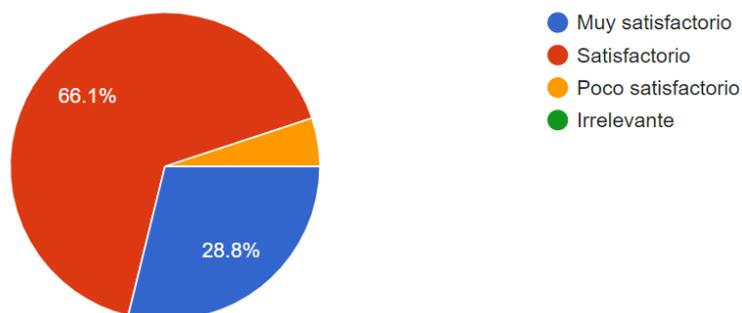


Gráfico 6-4: Herramientas para el cálculo del volumen de sólidos de revolución

Realizado por: Bonilla, J. 2022

La mayor parte de los estudiantes encuestados (66,1%) expresaron que es satisfactorio la utilidad de la interfaz gráfica, mientras que el 28,8% la consideró muy satisfactorio, lo que significa que en conjunto casi la totalidad de ellos (95% de los encuestados) aprueba la propuesta para su utilización en el proceso de enseñanza y aprendizaje de sólidos de revolución, lo cual es muestra de validez de la misma.

2. Considera que, ¿su nivel de comprensión en materia de cálculo ha mejorado después de haber utilizado la aplicación?

Tabla 7-4: Nivel de comprensión en la materia de Cálculo Integral

	Ni	Ni	Fi	Fi
Sí	146	146	84,7%	84,7%
No	26	172	15,3%	100,0%
Total	172		100,0%	

Realizado por: Bonilla, J. 2022

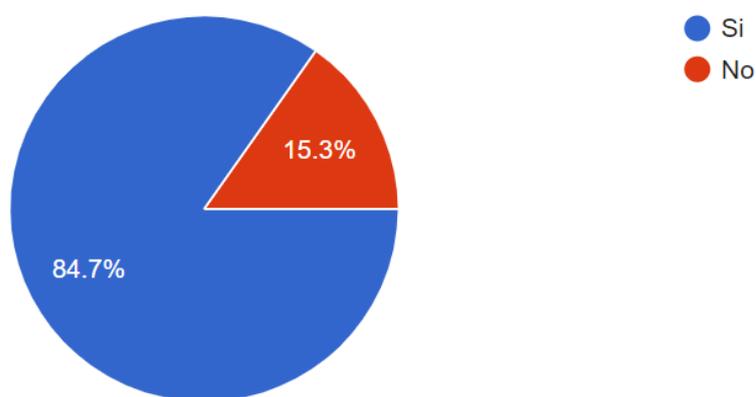


Gráfico 7-4: Nivel de comprensión en la materia de cálculo integral

Realizado por: Bonilla, J. 2022

El 84,7% de los estudiantes manifestaron que sí ha mejorado su nivel de comprensión en materia de cálculo, lo que significa que los alumnos de la Facultad de Mecánica de la ESPOCH están avanzando en su desempeño académico y en su mayoría entienden la materia.

3. ¿Qué mejoraría en el programa presentado?

En esta parte, la pregunta fue de tipo abierta a los estudiantes, por lo que pudieron escribir su propia respuesta, de las cuales se enlistan algunas:

- “Todo está comprensible”
- “Presentación de las respuestas”
- “Nada le veo muy completo”
- “La capacidad de manipular las gráficas para verlas desde distintos ángulos”
- “Me parece que todo está súper bien, por lo tanto, no cambiaría nada”
- “Resultados detallados”
- “Me parece bastante completo, tal se podría agregar títulos a las gráficas solo por cuestiones de visualización”
- “El video es interesante, pero sería de hacerlo de una manera más explicativa y entendible para poder lograr un buen uso del software”
- “De momento está muy entendible y fácil de usar el programa”
- “Elegir autónomamente el método de resolución”
- “A mi opinión el programa me parece optimo además y tiene una interfaz comprensible”
- “Que sea un poco más sencillo de usar”
- “Me encuentro cómodo con el programa”
- “Más videos tutoriales”

La mayoría de los estudiantes encuestados manifestaron que, en el programa presentado, se mejoren las ilustraciones. En general, estos resultados, nos indica que, en su mayoría, el programa es aceptado, y lo más importante es que, es de gran ayuda para mejorar el rendimiento académico de los estudiantes de la Facultad de Mecánica.

4.2.2. Conclusión sobre los resultados obtenidos luego de aplicar la propuesta.

Los estudiantes, luego de probar el programa propuesto en Matlab, demostraron su aceptación casi total respecto a la misma, en torno al 95% de ellos la calificaron como satisfactoria y muy

satisfactoria. Además, el 84,7% considera que su nivel de aprendizaje ha mejorado luego de haber utilizado el programa respecto a sólidos de revolución. Entre las principales falencias que le encuentran al programa están: mejorar las ilustraciones de modo que se pueda manipular en diferentes ángulos, así como añadir más videos tutoriales para utilizarlo.

Por todo lo anterior se concluye que la actitud de los estudiantes respecto al aprendizaje de cálculo ha mejorado considerablemente. Es decir, ahora ellos se encuentran más dispuestos a aprender mediante el uso del programa propuesto, así como a seguir aprendiendo sobre el cálculo de volúmenes de sólidos de revolución. No solo eso, sino que también se ha probado que el programa y su rendimiento experimentó mejoras para el ciclo seleccionado (abril-septiembre 2021).

CAPÍTULO V

5. PROPUESTA

Por las encuestas realizadas en la Situación Previa se vio la factibilidad para utilizar la modelación matemática como complemento de la enseñanza de sólidos de revolución, y aplicar esta propuesta dentro del plan de estudios de la Facultad de Mecánica en los estudiantes de primer semestre de Ingeniería en la ESPOCH, en el período abril-septiembre 2021. Esta alternativa didáctica permitirá al estudiante verse motivado al constante aprendizaje y mejorar su rendimiento en el área de Cálculo.

5.1. Introducción.

El objetivo de la presente propuesta es desarrollar en los estudiantes habilidades y destrezas que ayuden a mejorar su capacidad de razonar, analizar y comprender el cálculo del volumen de sólidos de revolución, mediante el uso del programa creado en MATLAB. En especial, se pretende que los alumnos mejoren su percepción gráfica al resolver los ejercicios, pues en la encuesta se mostró que este es el principal deficiente para su rendimiento.

La propuesta se desarrolla en Matlab, aprovechando que el uso de este software también se imparte en clases de ingeniería de la ESPOCH. El programa se basa en modelos matemáticos de cálculo de volumen de sólidos de revolución tales como método de discos y método de arandelas, vistas en el Cap. 2.2. Estos permiten que la modelación matemática sea correcta, pues son modelos válidos. El programa permitirá optimizar el tiempo, con lo cual es factible el desarrollo de más ejercicios en horas libres de los estudiantes, así como su uso a modo de auto aprendizaje y complemento de enseñanza.

Lo que se pretende con este programa es reforzar las explicaciones del profesor, pero no sustituirlas. El docente, después de haber explicado la base teórica junto con la resolución de ejercicios, utilizará Matlab para corroborar lo explicado, dando cabida al análisis y discusión, dinamizando la clase y permitiendo a los estudiantes encontrarse más motivados. Así mismo, se podrá visualizar y manipular la gráfica en 3D del ejercicio propuesto en clase, mejorando la percepción por parte de los alumnos.

5.2. Justificación.

Luego del estudio realizado para la recolección de datos (CAPÍTULO IV), se evidencia que los estudiantes de la Facultad de Mecánica presentan problemas en el aprendizaje en el cálculo integral, pues les resulta complicado comprender y asimilar teorías y procesos sobre sólidos de revolución, especialmente la visualización tridimensional del sólido de revolución.

El uso del programa desarrollado en MATLAB como un software educativo permite el apoyo tanto para el docente como para el estudiante, y es que esta herramienta facilita no solo la resolución para verificar la respuesta correcta del ejercicio, sino la visualización en 3D del Sólido de Revolución. Usar MATLAB permite también que el docente corrobore sus teorías explicadas mediante la práctica, porque facilita el trabajo simultáneo de forma analítica y gráfica, e influye de manera positiva en el análisis y comprensión por parte del estudiante.

5.3. Descripción.

La presente propuesta fue diseñada considerando la teoría constructivista del aprendizaje, en donde el estudiante es el elemento principal y protagonista del proceso de enseñanza-aprendizaje. El docente es un guía que tiene la función de motivar a los estudiantes a construir su propio conocimiento, para lo cual empleará MATLAB como estrategia didáctica. Esto permitirá la construcción gráfica y el análisis de resultados, y también que los alumnos lleguen a desarrollar por sí solos los ejercicios, analizando todas las posibles soluciones.

El programa desarrollado en MATLAB permite una metodología interactiva y participativa, para que los alumnos puedan discutir, interpretar y sobre todo analizar las posibles soluciones en el cálculo del volumen de sólidos de revolución. Además, permite que sea un proceso de aprendizaje dinámico, ya que los alumnos pueden graficar, variar los ejes de revolución de forma muy sencilla, explorar las funciones del software, direccionando a que analice e interprete sus resultados.

El papel del docente será explicar los conceptos, definiciones, formas de calcular el volumen de sólidos de revolución, métodos de solución, ejercicios resueltos manualmente y comprobarlos mediante el programa en MATLAB. Ejercicios con diferentes condiciones que mediante el software se visualice el comportamiento, asimismo ejercicios de aplicación que al graficarlos permita el análisis y discusión por parte de los alumnos, lo que conlleva a que los mismos participe de manera activa en la clase y por tanto sea cada uno el constructor de su propio conocimiento.

Al final se plantearán ejercicios para que el alumno los resuelva de forma individual o grupal utilizando el software, lo que conllevará a que utilice, analice, investigue en el software y con sus compañeros compartan sus conocimientos, así como también la experiencia con el software.

La propuesta didáctica (su programación en Matlab) se encuentra disponible en la sección ANEXOS.

5.4. Objetivos de la propuesta didáctica.

- Desarrollar en los estudiantes habilidades y destrezas que ayuden a mejorar su capacidad de razonar, analizar y comprender el cálculo del volumen de sólidos de revolución, mediante el uso del programa creado en MATLAB
- Fomentar la participación del estudiante a través de problemas y ejercicios para la construcción del propio conocimiento.
- Utilizar el programa desarrollado en MATLAB como una estrategia didáctica que brinde apoyo al estudiante para comprobar los resultados obtenidos de una forma analítica y gráfica.

5.5. Resultados de aprendizaje.

- Mejorar en los estudiantes la percepción tridimensional de los problemas de sólidos de revolución.
- Identificar, entender y resolver ejercicios pertinentes al cálculo del volumen de sólidos de revolución en el ámbito real aplicando los distintos métodos con solvencia y calidad.
- Reconocer, comprender y resolver ejercicios pertinentes al cálculo del volumen de sólidos de revolución a través del uso de los métodos descritos para cada tipo con criterio de ingeniería.

5.6. Recursos.

5.6.1. Recursos Materiales.

- Computador. – Permite al alumno de interactuar con el software durante la clase.
- Otros. –Bibliografía básica pertinente a la materia de Cálculo, libreta de apuntes, esferográficos, calculadora científica, formulario de integrales. Es necesario que se tome nota manualmente para resolver los ejercicios planteados.

5.6.2. Recursos Humanos.

- Profesor de la asignatura. -Permite la guía y apoyo en situaciones de dificultad a los alumnos
- Estudiantes. - Los protagonistas de este proyecto

5.6.3. Recursos Tecnológicos.

- Software MATLAB. – Principal recurso tecnológico para realizar esta propuesta, puede usarse la versión de escritorio (instalado en el computador) o la versión en línea de MathWorks (s.f.).
- Aula virtual utilizada. - Plataforma Zoom, Teams, Webex, entre otras

5.7. Capacitación sobre el programa desarrollado en MATLAB.

Para cumplir con la propuesta establecida en este trabajo, a continuación, se describe el funcionamiento del programa a usar en Matlab, el mismo se puede encontrar en el apartado ANEXOS.

5.7.1. Modelo matemático utilizado

Para su funcionamiento, el programa realizado utiliza la integración simbólica, vista en el capítulo 2.2.20, de donde es posible obtener la solución en forma analítica tanto para integrales definidas como indefinidas, pues este método se basa en la cuadratura de Simpson, cuadratura de Lobatto, cuadratura de Gauss, vistas en el capítulo 2.2.7. (MathWorks, s.f.). Se pide encarecidamente al lector revisar dichos capítulos.

Se eligió la integración simbólica ya que permite trabajar con integrales en forma algebraica, es decir, en función de la variable x (MathWorks, s.f.). Si bien es cierto, Matlab cuenta con funciones para hallar integrales mediante el método de trapecios $Q = trapz(Y)$, o la función $q = integral(fun, xmin, xmax)$, pero estos métodos representan una desventaja al trabajar solamente con valores numéricos (MathWorks, s.f.). Lo que se necesita es trabajar con variables simbólicas para que sea posible visualizar el desarrollo del cálculo de las integrales en su aplicación a los estudiantes.

Es así que el programa emplea la función *int* para los cálculos del programa desarrollado en la propuesta y cuenta con la ventaja de que se puede utilizar tanto en integrales definidas como indefinidas, así como trabajar con varias variables (MathWorks, s.f.).

Entonces, dependiendo de si el método a usar es el de discos, de arandelas o capas cilíndricas, se utiliza el formato de la Tabla 1-5 donde f , g , a y b serán datos de entrada ingresados por el usuario, y se basa en la teoría de cálculo de sólidos de revolución, vista en el capítulo 2.2.6.

Tabla 1-5: Modelos matemáticos implementados en el programa

Método de Discos	Método de Arandelas	Método de Capas cilíndricas
$V = \lim_{\ \Delta x\ \rightarrow 0} \sum_{i=1}^n \pi * [f(x)]^2 \Delta_i x$	$V = \lim_{\ \Delta x\ \rightarrow 0} \sum_{i=1}^n \pi * ([f(x)]^2 - [g(x)]^2) \Delta_i x$	$V = \lim_{\ \Delta x\ \rightarrow 0} \sum_{i=1}^n 2\pi \bar{x}_i * f(\bar{x}_i) \Delta_i x$

Realizado por: Bonilla, J. 2022

La forma en que se elige cuál método usar se describe más adelante.

5.7.2. *Funcionamiento del programa desarrollado en MATLAB.*

Este programa, que emplea la integración simbólica, permitirá al estudiante encontrar el resultado de aplicar una integral definida para hallar el área bajo o entre las curvas y el volumen del sólido de revolución generado al rotar sobre un eje. La interfaz del programa fue desarrollada en MATLAB de modo que fuera de naturaleza intuitiva y simple, por lo cual no se necesita de grandes bases de programación. Lo único que se requiere para saber utilizarlo, es familiarizarse con su entorno gráfico y el modo en que se usa el programa. La propuesta siguió los modelos matemáticos de integrales correspondientes para cálculo de sólidos de revolución, vistos en el Cap. 2.2.6, y haciendo uso de programación en Matlab, como el bucle *if* y gráficas *plot*, (ver ANEXO A).

El programa fue diseñado para seleccionar por sí solo el método de discos, arandelas o capas cilíndricas, de modo que se cumpla la condiciones de que, si el eje de revolución es paralelo al diferencial de área, se aplique el método de capas cilíndricas (Leithold, 1998). Esto se muestra en la interfaz, vista más adelante. El diseño del programa también permite operar con una variable dependiente, por lo que, si se ingresa la variable incorrecta se producirá un error que impedirá continuar con la ejecución del programa, adicional se presentará un mensaje en el que indicará el problema existente. La interfaz principal facilitará elegir, ya sea dos funciones o una función, también permitirá elegir con qué variable dependiente trabajar.

Este programa controlará que el límite inferior sea menor que el límite superior, es decir, que $a > b$. Se tendrá un control sobre las variables a , b , m o k para garantizar que se está trabajando con valores numéricos reales, excluyendo así los datos tipo NAN y tipo INF.

Además, el programa presenta unas características con el fin de garantizar un resultado óptimo de los datos ingresados, como el hecho de requerir funciones despejadas con el fin de evitar que al despejar una de ellas se termine encontrando con múltiples resultados y generar errores en los resultados obtenidos.

El siguiente diagrama representa el programa del cálculo del volumen de sólidos de revolución implementado en MATLAB. En la parte izquierda se encuentran las entradas como función a girar, límite superior e inferior, rotación para eje x , para eje y , un desplazamiento en $x=k$, $y=k$, o incluso una recta en la forma $y=mx+b$. En la parte derecha se encuentra la salida, la cual es el volumen calculado. Y en las partes superior e inferior de este esquema se muestra los resultados, procesos y condiciones empleadas para obtener la respuesta, así como las gráficas en 2D y 3D, y los errores comunes que pueden darse como dar a correr el programa sin haber llenado los campos requeridos.

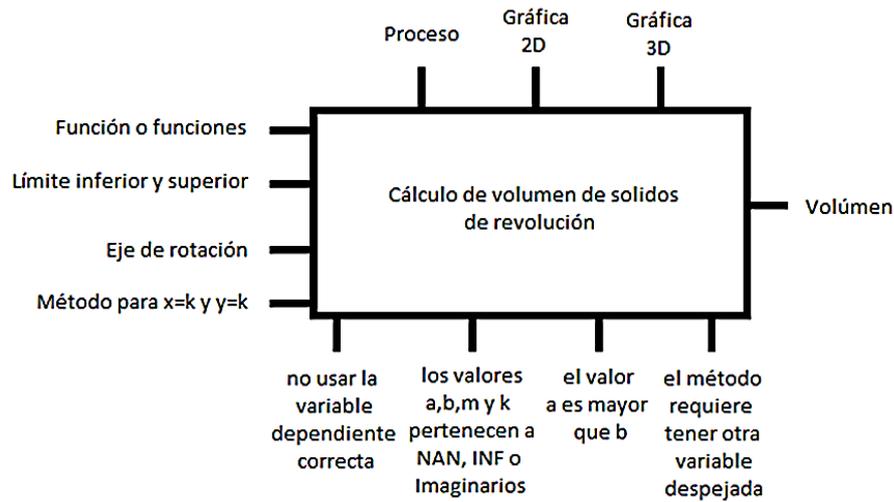


Figura 1-5: Diseño del programa en MATLAB
 Realizado por: Bonilla, J. 2022

5.7.2.1. *Vistas Disponibles.*

a) Interfaz principal.

La interfaz principal del programa cuenta con un menú en el que se puede elegir el método a resolver el ejercicio deseado. Además, cuenta con un enlace a un video tutorial respecto a cómo utilizar esta aplicación, que será útil para los nuevos usuarios del mismo. Las funciones que se encuentran en la interfaz son las siguientes, y se muestran en la Figura 2-5.

