

UNIVERSIDAD NACIONAL DE CAJAMARCA

FACULTAD DE INGENIERÍA

ESCUELA ACADÉMICO PROFESIONAL DE INGENIERÍA CIVIL



**“COMPARACIÓN DE LA RESPUESTA ESTRUCTURAL DE
ESTRUCTURAS BIDIMENSIONALES USANDO EL SOFTWARE EDUCATIVO
PAME UNC 2D Y SAP2000”**

TESIS PARA OPTAR EL TÍTULO DE:

INGENIERO CIVIL

AUTOR:

BACH. JARA BARDALES, EDWIN HENRY

ASESOR:

DR. ING. MOSQUEIRA MORENO, MIGUEL ANGEL

CAJAMARCA – PERÚ

2025

CONSTANCIA DE INFORME DE ORIGINALIDAD

- FACULTAD DE INGENIERÍA -

1. **Investigador:** EDWIN HENRY JARA BARDALES
DNI: 72726796
Escuela Profesional: INGENIERÍA CIVIL
2. **Asesor:** Dr. Ing. MIGUEL ANGEL MOSQUEIRA MORENO
Facultad: DE INGENIERÍA
3. **Grado académico o título profesional**
☐ Bachiller ☒ Título profesional ☐ Segunda especialidad
☐ Maestro ☐ Doctor
4. **Tipo de Investigación:**
☒ Tesis ☐ Trabajo de investigación ☐ Trabajo de suficiencia profesional
☐ Trabajo académico
5. **Título de Trabajo de Investigación:**
COMPARACIÓN DE LA RESPUESTA ESTRUCTURAL DE ESTRUCTURAS BIDIMENSIONALES
USANDO EL SOFTWARE EDUCATIVO PAME UNC 2D Y SAP2000
6. **Fecha de evaluación:** 05/07/2025
7. **Software antiplagio:** ☒ TURNITIN ☐ URKUND (OURIGINAL) (*)
8. **Porcentaje de Informe de Similitud:** 10%
9. **Código Documento: Oid:** 3117:471878638
10. **Resultado de la Evaluación de Similitud:**
☒ APROBADO ☐ PARA LEVANTAMIENTO DE OBSERVACIONES O DESAPROBADO

Fecha Emisión: 06/07/2025

	 Firmado digitalmente por: BAZAN DIAZ Laura Sofia FAU 20148258601 soft Motivo: En señal de conformidad Fecha: 06/07/2025 15:25:09-0500
FIRMA DEL ASESOR Dr. Ing. MIGUEL ANGEL MOSQUEIRA MORENO DNI: 26733060	UNIDAD DE INVESTIGACIÓN FI



Universidad Nacional de Cajamarca

"Norte de la Universidad Peruana"

Fundada por Ley 14015 del 13 de Febrero de 1962

FACULTAD DE INGENIERÍA

Teléf. N° 365976 Anexo N° 1129-1130



ACTA DE SUSTENTACIÓN PÚBLICA DE TESIS.

TITULO : COMPARACIÓN DE LA RESPUESTA ESTRUCTURAL DE ESTRUCTURAS BIDIMENSIONALES USANDO EL SOFTWARE EDUCATIVO PAME UNC 2D Y SAP 2000"

ASESOR : Dr. Ing. Miguel Angel Mosqueira Moreno.

En la ciudad de Cajamarca, dando cumplimiento a lo dispuesto por el Oficio Múltiple N° 0391-2025-PUB-SA-FI-UNC, de fecha 17 de julio de 2025, de la Secretaría Académica de la Facultad de Ingeniería, a los **veintidós días del mes de julio de 2025**, siendo las ocho horas (08:00 a.m.) en la Sala de Audiovisuales (Edificio 1A – Segundo Piso), de la Facultad de Ingeniería se reunieron los Señores Miembros del Jurado Evaluador:

Presidente : Ing. Marcos Mendoza Linares.
Vocal : Dr. Ing. Mauro Augusto Centurión Vargas.
Secretario : M.Cs. Ing. Manuel Lincoln Minchán Pajares.