- Función Generatriz $f(x)$,
- Función Generatriz $f(y)$,
- Función Generatriz entre dos funciones $f(x)$ y $g(x)$,
- Función Generatriz entre dos funciones $f(y)$ y $g(y)$,
- Cinco funciones a intervalos $fn(x)$ y $gn(x)$,

Los modelos matemáticos que el programa emplea fueron descritos en el capítulo 2.2.6. La implementación de ellos en Matlab se basa en el uso de integración simbólica, visto en el cap. 2.2.22. La aplicación es capaz de resolver fácilmente las integrales “difíciles” como una raíz cúbica (Leithold, 1998, p. 391), y a través de métodos numéricos (Chapra & Canale, 2007, p. 5).



Figura 2-5: Interfaz Principal del programa en MATLAB

Realizado por: Bonilla, J. 2022

En este menú principal, en general, las dos primeras opciones representan el método de discos, las dos siguientes opciones representan el método de arandelas, la opción final (cinco intervalos) implica una función definida por partes, en donde puede usarse el método de discos o arandelas, según la definición de f y g . Sin embargo, puede haber excepciones cuando se trabaja con desplazamientos de $x=k$, o $y=k$, como en la Figura 3-5. Además, todas las cinco opciones de este menú pueden resolverse por el método de capas cilíndricas cuando el eje de revolución sea paralelo al diferencial de área (Leithold, 1998). Esto se resume en el siguiente esquema, que muestra todos los casos posibles a resolver y los métodos a emplear que fueron programados:



Figura 3-5: Casos posibles y métodos con que trabaja el programa en MATLAB

Realizado por: Bonilla, J. 2022

b) Facilidad de entrada.

El programa creado en MATLAB permite introducir directamente la función generatriz y los límites necesarios en el cuadro respectivo, además se despliegan las opciones para seleccionar el eje de giro, la pantalla donde se muestra el proceso, y varios botones para utilizar según las necesidades. Al hacer clic en una de las opciones del menú principal, se muestra una interfaz similar a la de la Figura 4-5.

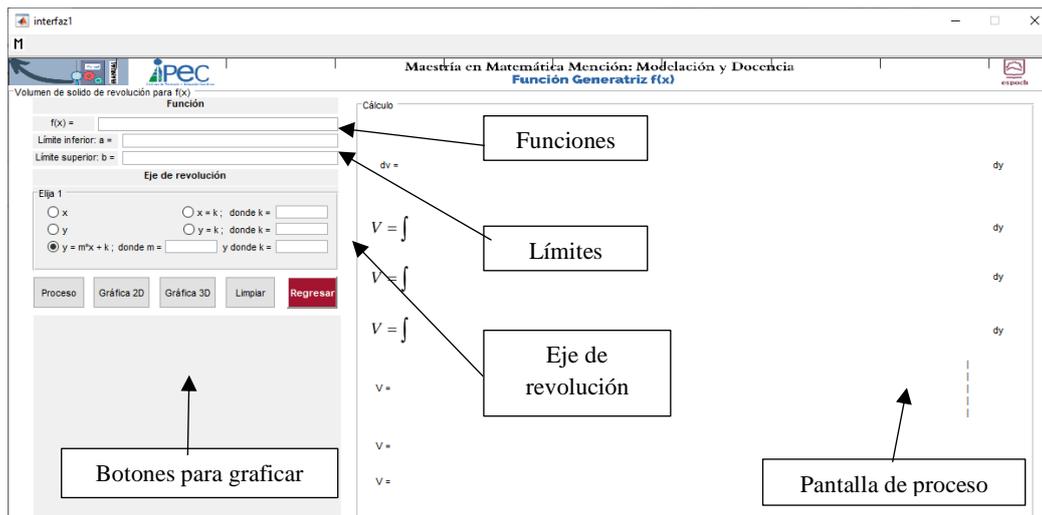


Figura 4-5: Interfaz función generatriz $f(x)$

Realizado por: Bonilla, J. 2022

En esta pantalla se deberá ingresar los datos de: función a integrar, límites a y b (el sistema los reconoce y ordena automáticamente de menor a mayor, para evitar errores), selección del eje de revolución, pantalla del proceso y botones para graficar en 2D y 3D. Además, se puede reiniciar esta ventana dándole clic al botón “limpiar”, y volver al menú principal, con un clic en “regresar”.

En la sección ANEXOS se muestra un ejemplo de cálculo de volúmenes de sólidos de revolución utilizando el programa.

c) Representación.

En el programa desarrollado en MATLAB se trabaja con la parte analítica y gráfica, es decir que se muestra en una ventana la gráfica 2D y 3D correspondiente a los datos ingresados, como en la siguiente figura: Para este caso, las gráficas corresponden a una función cuadrática ($f(y) = y^2$).

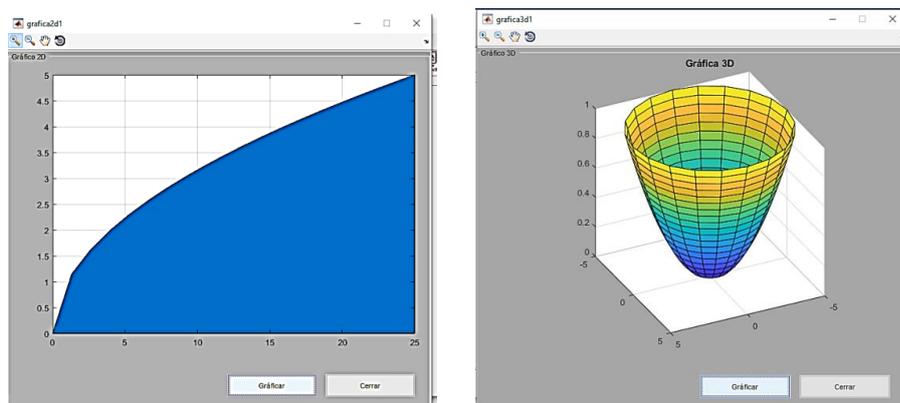


Figura 5-5: Gráfica 2D (izquierda) y 3D (derecha)

Realizado por: Bonilla, J. 2022

5.8. Fases de la propuesta.

Mediante la aplicación de la teoría de aprendizaje constructivista, se aplicaron 8 fases para el desarrollo de esta propuesta.

5.8.1. Fase 1 (Motivación).

Es la primera fase y corresponde a la motivación que se deben crear en los alumnos, para generar expectativas y disposiciones en ellos y que ellos tengan interés por aprender. En esta fase se propone al docente que inicie mencionando las aplicaciones del cálculo integral en el mundo real, es decir hablando sobre los fenómenos que rigen en la naturaleza o situaciones cotidianas. Por ejemplo, hablar sobre el volumen de sólido de revolución contenido en un sartén o tazón; los combustibles sólidos tales como, el carbón, la madera y la turba, entre otros, también se comportan como un sólido de revolución, que no es otra cosa que resolver una integral.

Este relato se recomienda que sea lo suficientemente pausado para permitir que los estudiantes asimilen el mismo y que puedan señalar más sucesos que se comportan de manera similar bajo los modelos matemáticos mencionados por el docente. Así se podrá analizar la materia en estudio y mejorar la percepción de la importancia de estudiar el cálculo integral.

5.8.2. Fase 2 (Comprensión).

En la segunda fase se necesita la atención y percepción de los estudiantes, con el objetivo de que retengan las ideas sobre los aspectos que se va a indicar en la clase. Para ello el docente debe haber despertado el interés en los estudiantes por aprender sobre el cálculo de volumen de sólidos de revolución con el cálculo integral y debe presentar aquellos subtemas a estudiar, de tal forma que todos los estudiantes conozcan los contenidos de estudio revisados y en los cuales pongan la atención necesaria.

Subtemas a estudiar:

- a) Conceptos básicos para el cálculo de volumen de sólidos de revolución, tipos de métodos de solución, como los vistos en el cap. 2.2 de este trabajo.
- b) Ejercicios con soluciones generales y ejercicios con soluciones particulares.
- c) Los mismos ejercicios, utilizando el programa desarrollado en MATLAB.

d) Aplicaciones del cálculo de volumen de sólidos de revolución como modelos matemáticos, usando el programa desarrollado en MATLAB.

5.8.3. Fase 3 (Adquisición).

Esta fase comprende la ganancia de nuevos conocimientos por parte del estudiante, enfatizando lo fundamental del aprendizaje, separando los contenidos por partes, y dándoles a conocer los mismos de forma organizada y estructurada, para que el alumno pueda comprenderlos fácilmente.

En la fase 3 el docente es quien estructura los temas a impartir en clase de una forma ordenada y clara, mostrando conceptos, definiciones, y los diferentes métodos para el cálculo de volumen de sólidos de revolución.

5.8.3.1. Teoría por Aprender.

La teoría corresponde con el cálculo de sólidos de revolución, el cual fue visto en el capítulo “Bases Teórica

Introducción a los Sólidos de Revolución” de este trabajo.

5.8.4. Fase 4 (Retención).

En la cuarta fase se tiene que asegurar que el estudiante haya aclarado las ideas y conceptos recibidos, para que los almacene en su memoria. Por lo tanto, el docente deberá crear un ambiente propicio para que los alumnos comuniquen sus dudas, inquietudes y opiniones, y que así los contenidos permanezcan claros y retenidos en su memoria.

5.8.5. Fase 5 (Recuperación).

En esta fase se debe verificar que los estudiantes recuerdan tanto conceptos antiguos como nuevos, es decir lo almacenado en su memoria. Es así como el docente realizará diversas actividades de participación de los alumnos, de tal forma que los mismos estudiantes sean quienes desarrollen las preguntas planteadas por el docente. De esta manera el estudiante será principal actor de la actividad, generando así una memoria prolongada y que relacione los conocimientos previos con los nuevos.

5.8.6. Fase 6 (Generalización o Transferencia).

En esta fase el estudiante ya debe ser capaz de aplicar los conocimientos recibidos en la resolución de problemas nuevos. Para tal fin, el docente plantea ejercicios y los resuelve de forma conjunta

con los estudiantes. En la resolución debe aplicar los conocimientos adquiridos en la fase 3, que corresponden a la identificación de los sólidos de revolución, el cálculo de volumen de sólidos de revolución, los métodos para resolver, la solución paso a paso, encontrar soluciones general y particular, y realizar un análisis de estas, pero siempre tomando en cuenta la constante participación de los alumnos. Por ejemplo, el docente puede plantear los siguientes ejercicios, que serán resueltos por los estudiantes, y luego el docente les mostrará la corrección de estos, de una manera similar a esta:

5.8.6.1. Ejercicios Para Resolver.

A manera de puesta en práctica, y con el fin de cumplir con las fases de la propuesta, se plantean los siguientes ejercicios:

Ejercicio 1: Sea $y = -x + 1$, $x = 1$ & $x = 0$. Formular y evaluar la integral que da el volumen del sólido formado al girar la región alrededor del eje x .

Paso1: Lo primero que se debe realizar es graficar la función, al hacerlo se obtiene:

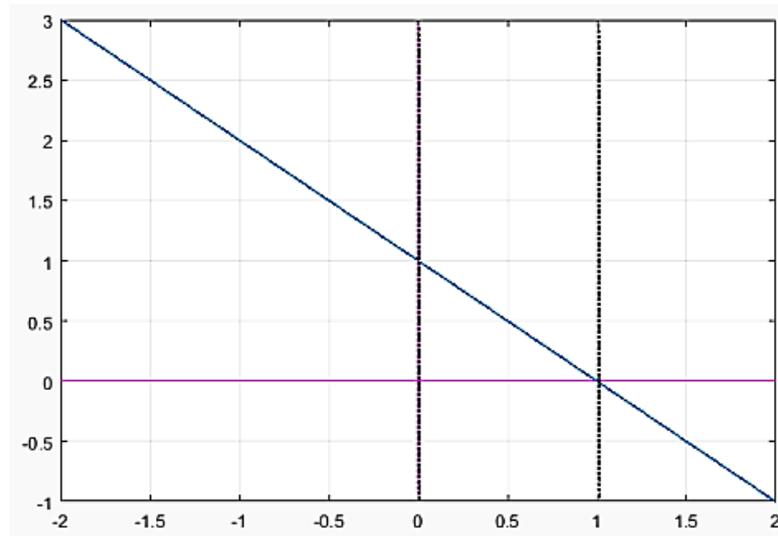


Figura 6-5: Representación gráfica de las funciones

Realizado por: Bonilla, J. 2022

Paso2: Una vez realizada la gráfica, se puede observar mejor los límites del sólido formado por la función, como se obtuvo rectángulos verticales, se integra respecto a x . Aplicando el método de discos, mediante la ec. (8).

$$V = \pi \int_a^b [f(x)]^2 dx$$

$$V = \pi \int_0^1 [-x + 1]^2 dx$$

Paso3: Se procede a integrar y evaluar.

$$V = \pi \int_0^1 (x^2 - 2x + 1) dx$$

$$V = \pi \left[\frac{x^3}{3} - x^2 + x \right]_0^1$$

$$V = \pi \left[\frac{1^3}{3} - (1)^2 + 1 \right] = \pi \left[\frac{1}{3} - 1 + 1 \right]$$

$$V = \frac{1}{3} \pi u^3$$

Ejercicio 2: Dadas las funciones $y = x^2$ y $y = x$, determine el volumen del sólido de revolución que se forma al girar el área comprendida entre ambas funciones, alrededor del eje X.

Paso 1: Se realiza la gráfica de las dos funciones, después se gira las dos funciones alrededor de la función dada (en este caso es el eje X).

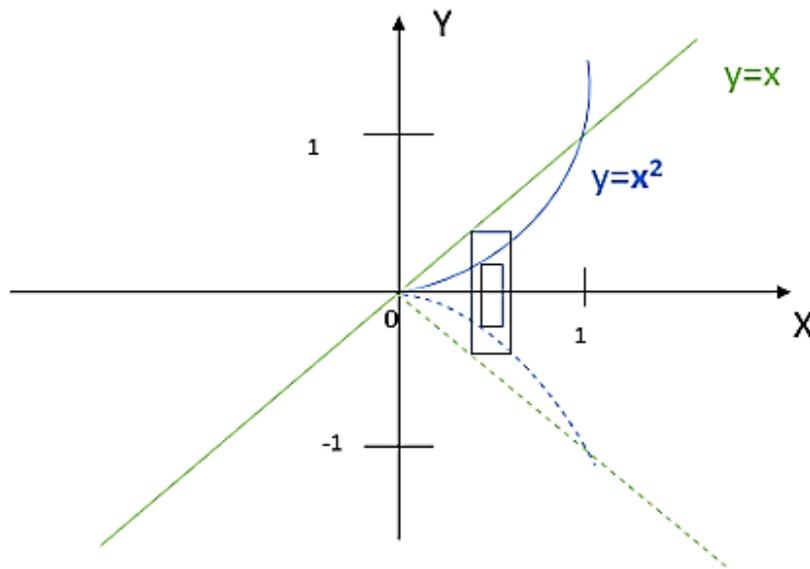


Figura 7-5: Representación gráfica de las funciones planteadas

Realizado por: Bonilla, J. 2022

Paso 2: Se procede a seleccionar el eje de giro, empleando el método de arandelas, como se mostró en las ecuaciones (11) - (13). En este caso se aplica la ec. (11) por tratarse de un giro en torno al eje x.

Paso 3: Se define cuál función será el radio interno y radio externo.

$$r_E = x; r_{E^2} = x^2$$

$$r_I = x^2; r_{I^2} = x^4$$

Paso 4: Se sustituye

$$Vf(x) = \pi \int_a^b [Re^2 - Ri^2]dx$$

$$V = \pi \int_0^1 [x^2 - x^4]dx$$

Paso 5: Se resuelve la integral evaluando ente los límites establecidos

$$V = \pi \left[\left[\frac{x^3}{3} - \frac{x^5}{5} \right] \right]_0^1$$

$$V = \pi \left[\frac{1^3}{3} - \frac{5^5}{5} \right]$$

$$V = \frac{2}{15} \pi u^3$$

5.8.7. Fase 7 (Desempeño).

En la séptima fase el estudiante debe ser capaz de dar una respuesta, analizarla y discutirla. Para ello el docente hará uso del programa creado en MATLAB como elemento de apoyo, para comprobar las soluciones obtenidas en los ejercicios planteados y a su vez que en cada uno de ellos los estudiantes puedan analizar la solución y discutirla, focalizando en la solución algebraica pero también en la solución gráfica.

5.8.7.1. Procedimiento para utilizar el programa desarrollado en MATLAB.

- Abrir MATLAB

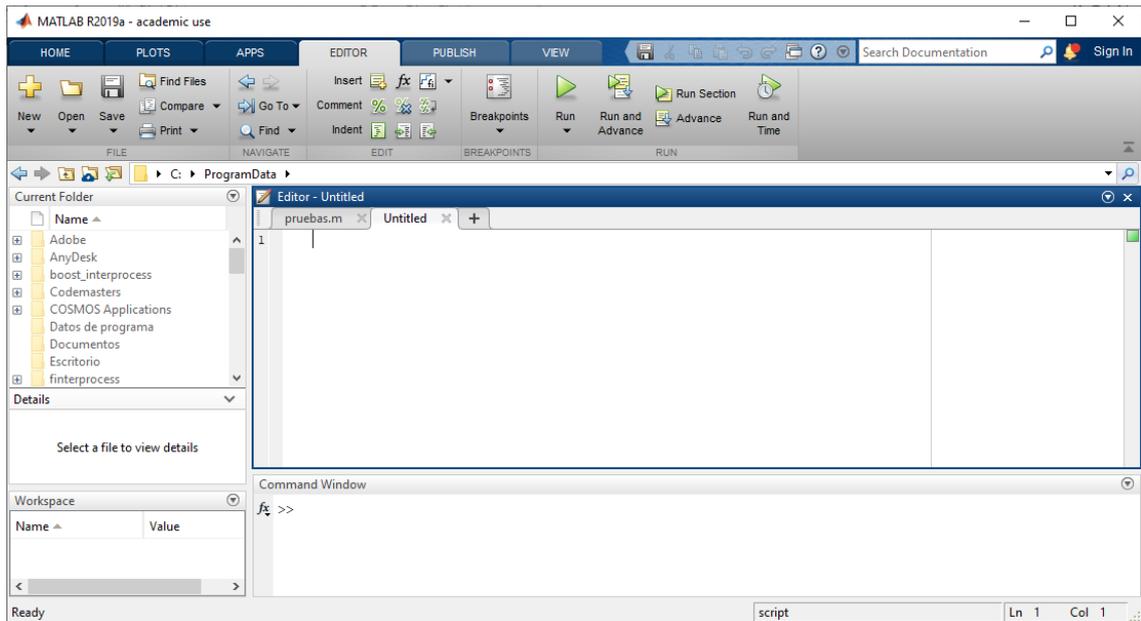


Figura 8-5: Ventana principal de MATLAB

Realizado por: Bonilla, J. 2022

- Abrir el directorio correspondiente (ctrl + O) y seleccionar el archivo MENU_PRINCIPAL.m

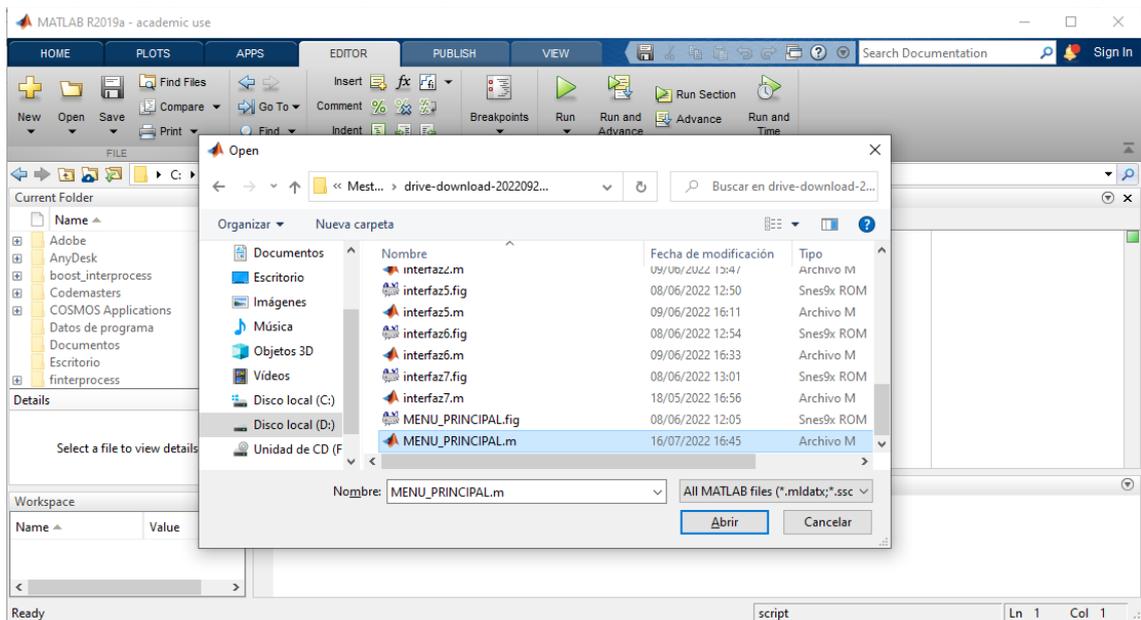


Figura 9-5: Abrir programa creado en MATLAB

Realizado por: Bonilla, J. 2022

- En el archivo MENU_PRINCIPAL.m dar clic al botón ejecutar o presionar la tecla F5.

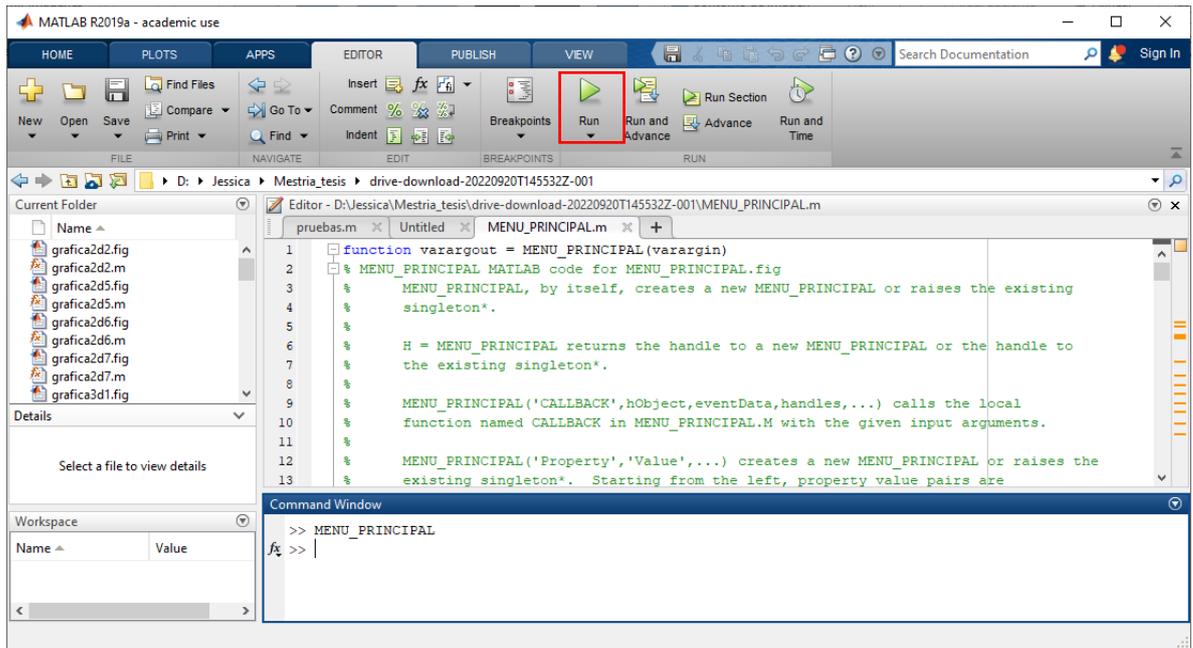


Figura 10-5: Ejecución del programa creado en MATLAB

Realizado por: Bonilla, J. 2022

- Se mostrará los diferentes métodos para el cálculo de volumen de sólidos de revolución.

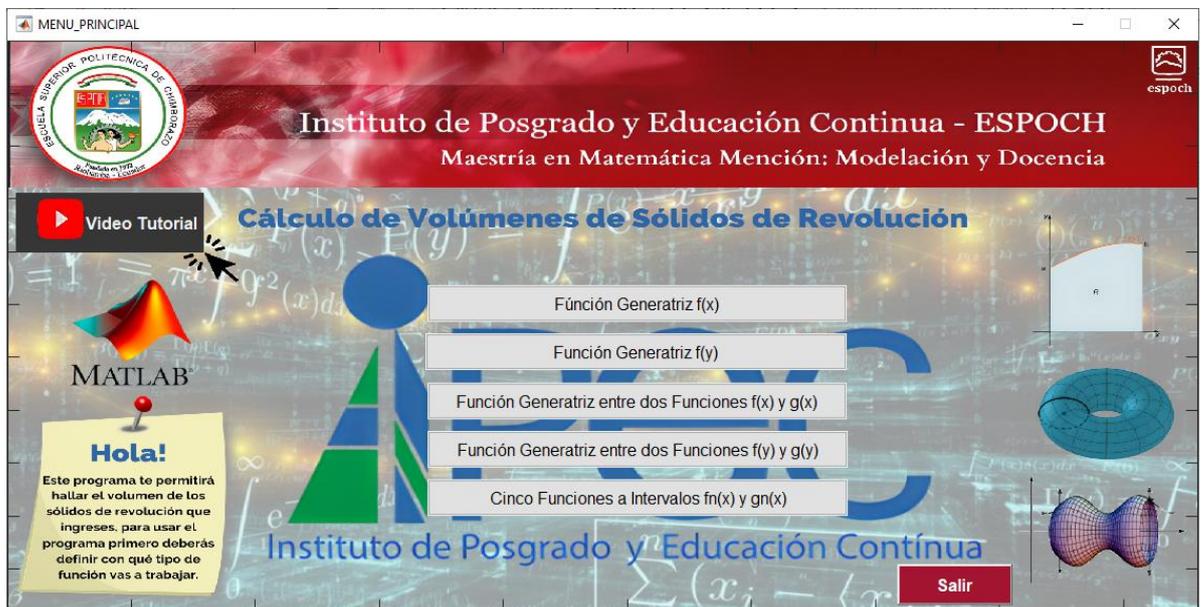


Figura 11-5: Interfaz Principal del programa desarrollado en MATLAB para el cálculo de volumen de sólidos de revolución.

Realizado por: Bonilla, J. 2022

En el ANEXO B se muestran ejemplos a resolver utilizando el programa de esta propuesta.

5.8.8. Fase 8 (Retroalimentación).

Esta es la octava fase, y el propósito es la verificación entre un criterio establecido de antemano y la validez de la respuesta asimilada. Por lo cual, el docente deberá realizar una retroalimentación de todos los contenidos impartidos en clase, permitiendo así la activa intervención de los estudiantes con dudas, inquietudes y opiniones para que puedan verificar la información recibida.

El docente en esta parte deberá plantear ejercicios como refuerzo, estos deben ser los suficientes con el fin de que el estudiante llegue a utilizar su tiempo de aprendizaje autónomo para la realización de estos. Así el estudiante podrá analizarlos y verificarlos con el programa desarrollado en MATLAB para el cálculo de volúmenes de sólidos de revolución.

5.8.8.1. Tarea complementaria.

Resolver los siguientes ejercicios sobre volumen de Sólidos de evolución, comprobarlos con el programa desarrollado en MATLAB y analizar el comportamiento de los mismos:

- Hallar el volumen del tronco de cono engendrado por la rotación alrededor OX del área limitada por $y = 6 - x$, $y = 0$, $x = 0$, $x = 4$.
- Encontrar la región acotada por la curva $y = x^2 + 1$ y la recta $y = -x + 3$ que gira alrededor del eje x para generar el sólido.

CONCLUSIONES

En este trabajo se utilizó la modelación matemática como estrategia didáctica en el cálculo de volumen de sólidos de revolución, con el objetivo de mejorar el aprendizaje de los estudiantes de Escuela Superior Politécnica de Chimborazo, Facultad de Mecánica, periodo abril-septiembre 2021.

Los resultados de la encuesta realizada antes de implementar la propuesta demuestran la situación en que se encontraban los estudiantes: la suma de los porcentajes de aquellos estudiantes que consideran que su nivel de comprensión respecto al cálculo de volumen de sólidos de revolución califica como satisfactorio y muy satisfactorio es de 47,5% que, en términos relativos, es menos de la mitad. Por otra parte, a pesar de que el 55,9% del total de los encuestados manifiestan hacer uso frecuente de herramientas tecnológicas para aprender la materia de cálculo, solo el 27,1% las emplea para el tema de sólidos de revolución. Este pequeño grupo representa 47 de las 172 personas encuestadas, entonces es necesario implementar y popularizar las herramientas de modelación matemática, como la aplicación del presente trabajo.

Se desarrolló una propuesta mediante modelación matemática que consistió en implementar los modelos matemáticos para el cálculo de volúmenes de sólidos de revolución, haciendo hincapié en la visualización tridimensional del ejercicio que, según las encuestas, es la parte que más les cuesta entender a los estudiantes.

Para ejecutar el programa desarrollado en Matlab, solo se debe introducir la función generatriz, o las funciones (dado el caso), llenar los campos requeridos para: límites de integración, eje de revolución, y darle clic al botón de proceso de resolución, gráfica 2D, o gráfica 3D validando y el aprendizaje de los estudiantes.

Luego de la aplicación de la propuesta, el 84,7% considera que sí ha mejorado su nivel de comprensión sobre la materia de cálculo integral, en el tema de sólidos de revolución, permitiendo dar validez a la misma. Entre los comentarios obtenidos, los estudiantes manifestaron que la única mejora que harían es el formato de las ilustraciones en 3D.

Los estudiantes expresaron, en la encuesta posterior a la aplicación de la propuesta, que la interfaz presentada en Matlab les parece satisfactoria, y muy útil, sumando un 95% de aceptación, y entre las sugerencias consta que agregarían más videos tutoriales. Con esto se comprueba que su aceptación por el programa es apreciable y que su actitud hacia el aprendizaje de cálculo del volumen de sólidos de revolución ha mejorado.

RECOMENDACIONES

Se recomienda, a los estudiantes, emplear el programa con una base de conocimiento sobre el cálculo de volúmenes de revolución, ya que no es lo mismo usarlos sin saber nada que aplicarlos sabiendo algo de antemano.

Al abrir la interfaz por primera vez puede que no se reconozca la variable de rotación o el método elegido, por lo que se recomienda intentar con otro eje, volver a darle clic a proceso, o volver a abrir la interfaz. Este problema es debido a que aún no se reconoce la ubicación de la variable en memoria. Por esta razón se está dando de baja el sistema de *Guide* en Matlab y se está adoptando *App Designer* de Matlab.

En caso de error, revisar que los datos ingresados sean adecuados a los requerimientos del programa, por ejemplo, no dejar en blanco, el eje de revolución antes de darle a correr.

Esta herramienta didáctica servirá para futuros estudiantes de ingeniería, no solo de la Facultad de Mecánica, por lo que es recomendable fomentar su uso en toda la Escuela Superior Politécnica de Chimborazo, para que los futuros estudiantes puedan tener un adecuado proceso de aprendizaje.

También se recomienda a los estudiantes practicar a mano los ejercicios de Sólidos de Revolución, ya que esto impedirá el acostumbrarse a usar la herramienta como calculadora (modelación matemática de esta propuesta) y que retrase su progreso en el aprendizaje de esta materia.

Los docentes de Matemática para complementar los conocimientos impartidos deben hacer uso de herramientas tecnológicas, de esta forma las clases serán más participativas e interactivas en cualquier tema, motivando así a los estudiantes a su autoaprendizaje mediante la teoría constructivista.

GLOSARIO

Aprendizaje: es la adquisición de conocimiento de algo por medio del estudio.

Autoaprendizaje: es la habilidad de una persona para adquirir conocimiento por si sola.

Cálculo: es la operación o conjunto de operaciones matemáticas necesarias para encontrar un resultado.

Derivada: representa la pendiente de una recta tangente a la gráfica de una función en un punto.

Integral: es una generalización de la suma de infinitos sumandos.

Intervalo: es la porción de tiempo o espacio que hay entre dos hechos o dos cosas de la misma naturaleza.

Método: procedimiento que se sigue para encontrar un resultado.

Muestra: es la parte de una población que se considera representativa del total.

Matemática: es una ciencia que estudia los números y sus propiedades, y la relación que existe entre ellos.

BIBLIOGRAFÍA

- Agreda, H. (Agosto de 2021). Sistemas operativos. Obtenido de Ministerio del Poder Popular para la Educación Universitaria Ciencia y Tecnología: <https://es.scribd.com/document/524166392/SISTEMAS-OPERATIVOS1>
- Alfonso, A., De Armas, R. J., & Macías, D. (s.f.). Modelado Matemático y Simulación con MatLab Para estudiantes de Ingeniería (Vol. 1). Recuperado agosto 28, 2022, de <https://pdfcoffee.com/libro-modelamiento-y-simulacionpdf-4-pdf-free.html>
- Aranda, T. (2005). Algoritmos. Manual formativo de ACTA, 35, 43-50. Obtenido de <https://dialnet.unirioja.es/servlet/articulo?codigo=5098610>
- Aravena, M., Caamaño, C., & J. G. (2008). Modelos matemáticos a través de proyectos. Revista latinoamericana de investigación en matemática educativa, 11(1), 49-92. Obtenido de https://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S1665-24362008000100003
- Atllo, B. (Febrero de 2018). Estrategias didácticas para el uso de TiC's en la docencia universitaria presencia. Obtenido de Un manual para los ciudadanos del Ágora: <http://eprints.rclis.org/9542/1/manualedTICS.pdf>
- Barragán, D. (2008). MANUAL DE INTERFAZ GRÁFICA DE USUARIO EN MATLAB, Parte I. Ecuador. doi:https://www.utm.mx/~vero0304/HCPM/MATLAB_GUIDE.pdf
- Bassanezi, R., & Biembengut, S. (1997). Modelación Matemática. Didáctica de las matemáticas, 13-25. Obtenido de <http://funes.uniandes.edu.co/3171/1/C1997Modelaci%C3%B3nNumeros32.pdf>
- Becerra, J. (2016). Matemáticas VI. Un Paseo Sencillo e Introductorio al Cálculo. México: Universidad Nacional Autónoma de México. Obtenido de <https://www.worldcat.org/es/title/651345313>
- Belando-Montoro, M. R. (2014). Formación permanente del profesorado. Algunos recursos TIC para la docencia universitaria. 65(1), 1-11. doi:10.35362/rie651324
- Biembengut, S., & Hein, N. (02 de Agosto de 2004). Modelación matemática y los desafíos. Grupo Santillana México, 16(2), 105-125. Obtenido de <https://www.redalyc.org/pdf/405/40516206.pdf>
- Bocco, M. (2010). Funciones Lementales para construir Modelos Matemáticos (1 ed.). Ciudad Autónoma de Buenos Aires, Argentina: Ministerio de Educación. Recuperado de https://isfdarientiers.infed.edu.ar/sitio/upload/Funciones_elementales_para_construir_modelos_matem%Eltcos_Bocco_2013.pdf
- Brito, L., Aleman, I., Fraguerra, E., Para, J., & Arias, R. (2011, Agosto 02). Papel de la Modelación Matemática en la Formación de los Ingenieros. Ingeniería Mecánica, 14(2), 129-139. Recuperado de <http://scielo.sld.cu/pdf/im/v14n2/im05211.pdf>

- Carvajal, A. (2002). Teorías y modelos. Comunicación, 1-14. Recuperado de <https://www.redalyc.org/pdf/166/16612103.pdf>
- CEDED. (2019). Universidad Estatal a Distancia. Obtenido de Vicerrectoría Académica: https://www.uned.ac.cr/academica/images/ceced/docs/Estaticos/contenidos_curso_2013.pdf
- Chapra, C., & Canale, R. (2007). Métodos numéricos para ingenieros (5 ed.). Nueva York: McGraw Hill Interamericana. Recuperado de <http://artemisa.unicauca.edu.co/~cardila/Chapra.pdf>
- CK-12. (2015). Conceptos de Cálculo - en español. Obtenido de <https://www.ck12.org/book/ck-12-conceptos-de-c%C3%A1lculo-en-espa%C3%B1ol/>
- Contreras, G., & García, S. (2010). Uso de simuladores como recurso digital para la transferencia de conocimiento. Obtenido de Universidad de Guadalajara: <http://www.udgvirtual.udg.mx/apertura/index.php/apertura/article/view/22/32>
- Díaz Lozada, J. A., & Díaz Fuentes, R. (2018). Los Métodos de Resolución de Problemas y el Desarrollo del Pensamiento Matemático. Bolema: Boletín de Educación Matemática, 32(60), 57-74. doi:10.1590/1980-4415v32n60a03
- Díaz, L. (2018). A P R E N D I E N D O. Obtenido de Matemáticas: <https://www.cecyl3.ipn.mx/ibiblioteca/mundodelasmaticas/iu3volum.html>
- Duval, R. (2016). Un tema crucial en la educación matemática: La habilidad para cambiar el registro de representación. Dialnet, 143-168. Obtenido de <https://dialnet.unirioja.es/servlet/articulo?codigo=1984436>
- Editorial Etecé. (2021). Concepto. Recuperado el 15 de 01 de 2022, de Concepto Volumen: <https://concepto.de/volumen/>
- Espeleta, A., & Zamora, W. (Junia de 2016). Una experiencia de aprendizaje en el marco de los planteamientos del NCTM. FESTIVAL INTERNACIONAL DE MATEMÁTICA,, 185-191. Obtenido de <http://funes.uniandes.edu.co/17118/1/Espeleta2016Una.pdf>
- Espinoza, E. (2002). Análisis Matemático II (3 ed.). Lima, Perú: Urbanización Ingeniería. Obtenido de https://www.academia.edu/24305420/AN%C3%81LISIS_MATEM%C3%81TICO_II_CALCULO_II_Espinoza_Ramos_
- Fandos, G. (13 de Octubre de 2003). Tipos de software educativo. Obtenido de: https://www.tdx.cat/bitstream/handle/10803/8909/Etesis_1.pdf
- Farias, I. (12 de Septiembre de 2018). Psicocode. Obtenido de Software educativo: 10 tipos, características y usos: <https://psicocode.com/tecnologia/tipos-software-educativo/>
- Forteza, M. (2019). Metodologías didácticas para la enseñanza/aprendizaje de competencias. Obtenido de Unitat de Formació i Innovació Educativa: <https://pdfs.semanticscholar.org/c76a/c3e020d51434724a32f5b87032f12bcac7aa.pdf>