Para proceder a escuchar y evaluar la sustentación pública de la tesis titulada **COMPARACIÓN DE LA RESPUESTA ESTRUCTURAL DE ESTRUCTURAS BIDIMENSIONALES USANDO EL SOFTWARE EDUCATIVO PAME UNC 2D Y SAP 2000**, presentado por el Bachiller en Ingeniería Civil **EDWIN HENRY JARA BARDALES**, asesorado por el Dr. Ing. Miguel Angel Mosqueira Moreno, para la obtención del Título Profesional

Los Señores Miembros del Jurado replicaron al sustentante debatieron entre sí en forma libre y reservada y lo evaluaron de la siguiente manera:

EVALUACIÓN PRIVADA :07..... PTS.
EVALUACIÓN PÚBLICA :11..... PTS.
EVALUACIÓN FINAL :18..... PTS Diecerocho (En letras)

En consecuencia, se lo declara APROBADO con el calificativo de 18 Diecerocho acto seguido, el presidente del jurado hizo saber el resultado de la sustentación, levantándose la presente a las 9:30 a.m. horas del mismo día, con lo cual se dio por terminado el acto, para constancia se firmó por quintuplicado.

Ing. Marcos Mendoza Linares.
Presidente

Dr. Ing. Mauro Augusto Centurión Vargas.
Vocal

M.Cs. Ing. Manuel Lincoln Minchán Pajares.
Secretario

Dr. Ing. Miguel Angel Mosqueira Moreno.
Asesor

COPYRIGHT © 2025 by
EDWIN HENRY JARA BARDALES
Todos los derechos reservados

AGRADECIMIENTO

A Dios, fuente de vida y sabiduría, su luz divina me ha encaminado en los momentos más difíciles. Al ser supremo le agradezco por cada oportunidad, cada desafío superado y cada lección aprendida. Sin su amor y misericordia, este trabajo no hubiera sido posible.

A Agustina, mi madre, por su infinito amor, a ella le debo lo que soy y también lo que pretendo ser. De ella son mis memorias de mayor alegría y profunda nostalgia.

A Francisco, mi padre, por inculcarme el valor del trabajo.

A Esther, Arturo y Enrique, mis hermanos, de quienes siempre aprendo: Esther representa, para mí, la lucha por la unidad familiar; Arturo, mi compañero en la etapa universitaria, es mi modelo de probidad; Enrique es la revolución, el mensaje encarnado de que nunca es tarde y que los objetivos trazados se pueden concretar si te muestras decidido.

A mis abuelitas: Juanita, cuya ternura es desbordante, y Mercedes, por sus consejos.

A Liset, por su persistencia y por mantener con claridad la idea de que esta investigación podía y debía concretarse.

A los señores María y Augusto, por su paciencia, apoyo y consideración.

A mis tíos, ellos me han apoyado desde que tengo uso de razón. Permítaseme, al menos, mencionarlos: Isidro, Nicolasa, Víctor, José Pedro, Santos, Segundo, Cristhian, Luis, Rufina[†], Pedro Pascual, José[†], María, Teófilo y, finalmente, a mi querida tía Isidora, pues, con sus méritos, ella cultivó en mí el ánimo de superación a través del estudio.

Al señor J.H.O.R.[†], en tan corto tiempo ha sido un maestro de vida ejemplar.

A los señores Víctor Silva[†] y Amelia Justiniano, por su apoyo en mi etapa universitaria.

A los docentes de la Facultad de Ingeniería de la UNC, por compartir su conocimiento, sugerencias y ser una guía en este proceso de aprendizaje.

Al Dr. Ing. Miguel Mosqueira, por su asesoría, eficaz apoyo y don de enseñanza.

Finalmente, a mis hermanos espirituales Johan Edquén Pérez y Edwin Pajares Torres, pues me han demostrado uno de los valores más caros, la lealtad.

DEDICATORIA

A Agustina, mi amada madre, a ella he de dedicarle todo en esta vida, pues se lo merece.