- Geogebra. (s.f.). Geogebra: Recursos. Obtenido de <https://www.geogebra.org/t/calculus>
- Godino, J. (2013). Matemáticas y su Didáctica para Maestros. Granada: Facultad de Ciencias de la Educación. Obtenido de https://www.ugr.es/~jgodino/edumat-maestros/manual/1_Fundamentos.pdf
- Government, R. (Agosto de 2011). ORACLE. Obtenido de Guía de administración del sistema: administración avanzada: https://docs.oracle.com/cd/E24842_01/pdf/E23086.pdf
- Haro, F. (22 de Mayo de 2020). Aplicación del Software Libre Geogebra en el Aprendizaje de Sólidos de Revolución en Cálculo Integral para los Estudiantes de la Carrera de Pedagogía de las Ciencias Experimentales, Matemática y Física de la Universidad Central del Ecuador. Tesis de Grado, Universidad Central del Ecuador, Quito. Obtenido de <http://www.dspace.uce.edu.ec/bitstream/25000/22494/1/T-UCE-0010-FIL-1020.pdf>
- Hernández, G. (1983). La enseñanza de las matemáticas avanzadas para ingeniería. *Ciencia y Sociedad*, 3(1), 41-47. doi:10.22206/cys.1983.v8i1.pp41-7
- Herrera, M. (2012, Mayo). Las Actividades Reveladoras del Pensamiento. Una Opción de Aprendizaje en las Operaciones de Fracciones en Primero de Secundaria. Tesis de Maestría, México. Recuperado de EGE: https://repositorio.tec.mx/bitstream/handle/11285/571104/DocsTec_12090.pdf?sequence=1&isAllowed=y
- Hotmath. (12 de Septiembre de 2019). Perímetro, área, y volumen. Obtenido de Varsity Tutors: https://www.varsitytutors.com/hotmath/hotmath_help/spanish/topics/perimeter-area-volume
- Incropera, F. P., & DeWitt, D. P. (1999). Fundamentos de Transferencia de Calor (4 ed.). PEARSON. Recuperado de <https://www.pdfdrive.com/fundamentos-de-transferencia-de-calor-d158202052.html>
- Leithold, L. (1998). El Cálculo (7 ed., Vol. 343). México: Oxford University Press. Recuperado de http://kali.azc.uam.mx/clc/03_docencia/leithold.pdf
- Linares Pons, N., Verdecia Martínez, E. Y., & Álvarez Sánchez, E. A. (2014). Tendencias en el desarrollo de las TIC y su impacto en el campo de enseñanza. *Revista Cubana de Ciencias Informáticas*, 8(1), 127-139. Recuperado el 04 de agosto de 2022, de http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=s2227-18992014000100008
- Lluis-Puebla, E. (2006). Teorías matemáticas, matemática aplicada y computación. *Ciencia Ergo Sum, Revista Científica Multidisciplinaria de Prospectiva*, 13(1), 91-98. Obtenido de <https://www.redalyc.org/pdf/104/10413112.pdf>
- Martínez, A. (17 de Noviembre de 2021). Definición de Cálculo. Obtenido de <https://conceptodefinicion.de/calculo/>. Consultado el 1 de diciembre del 2021
- Martínez, P. (1983). Metodología de la Enseñanza de las Matemáticas en las Escuelas de Ingeniería. *Ciencia y Sociedad*, 8(1), 12-18. doi:10.22206/cys.1983.v8i1.pp12-8
- Martínez, S. (2019). Capacidad de enseñanza para la educación superior: La Importancia de la formación pedagógica en los docentes. *Edurama*, 35-38. Obtenido de

<http://revistaedurama.com/capacidad-de-ensenanza-para-la-educacion-superior-la-importancia-de-la-formacion-pedagogica-en-los-docentes/>

- Mathworks. (nd). Modelizar y simular el comportamiento de un sistema dinámico con MATLAB, Simulink, Stateflow y Simscape. Obtenido de <https://la.mathworks.com/discovery/modeling-and-simulation.html#:~:text=La%20modelizaci%C3%B3n%20es%20una%20forma,para%20over%20c%C3%B3mo%20se%20comporta.>
- MathWorks. (s.f.). Centro de ayuda. Recuperado agosto 28, 2022, de function: <https://la.mathworks.com/help/matlab/ref/function.html>
- MathWorks. (s.f.). Help Center. Recuperado sep 05, 2022, de Int: <https://la.mathworks.com/help/symbolic/sym.int.html>
- MathWorks. (s.f.). Help Center. Recuperado sep 05, 2022, de Integral: <https://la.mathworks.com/help/matlab/ref/integral.html>
- MathWorks. (s.f.). MATLAB Online. Recuperado de <https://la.mathworks.com/products/matlab-online.html>
- McNeill, A. (2021). PLUS Bringing Mathematics to Life. Recuperado de Modeling, step by step: <https://plus.maths.org/content/os/issue13/features/walking/index>
- Mendezabal, M. J., & Tindowen, D. J. (2018). Improving students' attitude, conceptual understanding and procedural skills in differential calculus through Microsoft mathematics. JOTSE, 8(4), 385-397. Obtenido de <https://dialnet.unirioja.es/servlet/articulo?codigo=6623142>
- Meneses, G. (2007). El proceso de enseñanza – aprendizaje: el acto didáctico. Obtenido de UNIVERSITAT ROVIRA I VIRGILI: <https://www.tdx.cat/bitstream/handle/10803/8929/Elprocesodeensenanza.pdf>
- Ministerio de Educación. (2018). Física 1 BGU. Ecuador: Don Bosco obras salesianas de comunicación. Obtenido de https://educacion.gob.ec/wp-content/uploads/downloads/2016/09/Curriculo/FISICA/Fisica_1_BGU.pdf
- Molina, J. (2017). Experiencia de modelación matemática como estrategia didáctica para la enseñanza de tópicos de cálculo. Uniciencia, 31(2), 19-39. Obtenido de <https://www.redalyc.org/comocitar.oa?id=475952089002>
- Mora, D. (2003). Estrategias para el aprendizaje y la enseñanza de las matemáticas. Revista de Pedagogía, 27(40), 181-272. Recuperado de http://ve.scielo.org/scielo.php?script=sci_arttext&pid=S0798-97922003000200002
- Morales, J., & Peña, L. (2013). PROPUESTA METODOLÓGICA PARA LA ENSEÑANZA DEL CÁLCULO EN INGENIERÍA, BASADA EN LA MODELACIÓN MATEMÁTICA. VII CIBEM, 577-587. Recuperado de <https://core.ac.uk/download/pdf/328835257.pdf>

- Moreno, T. (2019). Didáctica de la Educación Superior: Nuevos Desafíos en el Siglo XXI. *Perspectiva Educativa*, 50(2), 26-54. doi:<https://www.redalyc.org/articulo.oa?id=333327290003>
- Muente, G. (28 de Abril de 2019). Software Educativo: Un Pilar de la Enseñanza Digital. Obtenido de [rockcontent.com](https://rockcontent.com/es/blog/software-educativo/#:~:text=%C2%BFQu%C3%A9%20es%20un%20software%20educativo,facilitando%20la%20adquisici%C3%B3n%20de%20conocimientos): <https://rockcontent.com/es/blog/software-educativo/#:~:text=%C2%BFQu%C3%A9%20es%20un%20software%20educativo,facilitando%20la%20adquisici%C3%B3n%20de%20conocimientos>.
- Peña, C. (2015). La Integración Numérica como recurso para el cálculo de la integral definida. Tesis de Maestría, Universidad Nacional de Colombia, Bogotá. Recuperado de <https://www.pdfdrive.com/la-integraci%C3%B3n-num%C3%A9rica-como-recurso-para-el-c%C3%A1culo-de-la-integral-definida-d34078320.html>
- Pere, M. (12 de Septiembre de 2017). Universidad Autónoma de Barcelona. Obtenido de El software educativo: http://www.lmi.ub.es/te/any96/marques_software/
- Perez, J., & Gardey, A. (2016). Definición de software educativo. Definicion de. Obtenido de definicion.de/software-educativo/
- Perez, S. (2018). Influencia de la estrategia didáctica “planificación - ejecución” en el nivel de desarrollo de las capacidades del área de matemática en los estudiantes de tercer grado de educación secundaria de la Institución Educativa El Cumbe – Cutervo-2014. Tesis de Doctorado, UNIVERSIDAD NACIONAL DE EDUCACIÓN Enrique Guzmán y Valle Alma Mater del Magisterio Nacional ESCUELA DE POSGRADO, Lima-Perú. Obtenido de <https://repositorio.une.edu.pe/bitstream/handle/UNE/3428/TD%20CE%202111%20P1%20-%20Perez%20Perez%20Segundo%20Santos%20.pdf?sequence=1&isAllowed=y>
- Plaza, L. (Octubre de 2016). Modelación matemática en ingeniería. *IE Revista de investigación educativa de la REDIECH*, 7(13), 47-57. Obtenido de https://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S2448-85502016000200047&fbclid=IwAR3jx6_SY_48whbWLqKlhve-hU23B7VnhOr-APoG9Wfp3VC25CuBdnvi2KI
- Plaza, L. (Octubre de 2016). Modelación Matemática en Ingeniería. *IE Revista de investigación educativa de la REDIECH*, 7(13), 47-57. Obtenido de https://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S2448-85502016000200047&fbclid=IwAR3jx6_SY_48whbWLqKlhve-hU23B7VnhOr-APoG9Wfp3VC25CuBdnvi2KI
- PUCP. (Octubre de 2019). MATLAB Y SIMULINK. Obtenido de Departamento de Ingeniería: <https://departamento.pucp.edu.pe/ingenieria/matlab/que-es-matlab/>
- QueGrande ORG. (2010). Practica 3: Cálculo Integral con MatLab. En Curso 2010-2011 (págs. 1-13). Obtenido de http://quegrande.org/apuntes/grado/1G/CalG/practicas/10-11/practica_3_-_calculo_integral.pdf
- Ramirez, C. (14 de Mayo de 2019). Características del software Educativo. doi:<https://blog.comparasoftware.com/caracteristicas-de-un-software-educativo/>

- Rodriguez, C. (Octubre de 2015). ELABORACIÓN DE INSTRUMENTOS PARA LA RECOLECCIÓN DE DATOS. Obtenido de <http://ri.uaemex.mx/oca/view/20.500.11799/34900/1/secme-20396.pdf>
- Rodríguez, D., & Gairín, J. (2009). Presentación. *Educación*, 47(1), 9-12. Obtenido de Facultad de Ciencias de la Educación: <https://www.redalyc.org/comocitar.oa?id=342130836001>
- Rodríguez, R. (2017). Repensando la enseñanza de las matemáticas para futuros ingenieros: actualidades y desafíos. *IE Revista de investigación educativa de la REDIECH*, 8(15), 69-85. Recuperado de https://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S2448-85502017000200069&fbclid=IwAR38k7DeTSdou-2I93xt2dh5SmTVPWH-mSME2tgDJPr6bCiWfWXZm-4FkH8
- Rovira, I. (26 de Octubre de 2019). *Psicología y Mente*. Obtenido de Software educativo: tipos, características y usos: <https://psicologiamente.com/desarrollo/software-educativo>
- Sánchez, A. (15 de Septiembre de 2021). Definición de Volumen. Recuperado el 01 de dic de 2021, de <https://conceptodefinicion.de/volumen/>
- Sanhermelando, J. (22 de Enero de 2019). El Español. Obtenido de Cómo calcular el volumen de las figuras geométricas: https://www.lespanol.com/como/calcular-volumen-figuras-geometricas/370213178_0.html
- SECUMATE. (12 de Noviembre de 2011). INTRODUCCIÓN BÁSICA A LOS ALGORITMOS. Obtenido de <https://sites.google.com/site/lestadmate/1-introduccion-basica-a-los-algoritmos>
- Trejo, T. E., Camarena, G. P., & Trejo, T. N. (2013). Las matemáticas en la formación de un ingeniero: la matemática en contexto como propuesta metodológica. *Revista de Docencia Universitaria*, 11(1), 397-424. Obtenido de <https://dialnet.unirioja.es/servlet/articulo?codigo=4522470>
- Universidad de los Andes (UniAndes). (s.f.). Tutorial de Vensim: un modelo de población. Recuperado agosto 30, 2022, de <https://misovirtual.virtual.uniandes.edu.co/codelabs/MIAD-Din%C3%A1micaSistemas-TutorialPoblaci%C3%B3n/index.html?index=..%2F..index#7>
- Universidad Nacional Autónoma de México. (s.f.). Volumen de un Sólido de Revolución. Recuperado agosto 09, 2022, de http://prepa8.unam.mx/academia/colegios/matematicas/paginacolmate/applets/matematicas_VI_12/Applets_Geogebra/volusolirevo.html
- Ventana Systems, Inc. (2007). Guía del Usuario de Vensim. (J. García, Trad.) Obtenido de <http://ia902807.us.archive.org/27/items/VensimGuia/VensimGuia.pdf>
- Vilca, R. (2019). APLICACIÓN DEL SOFTWARE GEOGEBRA Y SU INFLUENCIA EN EL APRENDIZAJE DE ÁREAS Y VOLÚMENES DE SÓLIDOS DE REVOLUCIÓN EN EL CÁLCULO INTEGRAL EN LOS ESTUDIANTES DEL PRIMER AÑO DE LA FACULTAD DE INGENIERÍAS DE LA UNIVERSIDAD CONTINENTAL AREQUIPA - 2017. Tesis de Maestría. Obtenido de

[http://repositorio.unsa.edu.pe/bitstream/handle/UNSA/8427/EDMvipar.pdf?sequence=1
&isAllowed=y](http://repositorio.unsa.edu.pe/bitstream/handle/UNSA/8427/EDMvipar.pdf?sequence=1&isAllowed=y)

Westreicher, G. (08 de Marzo de 2021). Economipedia. Obtenido de Cálculo:
<https://economipedia.com/definiciones/calculo.html>

ANEXOS

ANEXO A. Estructuras básicas de programación

Existen tres estructuras principales en la programación de Matlab: if, for, while (Mathworks, nd).

Estas pueden usarse por separado o combinarse para formar los siguientes pseudocódigos (Chapra & Canale, 2007, p. 45).

Tabla 1-Anexo: Estructuras de control fundamentales en Pseudocódigo y lenguaje de programación en Matlab

Pseudocódigo	MATLAB
IF/THEN	
<i>IF condición THEN</i> <i>Bloque verdadero</i> <i>ENDIF</i>	if b ~= 0 r1 = -c / b; end
IF/THEN/ELSE	
<i>IF condición THEN</i> <i>Bloque verdadero</i> <i>ELSE</i> <i>Bloque falso</i> <i>ENDIF</i>	if a < 0 b = sqrt(abs(a)); else b = sqrt(a); end
IF/THEN/ELSEIF	
<i>IF condición1 THEN</i> <i>Bloque1</i> <i>ELSEIF condición2</i> <i>Bloque2</i> <i>ELSE condición3</i> <i>Bloque3</i> <i>ENDIF</i>	if class == 1 x=x+8; elseif class < 1 x=x-8; else x=x-64; end
LOOP CONTROLADO POR CONTADOR:	
<i>DOFOR i=inicio, incremento, fin</i> <i>Bloque</i> <i>END</i>	for i=1:0.1:2 x=x+i end
DO/EXIT	
<i>DO</i> <i>Bloque1</i> <i>IF condición EXIT</i> <i>Bloque2</i> <i>END</i>	while (1) i=i+1; if i >= 10, break, end j=i*x; end

Fuente: (Chapra & Canale, 2007, p. 40).

En resumen, el bucle *if* se utiliza para establecer condiciones y consecuencias, en el esquema IFTHEN; si hay dos o más condiciones, se usa el esquema IF/THEN/ELSE. El bucle *for* se utiliza para especificar acciones repetitivas, enumeradas como lo especifique el bucle, en el esquema LOOP controlado por contador. El bucle *while* permite una combinación ente *for* e *if*, para cumplir con determinadas operaciones mientras se cumpla una condición dada, trabajando bajo el esquema DO/EXIT.

ANEXO B. Ejemplos de cálculo de volumen de sólido de revolución usando el programa.

Ejercicio 1: Sea $y = 4x^2$, entre $x = 1$ & $x = 5$. Formular y evaluar la integral del solido formado al girar la región alrededor del eje x.

Primero, ejecutar el archivo MENU_PRINCIPAL.m escoger la opción “Función generatriz f(x)”:



Figura 1-Anexo: Menú principal del programa desarrollado en MATLAB

Realizado por: Bonilla, J. 2022

Luego, llenar los campos y darle click a proceso, entonces se mostrará lo siguiente:

interfaz1

M

Macstría en Matemática Mención: Modelación y Docencia
Función Generatriz $f(x)$

Volumen de sólido de revolución para $f(x)$

Función

$f(x) =$

Límite inferior: $a =$

Límite superior: $b =$

Eje de revolución

Elija 1

x $x = k$; donde $k =$

y $y = k$; donde $k =$

$y = m*x + k$; donde $m =$ y donde $k =$

En este caso se utiliza el método de discos debido a que los diferenciales de Area de la función son perpendiculares al eje de rotación. Por lo que al rotar sobre el mismo. Se generan discos en el espacio.

Cálculo

Cálculo del volumen del sólido de revolución (Método discos)

$$dv = \pi r^2 dy$$

$$V = \int_1^5 16 \pi x^4 dy$$

$$V = \int_1^5 16 \pi x^4 dy$$

$$V = \int_1^5 16 \pi x^4 dy$$

$$V = \frac{16 \pi x^5}{5} \Big|_1^5$$

$$V = \frac{49984 \pi}{5}$$

$$V = 31405.9 \text{ u}^3$$

Figura 2-Anexo: Solución analítica del ejercicio planteado para cálculo de volumen del sólido de revolución en el programa desarrollado en MATLAB

Realizado por: Bonilla, J. 2022

En este caso, el ejercicio corresponde a una integral por el método de discos en torno al eje x (ver ec. (8)), y los cálculos a realizar son presentados en la interfaz gráfica de Matlab, como se mostró en la Figura 2-Anexo. También se podrá visualizar la gráfica bidimensional o tridimensional del sólido generado por la función generatriz al girar sobre el eje establecido, y ente los límites seleccionados, como en la Figura 3-Anexo al darle click en los botones correspondientes:

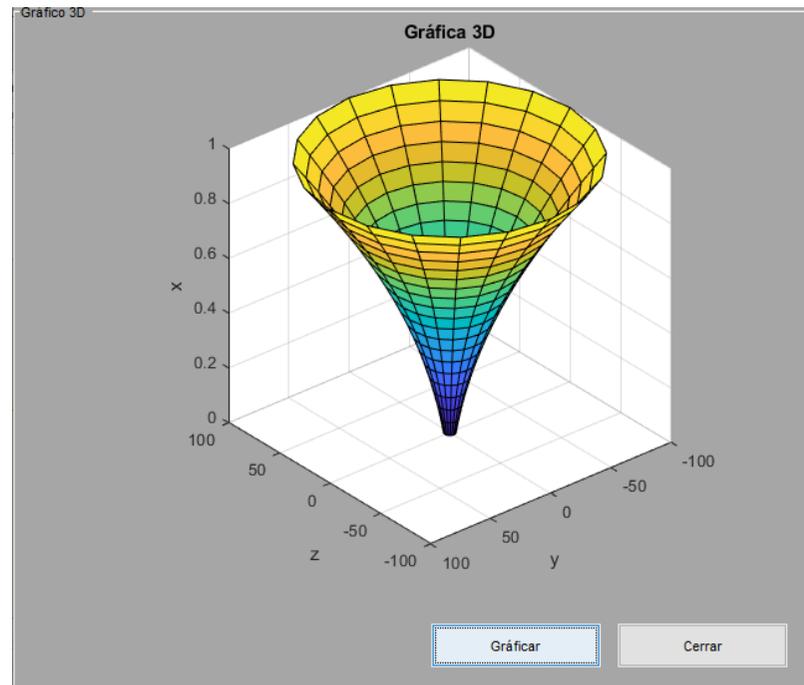


Figura 3-Anexo: Solución gráfica 3D del ejercicio planteado para cálculo de volumen del sólido de revolución en el programa desarrollado en MATLAB

Realizado por: Bonilla, J. 2022

Ejercicio 2: Sean las funciones $y = x^2$ y $y = x$, determine el volumen del sólido de revolución que se forma al girar el área comprendida entre ambas funciones, alrededor del eje X.

Similar al ejercicio anterior, primero se ingresará cada una de las funciones, este caso corresponde a la tercera opción “Función Generatriz entre dos Funciones $f(x)$ y $g(x)$ ” de la pantalla principal del programa (ver Figura 1-Anexo). Luego se graficarán las funciones en 2D, como en la Figura 4-Anexo.

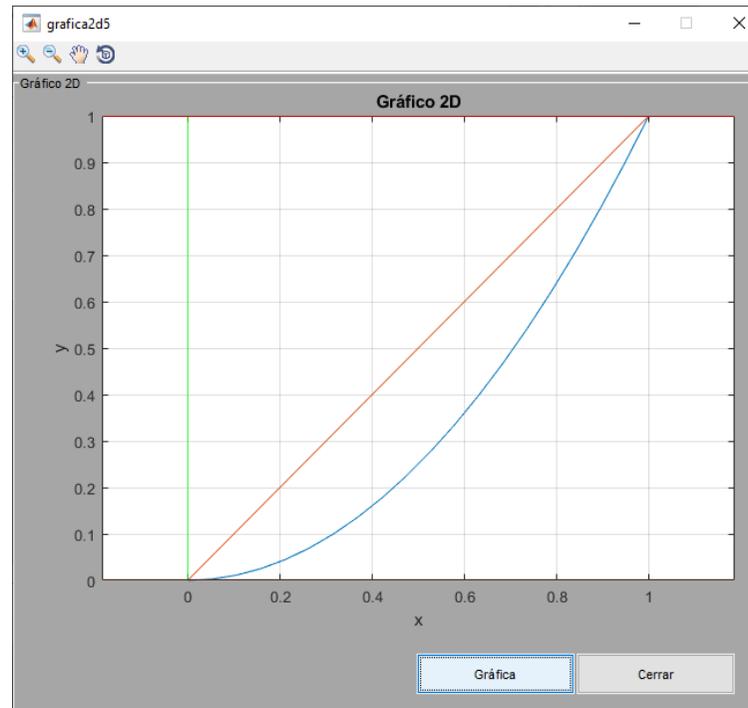


Figura 4-Anexo: Gráfica 2D del ejercicio planteado para cálculo de volumen del sólido de revolución en el programa desarrollado en MATLAB

Realizado por: Bonilla, J. 2022

También se mostrará la resolución del ejercicio al dar clic en el botón “proceso”:

The screenshot shows a MATLAB window titled 'interfaz5' with a header for 'Maestría en Matemática Mención: Modelación y Docencia' and 'Función Generatriz entre dos Funciones f(x) y g(x)'. The main area is divided into two sections: 'Función' and 'Eje de revolución'. In the 'Función' section, $f(x) = x^2$ and $g(x) = x$ are entered, with limits $a = 0$ and $b = 1$. The 'Eje de revolución' section has $y = k$ selected with $k = 0$. A 'Proceso' button is highlighted, leading to the 'Proceso' section which shows the analytical solution for the volume of the solid of revolution using the disk method. The solution involves the integral $V = \int_0^1 \pi (R^2 - r^2) dx$, where $R = x^2$ and $r = x$. The final result is $V = 0.418879$.

Figura 5-Anexo: Solución analítica del segundo ejercicio planteado para cálculo de volumen del sólido de revolución en el programa desarrollado en MATLAB.

Realizado por: Bonilla, J. 2022

Y al final, se graficará dicho sólido de revolución en 3D, como en la Figura 6-Anexo:

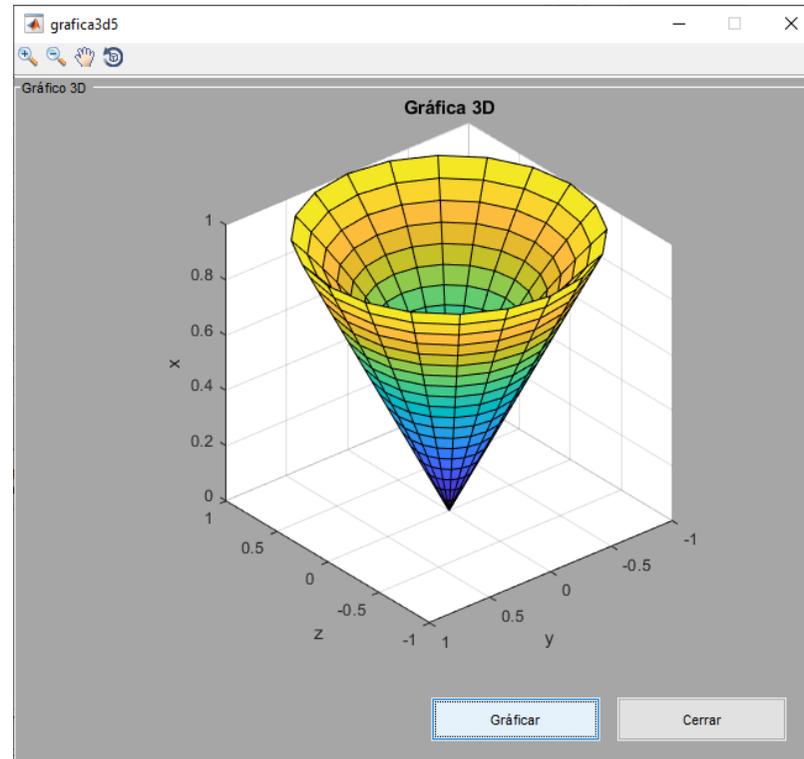


Figura 6-Anexo: Solución gráfica 3D del segundo ejercicio planteado para cálculo de volumen del sólido de revolución en el programa desarrollado en MATLAB

Realizado por: Bonilla, J. 2022

ANEXO C. Código implementado en el desarrollo del programa

A continuación, una muestra de todos los *scripts* implementados en la programación de la propuesta. El programa completo se encuentra disponible en:

https://drive.google.com/drive/folders/1vzwSmtau6DE3aa1Q56Qwytq4G_iwgeSw?usp=sharing

Menú principal.

```
function varargout = MENU_PRINCIPAL(varargin)
% MENU_PRINCIPAL MATLAB code for MENU_PRINCIPAL.fig
%   MENU_PRINCIPAL, by itself, creates a new MENU_PRINCIPAL or
%   raises the existing
%   singleton*.
%
%   H = MENU_PRINCIPAL returns the handle to a new
MENU_PRINCIPAL or the handle to
%   the existing singleton*.
%
%   MENU_PRINCIPAL('CALLBACK',hObject,eventData,handles,...)
calls the local
%   function named CALLBACK in MENU_PRINCIPAL.M with the given
input arguments.
%
%   MENU_PRINCIPAL('Property','Value',...) creates a new
MENU_PRINCIPAL or raises the
%   existing singleton*. Starting from the left, property value
pairs are
%   applied to the GUI before MENU_PRINCIPAL_OpeningFcn gets
called. An
%   unrecognized property name or invalid value makes property
application
%   stop. All inputs are passed to MENU_PRINCIPAL_OpeningFcn
via varargin.
%
%   *See GUI Options on GUIDE's Tools menu. Choose "GUI allows
only one
%   instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help MENU_PRINCIPAL

% Last Modified by GUIDE v2.5 18-May-2022 18:49:07

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @MENU_PRINCIPAL_OpeningFcn, ...
                  'gui_OutputFcn',  @MENU_PRINCIPAL_OutputFcn, ...
                  'gui_LayoutFcn',  [] , ...
                  'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end
```

```

if nargin
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before MENU_PRINCIPAL is made visible.
function MENU_PRINCIPAL_OpeningFcn(hObject, eventdata, handles,
varargin)
axes('Units','Normalized','Position',[0 0 1 1]);

[x, map]=imread('IPEC_FONDO.png','png');
image(x)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to MENU_PRINCIPAL (see
VARARGIN)
[a,map]=imread('Video.JPG'); %Leemos imagen a mostrar.
[r,c,d]=size(a);
S=get(handles.pushbutton7,'Position'); %Preguntamos por el tamaño
del botón a personalizar
x=ceil(r/100); %Si quieres que la imagen tenga el tamaño del botón,
usa h aquí, si no, pon otro valor.
y=ceil(c/200); %Si quieres que la imagen tenga el tamaño del botón,
usa w aquí, si no, pon otro valor.
g=a(1:x:end,1:y:end,:);
g(g==255)=0.8*255;
set(handles.pushbutton7,'CData',g); %Añadimos imagen al botón.

% Choose default command line output for MENU_PRINCIPAL
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes MENU_PRINCIPAL wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = MENU_PRINCIPAL_OutputFcn(hObject, eventdata,
handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

```

```

% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
interfaz1

% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
interfaz2

% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
interfaz5

% --- Executes on button press in pushbutton4.
function pushbutton4_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton4 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
interfaz6

% --- Executes on button press in pushbutton5.
function pushbutton5_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton5 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
interfaz7

% --- Executes on button press in radiobutton1.

% --- Executes on button press in pushbutton6.
function pushbutton6_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton6 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
close(MENU_PRINCIPAL);

% --- Executes on button press in pushbutton7.
function pushbutton7_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton7 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
web('https://youtu.be/Qrr4dahmNRM')

```

Interfaz.

```
function varargout = interfaz1(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @interfaz1_OpeningFcn, ...
                  'gui_OutputFcn',  @interfaz1_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

function interfaz1_OpeningFcn(hObject, eventdata, handles,
varargin)
axes('Units','Normalized','Position',[0 0 1 1]);

[x, map]=imread('il.png','png');
image(x)
handles.output = hObject;
guidata(hObject, handles);
axes(handles.axes2);
back=imread('integral.jpeg');
imshow(back);
axes(handles.axes3);
imshow(back);
axes(handles.axes4);
imshow(back);

function varargout = interfaz1_OutputFcn(hObject, eventdata,
handles)
varargout{1} = handles.output;

function uibuttongroup1_SelectionChangedFcn(hObject, eventdata,
handles)
if hObject == handles.uno
    handles.eje_rot = 1;
end
if hObject == handles.dos
    handles.eje_rot = 2;
end
if hObject == handles.tres
    handles.eje_rot = 3;
end
if hObject == handles.cuatro
    handles.eje_rot = 4;
end
if hObject == handles.cinco
    handles.eje_rot = 5;
```

```
end
guidata(hObject,handles);

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

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

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

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

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

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

function edit7_Callback(hObject, eventdata, handles)
function edit7_CreateFcn(hObject, eventdata, handles)
```

```

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function pushbutton4_Callback(hObject, eventdata, handles)
close(interfaz1);
MENU_PRINCIPAL;

function pushbutton3_Callback(hObject, eventdata, handles)
if isempty(get(handles.edit1,'String')) == 1
    errordlg('Debe ingresar una función','Atención','modal');
else
    y = str2sym(get(handles.edit1,'String')); %Se declara y como
simbólica.
    sym x;
end
if isempty(get(handles.edit2,'String')) == 1
    errordlg('Debe ingresar a','Atención','modal');
else
    a=str2num(get(handles.edit2,'String'));
end
if isempty(get(handles.edit3,'String')) == 1
    errordlg('Debe ingresar b','Atención','modal');
else
    b=str2num(get(handles.edit3,'String'));
end

if ~contains(char(y),'x') &&
~isreal(y) %Si la función no
contiene x y no es real.
    errordlg('Debe ingresar una función de x.','Atención','modal');
end
if
a>b
%Si a es mayor que b.
    errordlg('a tiene que ser menor que b.','Atención','modal');
end
if ~isreal(a) || isnan(a) ||
isinf(a) %Si a no es un número
real.
    errordlg('a tiene que ser real y diferente de
infinito.','Atención','modal');
end
if ~isreal(b) || isnan(b) ||
isinf(b) %Si b no es un número
real.
    errordlg('b tiene que ser real y diferente de
infinito.','Atención','modal');
end
if isnan(int(y,a,b)) ||
~isreal(int(y,a,b)) %Si la función no es
continua o no real.
    errordlg('La función no es continua en el intervalo
[a,b]','Atención','modal');
end

```

```

switch handles.eje_rot
    case 1
        %Cálculo de volumen
        set(handles.l1,'String','Cálculo del volumen del sólido de
        revolución (Método discos)');
        set(handles.l2,'String','-----
        -----');
        set(handles.mensaje,'String','En este caso se utiliza el
        método de discos debido a que los diferenciales de Area de la
        función son perpendiculares al eje de rotación. Por lo que al rotar
        sobre el mismo. Se generan discos en el espacio.');
```

$$v_2 = \int_a^b \pi y^2 \, dy$$

```

        syms r;
        v2=int(pi*y^2,a,b);
        if v2>0
            outstr = evalc('pretty([pi*(r)^2])');
            outstrcell = regexp(outstr,'\n','split');
            set(handles.l3,'String',outstrcell);
            outstr1 = evalc('pretty((pi*y^2))');
            outstrcell1 = regexp(outstr1,'\n','split');
            set(handles.limsup1,'String',num2str(b));
            set(handles.limsup2,'String',num2str(b));
            set(handles.limsup3,'String',num2str(b));
            set(handles.limsup4,'String',num2str(b));
            set(handles.liminf1,'String',num2str(a));
            set(handles.liminf2,'String',num2str(a));
            set(handles.liminf3,'String',num2str(a));
            set(handles.liminf4,'String',num2str(a));
            set(handles.l4,'String',outstrcell1);
            set(handles.l5,'String',outstrcell1);
            set(handles.l6,'String',outstrcell1);
            v1=int(pi*y^2);
            outstr2 = evalc('pretty(v1)');
            outstrcell2 = regexp(outstr2,'\n','split');
            set(handles.l7,'String',outstrcell2);
            v2=int(pi*y^2,a,b);
            outstr3 = evalc('pretty(v2)');
            outstrcell3 = regexp(outstr3,'\n','split');
            set(handles.l8,'String',outstrcell3);
            v2=double(v2);
            set(handles.l9,'String',v2);
            syms u;
            g=u^3;
            outstr5 = evalc('pretty(g)');
            outstrcell5 = regexp(outstr5,'\n','split');
            set(handles.unidades1,'String',outstrcell5);
            set(handles.unidades2,'String',outstrcell5);
        else
            outstr = evalc('pretty([-pi*(r)^2])');
            outstrcell = regexp(outstr,'\n','split');
            set(handles.l3,'String',outstrcell);
            outstr1 = evalc('pretty((-pi*y^2))');
            outstrcell1 = regexp(outstr1,'\n','split');
            set(handles.limsup1,'String',num2str(b));
            set(handles.limsup2,'String',num2str(b));

```

```

        set(handles.limsup3,'String',num2str(b));
        set(handles.limsup4,'String',num2str(b));
        set(handles.liminf1,'String',num2str(a));
        set(handles.liminf2,'String',num2str(a));
        set(handles.liminf3,'String',num2str(a));
        set(handles.liminf4,'String',num2str(a));
        set(handles.l4,'String',outstrcell1);
        set(handles.l5,'String',outstrcell1);
        set(handles.l6,'String',outstrcell1);
        v1=-int(pi*y^2);
        outstr2 = evalc('pretty(v1)');
        outstrcell2 = regexp(outstr2,'\n','split');
        set(handles.l7,'String',outstrcell2);
        v2=-int(pi*y^2,a,b);
        outstr3 = evalc('pretty(v2)');
        outstrcell3 = regexp(outstr3,'\n','split');
        set(handles.l8,'String',outstrcell3);
        v2=double(v2);
        set(handles.l9,'String',v2);
        syms u;
        g=u^3;
        outstr5 = evalc('pretty(g)');
        outstrcell5 = regexp(outstr5,'\n','split');
        set(handles.unidades1,'String',outstrcell5);
        set(handles.unidades2,'String',outstrcell5);
    end
    case 2
        %Cálculo de volumen
        set(handles.l11,'String','Cálculo del volumen del sólido de
    revolución (Método capas cilindricas)');
        set(handles.l12,'String','-----
    -----');
        set(handles.mensaje,'String','En este caso se utiliza el
    método de capas cilíndricas debido a que los diferenciales de Area
    de la función son paralelos al eje de rotación. Por lo que al rotar
    sobre el mismo. Se generan capas de cilindros en el espacio.');
```

```

        syms f(x);
        v2=int(2*pi*x*y,a,b);
        if v2>0
            outstr = evalc('pretty(2*pi*x*f(x))');
            outstrcell = regexp(outstr,'\n','split');
            set(handles.l13,'String',outstrcell);
            outstr1 = evalc('pretty(2*pi*x*y)');
            outstrcell1 = regexp(outstr1,'\n','split');
            set(handles.limsup1,'String',num2str(b));
            set(handles.limsup2,'String',num2str(b));
            set(handles.limsup3,'String',num2str(b));
            set(handles.limsup4,'String',num2str(b));
            set(handles.liminf1,'String',num2str(a));
            set(handles.liminf2,'String',num2str(a));
            set(handles.liminf3,'String',num2str(a));
            set(handles.liminf4,'String',num2str(a));
            set(handles.l4,'String',outstrcell1);
            set(handles.l5,'String',outstrcell1);
        end
    end
end