A Johann Sebastián[†], mi hijo, y a toda mi familia que está en el cielo que Vallejo y Arguedas imaginaron, allí donde su existencia ha trascendido en poesía eterna, susurrando cantos de esperanza entre los valles andinos a los que aún quedamos.

A “Chato”[†], mi fiel amigo de cuatro patas. ¡Sé libre siempre!

A todo aquel que tenga sed de conocimiento y tome a bien estudiar los aportes de esta investigación, para provecho colectivo.

— Edwin Jara.

CONTENIDO

AGRADECIMIENTO.....	iii
DEDICATORIA.....	iv
CONTENIDO.....	v
ÍNDICE DE TABLAS	vii
ÍNDICE DE FIGURAS	ix
RESUMEN	1
ABSTRACT	2
CAPÍTULO I: INTRODUCCIÓN	3
1.1. PLANTEAMIENTO DEL PROBLEMA.....	3
1.2. FORMULACIÓN DEL PROBLEMA	4
1.3. HIPÓTESIS	4
1.4. JUSTIFICACIÓN DE LA INVESTIGACIÓN	5
1.5. ALCANCES Y DELIMITACIÓN DE LA INVESTIGACIÓN	6
1.6. LIMITACIONES DE LA INVESTIGACIÓN	8
1.7. OBJETIVOS.....	9
1.8. DESCRIPCIÓN DEL CONTENIDOS.....	10
CAPÍTULO II. MARCO TEÓRICO	12
2.1. ANTECEDENTES TEÓRICOS	12
2.1.1. ANTECEDENTES INTERNACIONALES	12
2.1.2. ANTECEDENTES NACIONALES.....	12
2.1.3. ANTECEDENTES LOCALES	13
2.2. BASES TEÓRICAS	14
2.3. DEFINICIÓN BÁSICA DE TÉRMINOS.....	37
CAPÍTULO III. MATERIALES Y MÉTODOS.....	39
3.1. UBICACIÓN GEOGRÁFICA Y TEMPORAL.....	39

3.2. MATERIALES Y SOFTWARE	41
3.2.1. MATERIALES	41
3.2.2. SOFTWARE	41
3.3. METODOLOGÍA.....	41
3.4. POBLACIÓN DE ESTUDIO	42
3.5. MUESTRA	43
3.6. VARIABLES DE ESTUDIO	43
3.6.1. VARIABLE INDEPENDIENTE.....	43
3.6.2. VARIABLE DEPENDIENTE	43
3.7. PROCEDIMIENTO	44
3.7.1. DESARROLLO DE CASOS EN PAME UNC 2D	45
3.7.2. DESARROLLO DE CASOS EN SAP2000 V23.3.1	70
3.8. TÉCNICAS E INSTRUMENTOS DE RECOLECCIÓN DE INFORMACIÓN ...	73
CAPÍTULO IV. ANÁLISIS Y DISCUSIÓN DE RESULTADOS	74
4.1. ANÁLISIS DE RESULTADOS.....	74
4.2. DISCUSIÓN DE RESULTADOS	78
4.3. CORROBORACIÓN DE HIPÓTESIS	78
CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES	79
5.1. CONCLUSIONES.....	79
5.2. RECOMENDACIONES	80
REFERENCIAS BIBLIOGRÁFICAS	81
ANEXOS	83