```

```

set(handles.l16,'String',outstrcell1);
syms x;
v1=int(2*pi*x*y);
outstr2 = evalc('pretty(v1)');
outstrcell2 = regexp(outstr2,'\n','split');
set(handles.l17,'String',outstrcell2);
v2=int(2*pi*x*y,a,b);
outstr3 = evalc('pretty(v2)');
outstrcell3 = regexp(outstr3,'\n','split');
set(handles.l18,'String',outstrcell3);
v2=double(v2);
set(handles.l19,'String',v2);
syms u;
g=u^3;
outstr5 = evalc('pretty(g)');
outstrcell5 = regexp(outstr5,'\n','split');
set(handles.unidades1,'String',outstrcell5);
set(handles.unidades2,'String',outstrcell5);
else
outstr = evalc('pretty(-2*pi*x*f(x))');
outstrcell = regexp(outstr,'\n','split');
set(handles.l13,'String',outstrcell);
outstr1 = evalc('pretty(-2*pi*x*y)');
outstrcell1 = regexp(outstr1,'\n','split');
set(handles.limsup1,'String',num2str(b));
set(handles.limsup2,'String',num2str(b));
set(handles.limsup3,'String',num2str(b));
set(handles.limsup4,'String',num2str(b));
set(handles.liminf1,'String',num2str(a));
set(handles.liminf2,'String',num2str(a));
set(handles.liminf3,'String',num2str(a));
set(handles.liminf4,'String',num2str(a));
set(handles.l14,'String',outstrcell1);
set(handles.l15,'String',outstrcell1);
set(handles.l16,'String',outstrcell1);
syms x;
v1=-int(2*pi*x*y);
outstr2 = evalc('pretty(v1)');
outstrcell2 = regexp(outstr2,'\n','split');
set(handles.l17,'String',outstrcell2);
v2=-int(2*pi*x*y,a,b);
outstr3 = evalc('pretty(v2)');
outstrcell3 = regexp(outstr3,'\n','split');
set(handles.l18,'String',outstrcell3);
v2=double(v2);
set(handles.l19,'String',v2);
syms u;
g=u^3;
outstr5 = evalc('pretty(g)');
outstrcell5 = regexp(outstr5,'\n','split');
set(handles.unidades1,'String',outstrcell5);
set(handles.unidades2,'String',outstrcell5);
end
case 3

```

```

if isempty(get(handles.edit7,'String')) == 1
    errordlg('Debe ingresar k','Atención','modal');
else
    k = str2num(get(handles.edit7,'String'));
end
if ~isreal(k) || isnan(k) ||
isinf(k) %Si k no es un número
real.
    errordlg('"k" eje de rotación tiene que ser real y
diferente de infinito.','Atención','modal'); %Se manda
error.
end
set(handles.mensaje,'String','En este caso se utiliza el
método de capas cilíndricas debido a que los diferenciales de Area
de la función son paralelos al eje de rotación. Por lo que al rotar
sobre el mismo. Se generan capas de cilindros en el espacio.');
```

```

if k > 0
    set(handles.l1,'String','Cálculo del volumen del sólido
de revolución (Método capas cilíndricas)');
    set(handles.l2,'String','-----
-----');
    syms f(x);
if int(2*pi*(k-x)*y,a,b)>0
    outstr = evalc('pretty(2*pi*(k-x)*f(x))');
    outstrcell = regexp(outstr,'\n','split');
    set(handles.l3,'String',outstrcell);
    outstr1 = evalc('pretty((2*pi*(k-x)*y))');
    outstrcell1 = regexp(outstr1,'\n','split');
    set(handles.limsup1,'String',num2str(b));
    set(handles.limsup2,'String',num2str(b));
    set(handles.limsup3,'String',num2str(b));
    set(handles.limsup4,'String',num2str(b));
    set(handles.liminf1,'String',num2str(a));
    set(handles.liminf2,'String',num2str(a));
    set(handles.liminf3,'String',num2str(a));
    set(handles.liminf4,'String',num2str(a));
    set(handles.l4,'String',outstrcell1);
    set(handles.l5,'String',outstrcell1);
    set(handles.l6,'String',outstrcell1);
    syms x;
    v1=int(2*pi*(k-x)*y);
    outstr2 = evalc('pretty(v1)');
    outstrcell2 = regexp(outstr2,'\n','split');
    set(handles.l7,'String',outstrcell2);
    v2=int(2*pi*(k-x)*y,a,b);
    outstr3 = evalc('pretty(v2)');
    outstrcell3 = regexp(outstr3,'\n','split');
    set(handles.l8,'String',outstrcell3);
    v2=double(v2);
    set(handles.l9,'String',v2);
    syms u;
    g=u^3;
    outstr5 = evalc('pretty(g)');
    outstrcell5 = regexp(outstr5,'\n','split');
    set(handles.unidades1,'String',outstrcell5);

```

```

        set(handles.unidades2,'String',outstrcell5);
    else
        outstr = evalc('pretty(-2*pi*(k-x)*f(x))');
        outstrcell = regexp(outstr,'\n','split');
        set(handles.l13,'String',outstrcell);
        outstr1 = evalc('pretty(-(2*pi*(k-x)*y))');
        outstrcell1 = regexp(outstr1,'\n','split');
        set(handles.limsup1,'String',num2str(b));
        set(handles.limsup2,'String',num2str(b));
        set(handles.limsup3,'String',num2str(b));
        set(handles.limsup4,'String',num2str(b));
        set(handles.liminf1,'String',num2str(a));
        set(handles.liminf2,'String',num2str(a));
        set(handles.liminf3,'String',num2str(a));
        set(handles.liminf4,'String',num2str(a));
        set(handles.l14,'String',outstrcell1);
        set(handles.l15,'String',outstrcell1);
        set(handles.l16,'String',outstrcell1);
        syms x;
        v1=-int(2*pi*(k-x)*y);
        outstr2 = evalc('pretty(v1)');
        outstrcell2 = regexp(outstr2,'\n','split');
        set(handles.l17,'String',outstrcell2);
        v2=-int(2*pi*(k-x)*y,a,b);
        outstr3 = evalc('pretty(v2)');
        outstrcell3 = regexp(outstr3,'\n','split');
        set(handles.l18,'String',outstrcell3);
        v2=double(v2);
        set(handles.l19,'String',v2);
        syms u;
        g=u^3;
        outstr5 = evalc('pretty(g)');
        outstrcell5 = regexp(outstr5,'\n','split');
        set(handles.unidades1,'String',outstrcell5);
        set(handles.unidades2,'String',outstrcell5);
    end
else
    %Cálculo de volumen
    set(handles.l11,'String','Cálculo del volumen del sólido
de revolución (Método capas cilindricas)');
    set(handles.l12,'String','-----
-----');
    syms f(x);
    v2=int(2*pi*(x-k)*y,a,b);
    if v2>0
        outstr = evalc('pretty(2*pi*(x-k)*f(x))');
        outstrcell = regexp(outstr,'\n','split');
        set(handles.l13,'String',outstrcell);
        outstr1 = evalc('pretty(2*pi*(x-k)*y)');
        outstrcell1 = regexp(outstr1,'\n','split');
        set(handles.limsup1,'String',num2str(b));
        set(handles.limsup2,'String',num2str(b));
        set(handles.limsup3,'String',num2str(b));
        set(handles.limsup4,'String',num2str(b));
    end
end

```

```

set(handles.liminf1,'String',num2str(a));
set(handles.liminf2,'String',num2str(a));
set(handles.liminf3,'String',num2str(a));
set(handles.liminf4,'String',num2str(a));
set(handles.l14,'String',outstrcell1);
set(handles.l15,'String',outstrcell1);
set(handles.l16,'String',outstrcell1);
syms x;
v1=int(2*pi*(x-k)*y);
outstr2 = evalc('pretty(v1)');
outstrcell2 = regexp(outstr2,'\n','split');
set(handles.l17,'String',outstrcell2);
v2=int(2*pi*(x-k)*y,a,b);
outstr3 = evalc('pretty(v2)');
outstrcell3 = regexp(outstr3,'\n','split');
set(handles.l18,'String',outstrcell3);
v2=double(v2);
set(handles.l19,'String',v2);
syms u;
g=u^3;
outstr5 = evalc('pretty(g)');
outstrcell5 = regexp(outstr5,'\n','split');
set(handles.unidades1,'String',outstrcell5);
set(handles.unidades2,'String',outstrcell5);
else
outstr = evalc('pretty(-2*pi*(x-k)*f(x))');
outstrcell = regexp(outstr,'\n','split');
set(handles.l13,'String',outstrcell);
outstr1 = evalc('pretty(-2*pi*(x-k)*y)');
outstrcell1 = regexp(outstr1,'\n','split');
set(handles.limsup1,'String',num2str(b));
set(handles.limsup2,'String',num2str(b));
set(handles.limsup3,'String',num2str(b));
set(handles.limsup4,'String',num2str(b));
set(handles.liminf1,'String',num2str(a));
set(handles.liminf2,'String',num2str(a));
set(handles.liminf3,'String',num2str(a));
set(handles.liminf4,'String',num2str(a));
set(handles.l14,'String',outstrcell1);
set(handles.l15,'String',outstrcell1);
set(handles.l16,'String',outstrcell1);
syms x;
v1=-int(2*pi*(x-k)*y);
outstr2 = evalc('pretty(v1)');
outstrcell2 = regexp(outstr2,'\n','split');
set(handles.l17,'String',outstrcell2);
v2=-int(2*pi*(x-k)*y,a,b);
outstr3 = evalc('pretty(v2)');
outstrcell3 = regexp(outstr3,'\n','split');
set(handles.l18,'String',outstrcell3);
v2=double(v2);
set(handles.l19,'String',v2);
syms u;
g=u^3;

```

```

        outstr5 = evalc('pretty(g)');
        outstrcell5 = regexp(outstr5, '\n', 'split');
        set(handles.unidades1, 'String', outstrcell5);
        set(handles.unidades2, 'String', outstrcell5);
    end
end
case 4
    if isempty(get(handles.edit6, 'String')) == 1
        errordlg('Debe ingresar k', 'Atención', 'modal');
    else
        k = str2num(get(handles.edit6, 'String'));
    end
    if ~isreal(k) || isnan(k) ||
isinf(k) %Si k no es un número
real.
        errordlg('"k" eje de rotación tiene que ser real y
diferente de infinito.', 'Atención', 'modal'); %Se manda error.
    end
    set(handles.mensaje, 'String', 'En este caso se utiliza el
método de arandelas debido a que los diferenciales de Area de la
función son perpendiculares al eje de rotación. Por lo que al rotar
sobre el mismo. Se generan arandelas en el espacio. ');
    if k > 0
        %Cálculo de volumen
        set(handles.l1, 'String', ' Cálculo del volumen del
sólido de revolución (Metodo arandelas) ');
        set(handles.l2, 'String', '-----');
        -----');
        syms f(x);
        v2=int(pi*(k-y)^2,a,b);
        if v2>0
            outstr = evalc('pretty(pi*(k-f(x))^2)');
            outstrcell = regexp(outstr, '\n', 'split');
            set(handles.l3, 'String', outstrcell);
            outstr1 = evalc('pretty(pi*(k-y)^2)');
            outstrcell1 = regexp(outstr1, '\n', 'split');
            set(handles.limsup1, 'String', num2str(b));
            set(handles.limsup2, 'String', num2str(b));
            set(handles.limsup3, 'String', num2str(b));
            set(handles.limsup4, 'String', num2str(b));
            set(handles.liminf1, 'String', num2str(a));
            set(handles.liminf2, 'String', num2str(a));
            set(handles.liminf3, 'String', num2str(a));
            set(handles.liminf4, 'String', num2str(a));
            set(handles.l4, 'String', outstrcell1);
            set(handles.l5, 'String', outstrcell1);
            set(handles.l6, 'String', outstrcell1);
            v1=int(pi*(k-y)^2);
            outstr2 = evalc('pretty(v1)');
            outstrcell2 = regexp(outstr2, '\n', 'split');
            set(handles.l7, 'String', outstrcell2);
            v2=int(pi*(k-y)^2,a,b);
            outstr3 = evalc('pretty(v2)');
            outstrcell3 = regexp(outstr3, '\n', 'split');
            set(handles.l8, 'String', outstrcell3);

```

```

v2=double(v2);
set(handles.l9,'String',v2);
syms u;
g=u^3;
outstr5 = evalc('pretty(g)');
outstrcell5 = regexp(outstr5,'\n','split');
set(handles.unidades1,'String',outstrcell5);
set(handles.unidades2,'String',outstrcell5);
else
outstr = evalc('pretty(-pi*(k-f(x))^2)');
outstrcell = regexp(outstr,'\n','split');
set(handles.l3,'String',outstrcell);
outstr1 = evalc('pretty(-pi*(k-y)^2)');
outstrcell1 = regexp(outstr1,'\n','split');
set(handles.limsup1,'String',num2str(b));
set(handles.limsup2,'String',num2str(b));
set(handles.limsup3,'String',num2str(b));
set(handles.limsup4,'String',num2str(b));
set(handles.liminf1,'String',num2str(a));
set(handles.liminf2,'String',num2str(a));
set(handles.liminf3,'String',num2str(a));
set(handles.liminf4,'String',num2str(a));
set(handles.l4,'String',outstrcell1);
set(handles.l5,'String',outstrcell1);
set(handles.l6,'String',outstrcell1);
v1=-int(pi*(k-y)^2);
outstr2 = evalc('pretty(v1)');
outstrcell2 = regexp(outstr2,'\n','split');
set(handles.l7,'String',outstrcell2);
v2=-int(pi*(k-y)^2,a,b);
outstr3 = evalc('pretty(v2)');
outstrcell3 = regexp(outstr3,'\n','split');
set(handles.l8,'String',outstrcell3);
v2=double(v2);
set(handles.l9,'String',v2);
syms u;
g=u^3;
outstr5 = evalc('pretty(g)');
outstrcell5 = regexp(outstr5,'\n','split');
set(handles.unidades1,'String',outstrcell5);
set(handles.unidades2,'String',outstrcell5);
end
else
%Cálculo de volumen
set(handles.l1,'String','Cálculo del volumen del
sólido de revolución (Metodo arandelas)');
set(handles.l2,'String','-----
-----');
syms f(x);
v2=int(pi*(y-k)^2,a,b);
if v2>0
outstr = evalc('pretty(pi*(f(x)-k)^2)');
outstrcell = regexp(outstr,'\n','split');
set(handles.l3,'String',outstrcell);

```

```

    outstr1 = evalc('pretty(pi*(y-k)^2)');
    outstrcell1 = regexp(outstr1,'\n','split');
    set(handles.limsup1,'String',num2str(b));
    set(handles.limsup2,'String',num2str(b));
    set(handles.limsup3,'String',num2str(b));
    set(handles.limsup4,'String',num2str(b));
    set(handles.liminf1,'String',num2str(a));
    set(handles.liminf2,'String',num2str(a));
    set(handles.liminf3,'String',num2str(a));
    set(handles.liminf4,'String',num2str(a));
    set(handles.l14,'String',outstrcell1);
    set(handles.l15,'String',outstrcell1);
    set(handles.l16,'String',outstrcell1);
    v1=int(pi*(y-k)^2);
    outstr2 = evalc('pretty(v1)');
    outstrcell2 = regexp(outstr2,'\n','split');
    set(handles.l17,'String',outstrcell2);
    v2=int(pi*(y-k)^2,a,b);
    outstr3 = evalc('pretty(v2)');
    outstrcell3 = regexp(outstr3,'\n','split');
    set(handles.l18,'String',outstrcell3);
    v2=double(v2);
    set(handles.l19,'String',v2);
    syms u;
    g=u^3;
    outstr5 = evalc('pretty(g)');
    outstrcell5 = regexp(outstr5,'\n','split');
    set(handles.unidades1,'String',outstrcell5);
    set(handles.unidades2,'String',outstrcell5);
else
    outstr = evalc('pretty(-pi*(f(x)-k)^2)');
    outstrcell = regexp(outstr,'\n','split');
    set(handles.l13,'String',outstrcell);
    outstr1 = evalc('pretty(-pi*(y-k)^2)');
    outstrcell1 = regexp(outstr1,'\n','split');
    set(handles.limsup1,'String',num2str(b));
    set(handles.limsup2,'String',num2str(b));
    set(handles.limsup3,'String',num2str(b));
    set(handles.limsup4,'String',num2str(b));
    set(handles.liminf1,'String',num2str(a));
    set(handles.liminf2,'String',num2str(a));
    set(handles.liminf3,'String',num2str(a));
    set(handles.liminf4,'String',num2str(a));
    set(handles.l14,'String',outstrcell1);
    set(handles.l15,'String',outstrcell1);
    set(handles.l16,'String',outstrcell1);
    v1=-int(pi*(y-k)^2);
    outstr2 = evalc('pretty(v1)');
    outstrcell2 = regexp(outstr2,'\n','split');
    set(handles.l17,'String',outstrcell2);
    v2=-int(pi*(y-k)^2,a,b);
    outstr3 = evalc('pretty(v2)');
    outstrcell3 = regexp(outstr3,'\n','split');
    set(handles.l18,'String',outstrcell3);

```

```

        v2=double(v2);
        set(handles.l9,'String',v2);
        syms u;
        g=u^3;
        outstr5 = evalc('pretty(g)');
        outstrcell5 = regexp(outstr5,'\n','split');
        set(handles.unidades1,'String',outstrcell5);
        set(handles.unidades2,'String',outstrcell5);
    end
end
case 5
    if isempty(get(handles.edit4,'String')) == 1
        errordlg('Debe ingresar m','Atención','modal');
    else
        m =
str2num(get(handles.edit4,'String')); %
guardamos la pendiente de giro del solido de revolución
    end
    if isempty(get(handles.edit5,'String')) == 1
        errordlg('Debe ingresar k','Atención','modal');
    else
        k =
str2num(get(handles.edit5,'String')); %
guardamos el desplazamiento de la ecuación del eje
    end

    if ~isreal(m) || isnan(m) ||
isinf(m) %Si a no es un número
real.
        errordlg('m tiene que ser real y diferente de
infinito.','Atención','modal'); %Se manda error.
    end
    if ~isreal(k) || isnan(k) ||
isinf(k) %Si a no es un número
real.
        errordlg('el desplazamiento del eje "k" tiene que ser
real y diferente de infinito.','Atención','modal'); %Se manda
error.
    end
    set(handles.mensaje,'String','En este caso se utiliza una
modificación del método de discos que al combinarlo con
conocimientos geométricos se puede hacer que los diferenciales de
area de la función sean perpendiculares al eje de rotación. Por lo
que al rotar sobre el mismo. Se generan discos en el espacio.');
```

%Cálculo de volumen

```

        set(handles.l11,'String',' Cálculo del volumen del sólido de
revolución ');
        set(handles.l12,'String','-----
-----');
        syms x
        y_p=diff(y);
        v2 = pi/((1+(m)^2)^(3/2))* int((y-m*x-k)^2*(1+m*y_p),a,b);
        if v2>0
            syms f(x) fp(x);
            outstr = evalc('pretty(pi/((1+m^2)^(3/2))*((f(x)-m*x-
k)^2*(1+m*fp(x))))');
            outstrcell = regexp(outstr,'\n','split');
```

```

        set(handles.l13, 'String', outstrcell);
        y_p=diff(y);
        outstr1 = evalc('pretty(pi/((1+m^2)^(3/2))*((y-m*x-
k)^2*(1+m*y_p)))');
        outstrcell11 = regexp(outstr1, '\n', 'split');
        set(handles.limsup1, 'String', num2str(b));
        set(handles.limsup2, 'String', num2str(b));
        set(handles.limsup3, 'String', num2str(b));
        set(handles.limsup4, 'String', num2str(b));
        set(handles.liminf1, 'String', num2str(a));
        set(handles.liminf2, 'String', num2str(a));
        set(handles.liminf3, 'String', num2str(a));
        set(handles.liminf4, 'String', num2str(a));
        set(handles.l14, 'String', outstrcell11);
        set(handles.l15, 'String', outstrcell11);
        set(handles.l16, 'String', outstrcell11);
        syms x;
        v1 = pi/((1+(m)^2)^(3/2))*int((y-m*x-k)^2*(1+m*y_p));
        outstr2 = evalc('pretty(v1)');
        outstrcell12 = regexp(outstr2, '\n', 'split');
        set(handles.l17, 'String', outstrcell12);
        v2 = pi/((1+(m)^2)^(3/2))*int((y-m*x-
k)^2*(1+m*y_p), a, b);
        outstr3 = evalc('pretty(v2)');
        outstrcell13 = regexp(outstr3, '\n', 'split');
        set(handles.l18, 'String', outstrcell13);
        v2=double(v2);
        set(handles.l19, 'String', v2);
        syms u;
        g=u^3;
        outstr5 = evalc('pretty(g)');
        outstrcell15 = regexp(outstr5, '\n', 'split');
        set(handles.unidades1, 'String', outstrcell15);
        set(handles.unidades2, 'String', outstrcell15);
    else
        syms f(x) fp(x);
        outstr = evalc('pretty(-pi/((1+m^2)^(3/2))*((f(x)-m*x-
k)^2*(1+m*fp(x))))');
        outstrcell1 = regexp(outstr, '\n', 'split');
        set(handles.l13, 'String', outstrcell);
        y_p=diff(y);
        outstr1 = evalc('pretty(-pi/((1+m^2)^(3/2))*((y-m*x-
k)^2*(1+m*y_p)))');
        outstrcell11 = regexp(outstr1, '\n', 'split');
        set(handles.limsup1, 'String', num2str(b));
        set(handles.limsup2, 'String', num2str(b));
        set(handles.limsup3, 'String', num2str(b));
        set(handles.limsup4, 'String', num2str(b));
        set(handles.liminf1, 'String', num2str(a));
        set(handles.liminf2, 'String', num2str(a));
        set(handles.liminf3, 'String', num2str(a));
        set(handles.liminf4, 'String', num2str(a));
        set(handles.l14, 'String', outstrcell11);
        set(handles.l15, 'String', outstrcell11);
        set(handles.l16, 'String', outstrcell11);

```

```

        syms x;
        v1 = -pi/((1+(m)^2)^(3/2))*int((y-m*x-k)^2*(1+m*y_p));
        outstr2 = evalc('pretty(v1)');
        outstrcell2 = regexp(outstr2,'\n','split');
        set(handles.l7,'String',outstrcell2);
        v2 = -pi/((1+(m)^2)^(3/2))*int((y-m*x-
k)^2*(1+m*y_p),a,b);
        outstr3 = evalc('pretty(v2)');
        outstrcell3 = regexp(outstr3,'\n','split');
        set(handles.l8,'String',outstrcell3);
        v2=double(v2);
        set(handles.l9,'String',v2);
        syms u;
        g=u^3;
        outstr5 = evalc('pretty(g)');
        outstrcell5 = regexp(outstr5,'\n','split');
        set(handles.unidades1,'String',outstrcell5);
        set(handles.unidades2,'String',outstrcell5);
    end
end

function pushbutton2_Callback(hObject, eventdata, handles)
%aquí iba el código de la gráfica 2d
handles.a = str2num(get(handles.edit2,'String'));
handles.b = str2num(get(handles.edit3,'String'));
handles.x = linspace(handles.a, handles.b, 20);
t = str2sym(get(handles.edit1,'String'));
y = subs(t,handles.x);
handles.y = double(y);
switch handles.eje_rot
    case 3
        handles.k=str2num(get(handles.edit7,'String'));
    case 4
        handles.k=str2num(get(handles.edit6,'String'));
    case 5
        handles.k = str2num(get(handles.edit5,'String'));
        handles.m = str2num(get(handles.edit4,'String'));
end
guidata(hObject, handles);
grafica2d1;

function pushbutton1_Callback(hObject, eventdata, handles)
guidata(hObject, handles);
grafica3d1;

function manual_ClickedCallback(hObject, eventdata, handles)
[nombre dire]=uigetfile('*.pdf','Selecciona el
"manual"', 'manual.pdf');
winopen(nombre);

% --- Executes on button press in limpiar.

```

```
function limpiar_Callback(hObject, eventdata, handles)
set(handles.limsup1,'String','');
set(handles.limsup2,'String','');
set(handles.limsup3,'String','');
set(handles.limsup4,'String','');
set(handles.liminf1,'String','');
set(handles.liminf2,'String','');
set(handles.liminf3,'String','');
set(handles.liminf4,'String','');
set(handles.l1,'String','');
set(handles.l2,'String','');
set(handles.l3,'String','');
set(handles.l4,'String','');
set(handles.l5,'String','');
set(handles.l6,'String','');
set(handles.l7,'String','');
set(handles.l8,'String','');
set(handles.l9,'String','');
set(handles.unidades1,'String','');
set(handles.unidades2,'String','');
set(handles.edit1,'String','');
set(handles.edit2,'String','');
set(handles.edit3,'String','');
set(handles.edit4,'String','');
set(handles.edit5,'String','');
set(handles.edit6,'String','');
set(handles.edit7,'String','');
set(handles.mensaje,'String','');
clear all; %esto es para evitar que no exista memoria basura al
ejecutar el programa
```

Gráfica 2d

```
function varargout = grafica2d7(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn',  @grafica2d7_OpeningFcn, ...
                  'gui_OutputFcn',  @grafica2d7_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

function grafica2d7_OpeningFcn(hObject, eventdata, handles,
varargin)
handles.output = hObject;
MainHandles = guidata(interfaz7);
syms x;
if MainHandles.intervalo == 2
    handles.a1=MainHandles.a1;
    handles.b1=MainHandles.b1;
    handles.x1=MainHandles.x1;
    handles.a2=MainHandles.a2;
    handles.b2=MainHandles.b2;
    handles.x2=MainHandles.x2;
    handles.fun1=str2sym(get(MainHandles.edit1,'String'));
    handles.gun1=str2sym(get(MainHandles.edit8,'String'));
    handles.fun2=str2sym(get(MainHandles.edit9,'String'));
    handles.gun2=str2sym(get(MainHandles.edit12,'String'));
    handles.f1=double(subs(handles.fun1,handles.x1));
    handles.g1=double(subs(handles.gun1,handles.x1));
    handles.f2=double(subs(handles.fun2,handles.x2));
    handles.g2=double(subs(handles.gun2,handles.x2));
elseif MainHandles.intervalo == 3
    handles.a1=MainHandles.a1;
    handles.b1=MainHandles.b1;
    handles.x1=MainHandles.x1;
    handles.a2=MainHandles.a2;
    handles.b2=MainHandles.b2;
    handles.x2=MainHandles.x2;
    handles.a3=MainHandles.a3;
    handles.b3=MainHandles.b3;
    handles.x3=MainHandles.x3;
    handles.fun1=str2sym(get(MainHandles.edit1,'String'));
    handles.gun1=str2sym(get(MainHandles.edit8,'String'));
    handles.fun2=str2sym(get(MainHandles.edit9,'String'));
    handles.gun2=str2sym(get(MainHandles.edit12,'String'));
    handles.fun3=str2sym(get(MainHandles.edit13,'String'));
```

```
handles.gun3=str2sym(get(MainHandles.edit16,'String'));
handles.f1=double(subs(handles.fun1,handles.x1));
handles.g1=double(subs(handles.gun1,handles.x1));
handles.f2=double(subs(handles.fun2,handles.x2));
handles.g2=double(subs(handles.gun2,handles.x2));
handles.f3=double(subs(handles.fun3,handles.x3));
handles.g3=double(subs(handles.gun3,handles.x3));
elseif MainHandles.intervalo == 4
handles.a1=MainHandles.a1;
handles.b1=MainHandles.b1;
handles.x1=MainHandles.x1;
handles.a2=MainHandles.a2;
handles.b2=MainHandles.b2;
handles.x2=MainHandles.x2;
handles.a3=MainHandles.a3;
handles.b3=MainHandles.b3;
handles.x3=MainHandles.x3;
handles.a4=MainHandles.a4;
handles.b4=MainHandles.b4;
handles.x4=MainHandles.x4;
handles.fun1=str2sym(get(MainHandles.edit1,'String'));
handles.gun1=str2sym(get(MainHandles.edit8,'String'));
handles.fun2=str2sym(get(MainHandles.edit9,'String'));
handles.gun2=str2sym(get(MainHandles.edit12,'String'));
handles.fun3=str2sym(get(MainHandles.edit13,'String'));
handles.gun3=str2sym(get(MainHandles.edit16,'String'));
handles.fun4=str2sym(get(MainHandles.edit17,'String'));
handles.gun4=str2sym(get(MainHandles.edit20,'String'));
handles.f1=double(subs(handles.fun1,handles.x1));
handles.g1=double(subs(handles.gun1,handles.x1));
handles.f2=double(subs(handles.fun2,handles.x2));
handles.g2=double(subs(handles.gun2,handles.x2));
handles.f3=double(subs(handles.fun3,handles.x3));
handles.g3=double(subs(handles.gun3,handles.x3));
handles.f4=double(subs(handles.fun4,handles.x4));
handles.g4=double(subs(handles.gun4,handles.x4));
else
handles.a1=MainHandles.a1;
handles.b1=MainHandles.b1;
handles.x1=MainHandles.x1;
handles.a2=MainHandles.a2;
handles.b2=MainHandles.b2;
handles.x2=MainHandles.x2;
handles.a3=MainHandles.a3;
handles.b3=MainHandles.b3;
handles.x3=MainHandles.x3;
handles.a4=MainHandles.a4;
handles.b4=MainHandles.b4;
handles.x4=MainHandles.x4;
handles.a5=MainHandles.a5;
handles.b5=MainHandles.b5;
handles.x5=MainHandles.x5;
handles.fun1=str2sym(get(MainHandles.edit1,'String'));
```

```

handles.gun1=str2sym(get(MainHandles.edit8,'String'));
handles.fun2=str2sym(get(MainHandles.edit9,'String'));
handles.gun2=str2sym(get(MainHandles.edit12,'String'));
handles.fun3=str2sym(get(MainHandles.edit13,'String'));
handles.gun3=str2sym(get(MainHandles.edit16,'String'));
handles.fun4=str2sym(get(MainHandles.edit17,'String'));
handles.gun4=str2sym(get(MainHandles.edit20,'String'));
handles.fun5=str2sym(get(MainHandles.edit21,'String'));
handles.gun5=str2sym(get(MainHandles.edit24,'String'));
handles.f1=double(subs(handles.fun1,handles.x1));
handles.g1=double(subs(handles.gun1,handles.x1));
handles.f2=double(subs(handles.fun2,handles.x2));
handles.g2=double(subs(handles.gun2,handles.x2));
handles.f3=double(subs(handles.fun3,handles.x3));
handles.g3=double(subs(handles.gun3,handles.x3));
handles.f4=double(subs(handles.fun4,handles.x4));
handles.g4=double(subs(handles.gun4,handles.x4));
handles.f5=double(subs(handles.fun5,handles.x5));
handles.g5=double(subs(handles.gun5,handles.x5));
end
handles.intervalo=MainHandles.intervalo;

handles.eje_rot=MainHandles.eje_rot;
switch handles.eje_rot
    case 3
        handles.k=MainHandles.k;
    case 4
        handles.k = MainHandles.k;
end
guidata(hObject, handles);

function varargout = grafica2d7_OutputFcn(hObject, eventdata,
handles)
varargout{1} = handles.output;

function cerrar_Callback(hObject, eventdata, handles)
close(grafica2d7);

% --- Executes on button press in grafica.
function grafica_Callback(hObject, eventdata, handles)
if handles.intervalo == 2
    x1=handles.x1;
    f1=handles.f1;
    g1=handles.g1;
    x2=handles.x2;
    f2=handles.f2;
    g2=handles.g2;
    switch handles.eje_rot
        case 1
            title("Gráfico 2D"),xlabel("x"),ylabel("y");
            plot(x1,f1,x1,g1,x2,f2,x2,g2);
            hold on;

```

```

handles.gun1=str2sym(get(MainHandles.edit8,'String'));
    grid on,axis equal;
    hold off;
    case 2
        title("Gráfico 2D"),xlabel("x"),ylabel("y");
        plot(x1,f1,x1,g1,x2,f2,x2,g2);
        hold on;
        grid on,axis equal;
        hold off;
    case 3
        title("Gráfico 2D"),xlabel("x"),ylabel("y");
        k=handles.k;
        plot(x1,f1,x1,g1,x1,f2,x2,g2);
        hold on;
        grid on,axis equal;
        xline(k);
        hold off;
    case 4
        title("Gráfico 2D"),xlabel("x"),ylabel("y");
        k=handles.k;
        plot(x1,f1,x1,g1,x1,f2,x2,g2);
        hold on;
        grid on,axis equal;
        yline(k);
        hold off;
end
elseif handles.intervalo == 3
    x1=handles.x1;
    f1=handles.f1;
    g1=handles.g1;

    x2=handles.x2;
    f2=handles.f2;
    g2=handles.g2;

    x3=handles.x3;
    f3=handles.f3;
    g3=handles.g3;

    switch handles.eje_rot
        case 1
            title("Gráfico 2D"),xlabel("x"),ylabel("y");
            plot(x1,f1,x1,g1,x2,f2,x2,g2,x3,f3,x3,g3);
            hold on;
            grid on,axis equal;
            hold off;
        case 2
            title("Gráfico 2D"),xlabel("x"),ylabel("y");
            plot(x1,f1,x1,g1,x2,f2,x2,g2,x3,f3,x3,g3);
            hold on;
            grid on,axis equal;
            hold off;
        case 3
            title("Gráfico 2D"),xlabel("x"),ylabel("y");

```

```

        k=handles.k;
        plot(x1,f1,x1,g1,x2,f2,x2,g2,x3,f3,x3,g3);
        hold on;
        grid on,axis equal;
        xline(k);
        hold off;
    case 4
        title("Gráfico 2D"),xlabel("x"),ylabel("y");
        k=handles.k;
        plot(x1,f1,x1,g1,x2,f2,x2,g2,x3,f3,x3,g3);
        hold on;
        grid on,axis equal;
        yline(k);
        hold off;
    end
elseif handles.intervalo == 4
    x1=handles.x1;
    f1=handles.f1;
    g1=handles.g1;

    x2=handles.x2;
    f2=handles.f2;
    g2=handles.g2;

    x3=handles.x3;
    f3=handles.f3;
    g3=handles.g3;

    x4=handles.x4;
    f4=handles.f4;
    g4=handles.g4;
    switch handles.eje_rot
    case 1
        title("Gráfico 2D"),xlabel("x"),ylabel("y");
        plot(x1,f1,x2,f2,x3,f3,x4,f4,x1,g1,x2,g2,x3,g3,x4,g4);
        hold on;
        grid on,axis equal;
        hold off;
    case 2
        title("Gráfico 2D"),xlabel("x"),ylabel("y")
        plot(x1,f1,x2,f2,x3,f3,x4,f4,x1,g1,x2,g2,x3,g3,x4,g4);
        hold on;
        grid on,axis equal;
        hold off;
    case 3
        k=handles.k;
        title("Gráfico 2D"),xlabel("x"),ylabel("y");
        plot(x1,f1,x2,f2,x3,f3,x4,f4,x1,g1,x2,g2,x3,g3,x4,g4);
        hold on;
        xline(k);
        grid on,axis equal;
        hold off;
    case 4
        k=handles.k;

```

```

        title("Gráfico 2D"),xlabel("x"),ylabel("y");
        plot(x1,f1,x2,f2,x3,f3,x4,f4,x1,g1,x2,g2,x3,g3,x4,g4);
        hold on;
        yline(k);
        grid on,axis equal;
        hold off;
    end
else
    x1=handles.x1;
    f1=handles.f1;
    g1=handles.g1;

    x2=handles.x2;
    f2=handles.f2;
    g2=handles.g2;

    x3=handles.x3;
    f3=handles.f3;
    g3=handles.g3;

    x4=handles.x4;
    f4=handles.f4;
    g4=handles.g4;

    x5=handles.x5;
    f5=handles.f5;
    g5=handles.g5;

    switch handles.eje_rot
        case 1
            title("Gráfico 2D"),xlabel("x"),ylabel("y");
plot(x1,f1,x1,g1,x2,f2,x2,g2,x3,f3,x3,g3,x4,f4,x4,g4,x5,f5,x5,g5);
            hold on;
            grid on,axis equal;
            hold off;
        case 2
            title("Gráfico 2D"),xlabel("x"),ylabel("y");
plot(x1,f1,x1,g1,x2,f2,x2,g2,x3,f3,x3,g3,x4,f4,x4,g4,x5,f5,x5,g5);
            hold on;
            grid on,axis equal;
            hold off;
        case 3
            k=handles.k;
            title("Gráfico 2D"),xlabel("x"),ylabel("y");
plot(x1,f1,x1,g1,x2,f2,x2,g2,x3,f3,x3,g3,x4,f4,x4,g4,x5,f5,x5,g5);
            hold on;
            xline(k);
            grid on,axis equal;
            hold off;
        case 4
            k=handles.k;

```

```
        title("Gráfico 2D"),xlabel("x"),ylabel("y");  
plot(x1,f1,x1,g1,x2,f2,x2,g2,x3,f3,x3,g3,x4,f4,x4,g4,x5,f5,x5,g5);  
    hold on;  
    yline(k);  
    grid on,axis equal;  
    hold off;  
end  
end
```