ÍNDICE DE TABLAS

Tabla 1: Información general de inicio de ejercicio analizado	45
Tabla 2: Ingreso de coordenadas nodales.....	46
Tabla 3: Ingreso de geometría de las barras	46
Tabla 4: Secciones típicas usadas en el modelo de cercha.....	46
Tabla 5: Secciones asignadas a cada barra.....	47
Tabla 6: Restricciones en nodos para caso cercha	49
Tabla 7: ID de ejes locales para cada nodo	49
Tabla 8: Matrices de rigidez para cada barra de la cercha.	50
Tabla 9: Matriz de rigidez global ensamblada para los GDLL	50
Tabla 10: Desplazamientos nodales de armadura en PAME UNC 2D.....	51
Tabla 11: Resultados de desplazamientos nodales por eje	51
Tabla 12: Resultados de fuerzas axiales para cada barra.....	51
Tabla 13: Resultados de reacciones en los apoyos	51
Tabla 14: Ingreso de coordenadas nodales	53
Tabla 15: Ingreso de geometría de las barras.....	53
Tabla 16: Propiedades de material para cada barra	53
Tabla 17: Matrices de rigidez locales y globales	54
Tabla 18: Matriz de rigidez ensamblada, caso de viga	55
Tabla 19: Matriz de rigidez para los grados de libertad libres.....	56
Tabla 20: Inversa de [KLL]	56
Tabla 21: Desplazamientos nodales en los GDLL de la viga	57
Tabla 22: Matriz [KRL]	57
Tabla 23: Reacciones en los GDLR de la viga	58
Tabla 24: Propiedades de secciones.....	59

Tabla 25: Propiedades de las barras que conforman el pórtico	61
Tabla 26: Transpuestas de matrices de transformación para pórtico	64
Tabla 27: Inversa de la sub matriz de rigidez [KLL]	67
Tabla 28: Desplazamientos $\{U_L\}$ del pórtico en los GDLL.	68
Tabla 29: Sub matriz [KRL]	69
Tabla 30: Reacciones $\{FR\}$ en los GDLR.	69
Tabla 31: Desplazamientos nodales de la armadura en SAP2000 V23.3.1	70
Tabla 32: Reacciones en los apoyos de la armadura según SAP2000 V23.3.1	70
Tabla 33: Desplazamientos nodales de viga en SAP2000 V23.3.1	71
Tabla 34: Reacciones en los apoyos de la viga según SAP2000 V23.3.1	71
Tabla 35: Desplazamientos nodales del pórtico en SAP2000 V23.3.1.....	72
Tabla 36: Reacciones en los apoyos del pórtico según SAP2000 V23.3.1.....	72
Tabla 37: Comparación de desplazamientos de armadura.....	74
Tabla 38: Comparación de reacciones en eje “X”, para armadura.	74
Tabla 39: Comparación de reacciones en el eje “Y”, para la armadura.	74
Tabla 40: Comparación de los desplazamientos verticales de la viga.	75
Tabla 41: Comparación de los giros nodales de la viga.....	75
Tabla 42: Comparación de las reacciones en viga.	75
Tabla 43: Comparación de los desplazamientos horizontales en el pórtico.	76
Tabla 44: Comparación de los desplazamientos verticales en el pórtico.....	76
Tabla 45: Comparación de las rotaciones nodales en el pórtico.	76
Tabla 46: Comparación de las reacciones en el eje “X” para el pórtico.....	76
Tabla 47: Comparación de las reacciones en el eje “Y” para el pórtico.....	77
Tabla 48: Comparación de los momentos alrededor de "Z" para el pórtico.	77