Gráfica 3d.

```
function varargout = grafica3d7(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn', @grafica3d7_OpeningFcn, ...
                  'gui_OutputFcn',  @grafica3d7_OutputFcn, ...
                  'gui_LayoutFcn',  [], ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargin
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

function grafica3d7_OpeningFcn(hObject, eventdata, handles,
varargin)
handles.output = hObject;
MainHandles = guidata(interfaz7);
if MainHandles.intervalo == 2
    handles.a1=MainHandles.a1;
    handles.b1=MainHandles.b1;
    handles.x1=MainHandles.x1;
    handles.f1=MainHandles.f1;
    handles.g1=MainHandles.g1;
    handles.a2=MainHandles.a2;
    handles.b2=MainHandles.b2;
    handles.x2=MainHandles.x2;
    handles.f2=MainHandles.f2;
    handles.g2=MainHandles.g2;
elseif MainHandles.intervalo == 3
    handles.a1=MainHandles.a1;
    handles.b1=MainHandles.b1;
    handles.x1=MainHandles.x1;
    handles.f1=MainHandles.f1;
    handles.g1=MainHandles.g1;
    handles.a2=MainHandles.a2;
    handles.b2=MainHandles.b2;
    handles.x2=MainHandles.x2;
    handles.f2=MainHandles.f2;
    handles.g2=MainHandles.g2;
    handles.a3=MainHandles.a3;
    handles.b3=MainHandles.b3;
    handles.x3=MainHandles.x3;
    handles.f3=MainHandles.f3;
    handles.g3=MainHandles.g3;
elseif MainHandles.intervalo == 4
    handles.a1=MainHandles.a1;
    handles.b1=MainHandles.b1;
```

```

handles.x1=MainHandles.x1;
handles.f1=MainHandles.f1;
handles.g1=MainHandles.g1;
handles.a2=MainHandles.a2;
handles.b2=MainHandles.b2;
handles.x2=MainHandles.x2;
handles.f2=MainHandles.f2;
handles.g2=MainHandles.g2;
handles.a3=MainHandles.a3;
handles.b3=MainHandles.b3;
handles.x3=MainHandles.x3;
handles.f3=MainHandles.f3;
handles.g3=MainHandles.g3;
handles.a4=MainHandles.a4;
handles.b4=MainHandles.b4;
handles.x4=MainHandles.x4;
handles.f4=MainHandles.f4;
handles.g4=MainHandles.g4;
else
handles.a1=MainHandles.a1;
handles.b1=MainHandles.b1;
handles.x1=MainHandles.x1;
handles.f1=MainHandles.f1;
handles.g1=MainHandles.g1;
handles.a2=MainHandles.a2;
handles.b2=MainHandles.b2;
handles.x2=MainHandles.x2;
handles.f2=MainHandles.f2;
handles.g2=MainHandles.g2;
handles.a3=MainHandles.a3;
handles.b3=MainHandles.b3;
handles.x3=MainHandles.x3;
handles.f3=MainHandles.f3;
handles.g3=MainHandles.g3;
handles.a4=MainHandles.a4;
handles.b4=MainHandles.b4;
handles.x4=MainHandles.x4;
handles.f4=MainHandles.f4;
handles.g4=MainHandles.g4;
handles.a5=MainHandles.a5;
handles.b5=MainHandles.b5;
handles.x5=MainHandles.x5;
handles.f5=MainHandles.f5;
handles.g5=MainHandles.g5;
end
handles.intervalo=MainHandles.intervalo;
handles.eje_rot=MainHandles.eje_rot;
guidata(hObject, handles);

function varargout = grafica3d7_OutputFcn(hObject, eventdata,
handles)
varargout{1} = handles.output;

function cerrar_Callback(hObject, eventdata, handles)

```

```

close(grafica3d7);

% --- Executes on button press in graficar.
function graficar_Callback(hObject, eventdata, handles)
if handles.intervalo == 2
    f1=handles.f1;
    g1=handles.g1;
    f2=handles.f2;
    g2=handles.g2;
    if (handles.eje_rot == 1) || (handles.eje_rot == 3)
        %%
        subplot(2,2,1)
        [X,Y,Z] = cylinder(f1);

        i=1;
        hold on
        %
        while i <length(X)*5
            % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
            surf(X,Y,Z);
            i=i+1;
            view([20+2*i,30]);
            pause(0.01);
            title('Gráfica 3D');
            axis square;
            grid on;

        end
    subplot(2,2,2)
        [X,Y,Z] = cylinder(g1);
        j=1;
        hold on
        %
        while j <length(X)*5
            % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
            surf(X,Y,Z);
            j=j+1;
            view([20+2*j,30]);
            pause(0.01);
            title('Gráfica 3D');
            axis square;
            grid on;

        end
    subplot(2,2,3)
        [X,Y,Z] = cylinder(f2);
            k=1;
        hold on
        %
        while k <length(X)*5
            % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
            surf(X,Y,Z);
            k=k+1;

```

```

view([20+2*i,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
    subplot(2,2,4)
[X,Y,Z] = cylinder(g2);
l=1;
hold off
%
while l <length(X)*5
% surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
surf(X,Y,Z);
l=l+1;
view([20+2*l,30]);
pause(0.01);
title('Gráfica 3D');
axis square;
grid on;

end

elseif (handles.eje_rot == 2) || (handles.eje_rot == 4)
    %%
    subplot(2,2,1)
[X,Y,Z] = cylinder(g1);

    i=1;
    hold on
    %
    while i <length(X)*5
% surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
surf(X,Y,Z);
i=i+1;
view([20+2*i,30]);
pause(0.01);
title('Gráfica 3D');
axis square;
grid on;

    end
subplot(2,2,2)
[X,Y,Z] = cylinder(f1);
j=1;
hold on
%
while j <length(X)*5
% surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
surf(X,Y,Z);
j=j+1;
view([20+2*j,30]);
pause(0.01);

```

```

title('Gráfica 3D');
    axis square;
    grid on;

    end
    subplot(2,2,3)
    [X,Y,Z] = cylinder(g2);
        k=1;
    hold on
    %
    while k <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    k=k+1;
    view([20+2*i,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
    subplot(2,2,4)
    [X,Y,Z] = cylinder(f2);
    l=1;
    hold off
    %
    while l <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    l=l+1;
    view([20+2*l,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
end
elseif handles.intervalo == 3
    f1=handles.f1;
    g1=handles.g1;
    f2=handles.f2;
    g2=handles.g2;
    f3=handles.f3;
    g3=handles.g3;
    if (handles.eje_rot == 1) || (handles.eje_rot == 3)
        %%
        subplot(2,3,1)
        [X,Y,Z] = cylinder(f1);

        i=1;
        hold on
        %
        while i <length(X)*5

```

```

% surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    i=i+1;
    view([20+2*i,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
subplot(2,3,2)
[X,Y,Z] = cylinder(g1);
j=1;
hold on
%
while j <length(X)*5
% surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
surf(X,Y,Z);
j=j+1;
view([20+2*j,30]);
pause(0.01);
title('Gráfica 3D');
axis square;
grid on;

    end
subplot(2,3,3)
[X,Y,Z] = cylinder(f2);
k=1;
hold on
%
while k <length(X)*5
% surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
surf(X,Y,Z);
k=k+1;
view([20+2*i,30]);
pause(0.01);
title('Gráfica 3D');
axis square;
grid on;

    end
subplot(2,3,4)
[X,Y,Z] = cylinder(g2);
l=1;
hold on
%
while l <length(X)*5
% surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
surf(X,Y,Z);
l=l+1;
view([20+2*l,30]);
pause(0.01);
title('Gráfica 3D');

```

```

axis square;
    grid on;

    end
    subplot(2,3,5)
        [X,Y,Z] = cylinder(f3);
        k=1;
    hold on
    %
    while k <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    k=k+1;
    view([20+2*i,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
    subplot(2,3,6)
    [X,Y,Z] = cylinder(g3);
    l=1;
    hold off
    %
    while l <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    l=l+1;
    view([20+2*l,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end

elseif (handles.eje_rot == 2) || (handles.eje_rot == 4)
    subplot(2,3,1)
        [X,Y,Z] = cylinder(g1);

        i=1;
        hold on
        %
        while i <length(X)*5
        % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
        surf(X,Y,Z);
        i=i+1;
        view([20+2*i,30]);
        pause(0.01);
        title('Gráfica 3D');
        axis square;
        grid on;

```

```

end
subplot(2,3,2)
    [X,Y,Z] = cylinder(f1);
    j=1;
    hold on
    %
    while j <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    j=j+1;
    view([20+2*j,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
    subplot(2,3,3)
    [X,Y,Z] = cylinder(g2);
        k=1;
    hold on
    %
    while k <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    k=k+1;
    view([20+2*i,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
    subplot(2,3,4)
    [X,Y,Z] = cylinder(f2);
    l=1;
    hold on
    %
    while l <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    l=l+1;
    view([20+2*l,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
    subplot(2,3,5)
        [X,Y,Z] = cylinder(g3);
        k=1;
    hold on
    %

```

```

while k <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    k=k+1;
    view([20+2*i,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
    subplot(2,3,6)
    [X,Y,Z] = cylinder(f3);
    l=1;
    hold off
    %
    while l <length(X)*5
        % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
        surf(X,Y,Z);
        l=l+1;
        view([20+2*l,30]);
        pause(0.01);
        title('Gráfica 3D');
        axis square;
        grid on;

    end
end
elseif handles.intervalo == 4
    f1=handles.f1;
    g1=handles.g1;
    f2=handles.f2;
    g2=handles.g2;
    f3=handles.f3;
    g3=handles.g3;
    f4=handles.f4;
    g4=handles.g4;
    if (handles.eje_rot == 1) || (handles.eje_rot == 3)
        subplot(3,3,1)
        [X,Y,Z] = cylinder(f1);

        i=1;
        hold on
        %
        while i <length(X)*5
            % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
            surf(X,Y,Z);
            i=i+1;
            view([20+2*i,30]);
            pause(0.01);
            title('Gráfica 3D');
            axis square;
            grid on;

```

```

end
subplot(3,3,2)
[X,Y,Z] = cylinder(g1);
j=1;
hold on
%
while j <length(X)*5
% surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
surf(X,Y,Z);
j=j+1;
view([20+2*j,30]);
pause(0.01);
title('Gráfica 3D');
axis square;
grid on;

end
subplot(3,3,3)
[X,Y,Z] = cylinder(f2);
k=1;
hold on
%
while k <length(X)*5
% surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
surf(X,Y,Z);
k=k+1;
view([20+2*i,30]);
pause(0.01);
title('Gráfica 3D');
axis square;
grid on;

end
subplot(3,3,4)
[X,Y,Z] = cylinder(g2);
l=1;
hold on
%
while l <length(X)*5
% surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
surf(X,Y,Z);
l=l+1;
view([20+2*l,30]);
pause(0.01);
title('Gráfica 3D');
axis square;
grid on;

end
subplot(3,3,5)
[X,Y,Z] = cylinder(f3);
k=1;
hold on
%
while k <length(X)*5

```

```

% surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    k=k+1;
    view([20+2*i,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
    subplot(3,3,6)
    [X,Y,Z] = cylinder(g3);
    l=1;
    hold off
    %
    while l <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    l=l+1;
    view([20+2*l,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
    subplot(3,3,7)
    [X,Y,Z] = cylinder(f4);
    k=1;
    hold on
    %
    while k <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    k=k+1;
    view([20+2*i,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
    subplot(3,3,8)
    [X,Y,Z] = cylinder(g4);
    l=1;
    hold off
    %
    while l <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    l=l+1;
    view([20+2*l,30]);
    pause(0.01);
    title('Gráfica 3D');

```

```

axis square;
    grid on;

    end

elseif (handles.eje_rot == 2) || (handles.eje_rot == 4)
    subplot(3,3,1)
    [X,Y,Z] = cylinder(g1);

    i=1;
    hold on
    %
    while i <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    i=i+1;
    view([20+2*i,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
subplot(3,3,2)
[X,Y,Z] = cylinder(f1);
j=1;
hold on
%
while j <length(X)*5
% surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
surf(X,Y,Z);
j=j+1;
view([20+2*j,30]);
pause(0.01);
title('Gráfica 3D');
axis square;
grid on;

end
subplot(3,3,3)
[X,Y,Z] = cylinder(g2);
k=1;
hold on
%
while k <length(X)*5
% surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
surf(X,Y,Z);
k=k+1;
view([20+2*i,30]);
pause(0.01);
title('Gráfica 3D');
axis square;

```

```

grid on;

    end
    subplot(3,3,4)
    [X,Y,Z] = cylinder(f2);
    l=1;
    hold on
    %
    while l <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    l=l+1;
    view([20+2*l,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
    subplot(3,3,5)
        [X,Y,Z] = cylinder(g3);
        k=1;
    hold on
    %
    while k <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    k=k+1;
    view([20+2*i,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
    subplot(3,3,6)
    [X,Y,Z] = cylinder(f3);
    l=1;
    hold off
    %
    while l <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    l=l+1;
    view([20+2*l,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
    subplot(3,3,7)
        [X,Y,Z] = cylinder(g4);
        k=1;

```

```

hold on
    %
    while k <length(X)*5
        % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
        surf(X,Y,Z);
        k=k+1;
        view([20+2*i,30]);
        pause(0.01);
        title('Gráfica 3D');
        axis square;
        grid on;

        end
        subplot(3,3,8)
        [X,Y,Z] = cylinder(f4);
        l=1;
        hold off
        %
        while l <length(X)*5
            % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
            surf(X,Y,Z);
            l=l+1;
            view([20+2*l,30]);
            pause(0.01);
            title('Gráfica 3D');
            axis square;
            grid on;

            end
        end
else handles.intervalo == 5
    f1=handles.f1;
    g1=handles.g1;
    f2=handles.f2;
    g2=handles.g2;
    f3=handles.f3;
    g3=handles.g3;
    f4=handles.f4;
    g4=handles.g4;
    f5=handles.f5;
    g5=handles.g5;
    if (handles.eje_rot == 1) || (handles.eje_rot == 3)
        subplot(3,4,1)
        [X,Y,Z] = cylinder(f1);

        i=1;
        hold on
        %
        while i <length(X)*5
            % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
            surf(X,Y,Z);
            i=i+1;
            view([20+2*i,30]);
            pause(0.01);

```

```

title('Gráfica 3D');
    axis square;
    grid on;

    end
subplot(3,4,2)
    [X,Y,Z] = cylinder(g1);
    j=1;
    hold on
    %
    while j <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    j=j+1;
    view([20+2*j,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
    subplot(3,4,3)
    [X,Y,Z] = cylinder(f2);
        k=1;
    hold on
    %
    while k <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    k=k+1;
    view([20+2*i,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
    subplot(3,4,4)
    [X,Y,Z] = cylinder(g2);
    l=1;
    hold on
    %
    while l <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    l=l+1;
    view([20+2*l,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
    subplot(3,4,5)

```

```

[X,Y,Z] = cylinder(f3);
        k=1;
    hold on
    %
    while k <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    k=k+1;
    view([20+2*i,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
    subplot(3,4,6)
    [X,Y,Z] = cylinder(g3);
    l=1;
    hold off
    %
    while l <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    l=l+1;
    view([20+2*l,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
    subplot(3,4,7)
        [X,Y,Z] = cylinder(f4);
        k=1;
    hold on
    %
    while k <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    k=k+1;
    view([20+2*i,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
    subplot(3,4,8)
    [X,Y,Z] = cylinder(g4);
    l=1;
    hold on
    %
    while l <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));

```

```

surf(X,Y,Z);
    l=l+1;
    view([20+2*l,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

end
subplot(3,4,9)
                                [X,Y,Z] = cylinder(f5);
                                k=1;
    hold on
    %
    while k <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    k=k+1;
    view([20+2*i,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

end
subplot(3,4,10)
[X,Y,Z] = cylinder(g5);
l=1;
hold off
%
while l <length(X)*5
% surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
surf(X,Y,Z);
l=l+1;
view([20+2*l,30]);
pause(0.01);
title('Gráfica 3D');
axis square;
grid on;

end

elseif (handles.eje_rot == 2) || (handles.eje_rot == 4)
    subplot(3,4,1)
    [X,Y,Z] = cylinder(g1);

    i=1;
    hold on
    %
    while i <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    i=i+1;
    view([20+2*i,30]);

```

```

pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
subplot(3,4,2)
    [X,Y,Z] = cylinder(f1);
    j=1;
    hold on
    %
    while j <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    j=j+1;
    view([20+2*j,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
    subplot(3,4,3)
    [X,Y,Z] = cylinder(g2);
        k=1;
    hold on
    %
    while k <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    k=k+1;
    view([20+2*i,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
    subplot(3,4,4)
    [X,Y,Z] = cylinder(f2);
    l=1;
    hold on
    %
    while l <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    l=l+1;
    view([20+2*l,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
end

```

```

subplot(3,4,5)
    [X,Y,Z] = cylinder(g3);
    k=1;
    hold on
    %
    while k <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    k=k+1;
    view([20+2*i,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

    end
subplot(3,4,6)
[X,Y,Z] = cylinder(f3);
l=1;
hold off
%
while l <length(X)*5
% surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
surf(X,Y,Z);
l=l+1;
view([20+2*l,30]);
pause(0.01);
title('Gráfica 3D');
axis square;
grid on;

end
subplot(3,4,7)
[X,Y,Z] = cylinder(g4);
k=1;
hold on
%
while k <length(X)*5
% surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
surf(X,Y,Z);
k=k+1;
view([20+2*i,30]);
pause(0.01);
title('Gráfica 3D');
axis square;
grid on;

end
subplot(3,4,8)
[X,Y,Z] = cylinder(f4);
l=1;
hold on
%
while l <length(X)*5

```

```

% surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    l=l+1;
    view([20+2*l,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

end
subplot(3,4,9)
                                [X,Y,Z] = cylinder(g5);
                                k=1;
    hold on
    %
    while k <length(X)*5
    % surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
    surf(X,Y,Z);
    k=k+1;
    view([20+2*i,30]);
    pause(0.01);
    title('Gráfica 3D');
    axis square;
    grid on;

end
subplot(3,4,10)
[X,Y,Z] = cylinder(f5);
l=1;
hold off
%
while l <length(X)*5
% surf(X(1:i+1,1:i+1),Y(1:i+1,1:i+1),Z(1:i+1,1:i+1));
surf(X,Y,Z);
l=l+1;
view([20+2*l,30]);
pause(0.01);
title('Gráfica 3D');
axis square;
grid on;

end
end
end

```



epoch

Dirección de Bibliotecas y
Recursos del Aprendizaje

UNIDAD DE PROCESOS TÉCNICOS Y ANÁLISIS BIBLIOGRÁFICO Y
DOCUMENTAL

REVISIÓN DE NORMAS TÉCNICAS, RESUMEN Y BIBLIOGRAFÍA

Fecha de entrega: 19 / 07 / 2023

INFORMACIÓN DEL AUTOR/A (S)
Nombres – Apellidos: <i>Jessica Paola Bonilla Escudero</i>
INFORMACIÓN INSTITUCIONAL
<i>Instituto de Posgrado y Educación Continua</i>
Título a optar: <i>Magíster en Matemática mención Modelación y Docencia</i>
f. Analista de Biblioteca responsable: Lic. Luis Caminos Vargas Mgs.



Firmado electrónicamente por:
LUIS ALBERTO
CAMINOS
VARGAS



0075-DBRA-UTP-IPEC-2023