ÍNDICE DE FIGURAS

Figura 1: Relación gráfica de fuerzas y desplazamientos uniaxiales de barra	18
Figura 2: Relación gráfica entre desplazamientos y fuerzas en viga	21
Figura 3: Deformada de la viga, por imposición de $U5$	21
Figura 4: Diagrama de cortantes $V(x)$	27
Figura 5: Diagrama de momentos $M(x)$	27
Figura 6: Convención de signos para cortantes y momentos flectores en viga.....	27
Figura 7: Solicitaciones de cortantes y momentos en nodos extremos de viga.	28
Figura 8: Efectos de desplazamientos superpuestos para análisis de pórticos	30
Figura 9: Cleve Moler, desarrollador principal de MATLAB.	34
Figura 10: Modelo de interfaz creada en App Designer	35
Figura 11: Ejemplo de análisis de estructuras de aluminio en SAP2000 V23.3.1	36
Figura 12: Cajamarca – nivel de departamento, contextualizado en el mapa de Perú.	39
Figura 13: Provincia de Cajamarca, en mapa del departamento de Cajamarca	40
Figura 14: Distrito de Cajamarca, dentro de la provincia del mismo nombre.	40
Figura 15: Ejercicio 3.1 de “Matrix Structural Analysis”	45
Figura 16: Visualización de la armadura en el lienzo principal.....	47
Figura 17: Asignación y validación interactiva de apoyos	48
Figura 18: Asignación de carga en la estructura sobre el lienzo principal.....	48
Figura 19: grados de libertad de la viga	53
Figura 20: CASO3. Ejercicio para pórticos	59
Figura 21: Geometría y carga nodal aplicada en la armadura, en SAP2000 V23.3.1.....	70
Figura 22: Geometría y cargas asignadas a viga en SAP2000 V23.3.1.	71
Figura 23: Geometría y carga del pórtico en SAP2000 V23.3.1.....	72
Figura 24: Diagrama del elemento barra con su función de desplazamiento $u(x)$	83
Figura 25: Fuerzas y desplazamientos nodales	88

Figura 26: Diagrama de flujo del ciclo “for”	90
Figura 27: Diagrama de flujo del ciclo “While”	92
Figura 28: Flujograma del multiplicador de matrices en Flowchart de MATLAB.	94
Figura 29: Diagrama de flujo del ciclo IF	97
Figura 30: Ciclo IF anidado.	101

RESUMEN

La presente investigación está encauzada a comparar de la respuesta estructural de estructuras bidimensionales usando los programas PAME UNC 2D y SAP2000 v23.3.1. Este estudio nace a raíz de un problema en la práctica profesional: la necesidad de contrastar los resultados obtenidos en programas comerciales, de forma independiente a dicho entorno, para garantizar la validez de los cálculos, ya sea en la elaboración de memorias de cálculo o la verificación simplificada de elementos estructurales en campo. PAME UNC 2D usa el método matricial de rigidez (MMR), mientras que SAP2000 v23.3.1 emplea algoritmos asociados al método de elemento finito (MEF). Aun así, SAP2000 v23.3.1 puede adaptarse a los algoritmos de su predecesor: el método matricial de rigidez. La selección de casos de estudio, armadura, viga y pórtico, se realizó mediante muestreo no probabilístico intencional, utilizando ejercicios extraídos de "Matrix Analysis of Structures" (Kassimali, A., 2022), "Matrix Structural Analysis" (McGuire, W. et al., 2015), y un caso adaptado por el tesista. Los casos fueron analizados bajo condiciones equivalentes de geometría, propiedades mecánicas, apoyos y cargas. Se trabaja en el reango elástico lineal estático. Como métrica de comparación se utilizó el Error Porcentual Absoluto Medio (MAPE). El estudio concluye que la variación porcentual promedio de la respuesta estructural es 0.00% en desplazamientos nodales, reacciones y momentos (caso de viga y pórtico), entre PAME UNC 2D y SAP2000 v23.3.1.

Palabras clave: comparación, respuesta estructural, PAME UNC 2D, SAP2000 v23.3.1.

ABSTRACT

This research aims to compare the structural response of two-dimensional structures using the programs PAME UNC 2D and SAP2000 v23.3.1. The study originates from a problem observed in professional practice: the need to contrast the results obtained with commercial software independently of such environment, in order to ensure the validity of structural calculations, whether for preparing calculations reports or for the simplified verification of structural elements on site. PAME UNC 2D applies the matrix stiffness method (MSM), whereas SAP2000 v23.3.1 uses algorithms associated with the finite element method (FEM), although it can also adapt to the algorithms of its predecessor, the matrix stiffness method. The selection of case studies, including a truss, a beam, and a frame, was performed using intentional non-probabilistic sampling, with examples extracted from Matrix Analysis of Structures (Kassimali, 2022), Matrix Structural Analysis (McGuire et al., 2015), and one case adapted by the author. The cases were analyzed under equivalent conditions of geometry, mechanical properties, supports, and loads. The analysis was carried out within the linear elastic static range. Mean Absolute Percentage Error or MAPE was used as the comparison metric. The study concludes that the mean percentage variation of the structural response is 0.00% in nodal displacements, reactions, and bending moments (for the beam and frame cases) between PAME UNC 2D and SAP2000 v23.3.1.

Keywords: comparison, structural response, PAME UNC 2D, SAP2000 v23.3.1.

CAPÍTULO I: INTRODUCCIÓN

1.1. PLANTEAMIENTO DEL PROBLEMA

A nivel global, el análisis estructural constituye una actividad esencial en la ingeniería civil, dado que permite estimar la respuesta estructural ante diferentes tipos de cargas. El uso de programas de cálculo especializados se ha generalizado como un recurso indispensable para este propósito. Entre los programas más empleados a nivel internacional se encuentran los que incorporan métodos matriciales y los basados en el método de elemento finito. (Kassimali, A., 2022, p. 2).

En el ejercicio profesional, la disponibilidad de resultados confiables es una condición crítica en la toma de cruciales decisiones y la preparación de memorias de cálculo para edificaciones. Sin embargo, debido a la diversidad de programas existentes y a la variedad de algoritmos que estos implementan, puede existir incertidumbre respecto a la magnitud de las discrepancias que se generan en los resultados, aun cuando se modelan casos bajo condiciones equivalentes de geometría, apoyos, propiedades mecánicas y cargas.

La comparación de la respuesta estructural generada por distintos programas cobra especial relevancia, ya que permite determinar el grado de concordancia de parámetros como desplazamientos nodales, reacciones en los apoyos y momentos flectores, según el tipo de estructura. Esta evaluación comparativa resulta necesaria para identificar posibles discrepancias que, de no ser detectadas, podrían trasladarse a malas decisiones de diseño o verificación estructural, afectando la confiabilidad de los cálculos.

En la práctica profesional, surge la necesidad de contrastar de manera sistemática los resultados obtenidos mediante herramientas de uso comercial ampliamente consolidadas, como SAP2000 v23.3.1, frente a otros programas libres de licencias, con el propósito de

cuantificar su variación en la respuesta estructural y asegurar que los valores calculados sean consistentes y válidos.

Por lo expuesto, la presente investigación se plantea como un estudio comparativo de la respuesta estructural de casos representativos de estructuras bidimensionales, con el objetivo de medir la variación porcentual promedio en los resultados obtenidos en ambos entornos de análisis. Este conocimiento permitirá aportar criterios técnicos que contribuyan a fortalecer la certeza en la validez de los cálculos estructurales, tanto en la elaboración de informes de diseño como en la verificación simplificada de elementos estructurales en campo.

1.2. FORMULACIÓN DEL PROBLEMA

¿En cuánto varían los resultados conseguidos en la comparación de la respuesta estructural de modelos bidimensionales utilizando el software educativo PAME UNC 2D y los generados por SAP2000 v23.3.1, en términos de desplazamientos, reacciones y momentos flectores, según el tipo de estructura?

1.3. HIPÓTESIS

La variación porcentual promedio en la comparación de la respuesta estructural obtenida mediante el software educativo PAME UNC 2D es inferior al 5% respecto a la respuesta generada por el software SAP2000 v23.3.1, para estructuras bidimensionales como armaduras, vigas y pórticos.

1.4. JUSTIFICACIÓN DE LA INVESTIGACIÓN

La presente investigación se justifica desde una perspectiva comparativa, ya que busca contrastar de forma sistemática y rigurosa la respuesta estructural obtenida mediante un software educativo denominado PAME UNC 2D, frente a un software comercial ampliamente utilizado (SAP2000 v23.3.1). Esta comparación se realiza sobre estructuras bidimensionales típicas en la práctica simplificada de la ingeniería civil: armaduras, vigas y pórticos.

El acceso a software comercial puede estar restringido por licencias, limitaciones funcionales o incluso empleo de versiones sin respaldo de la compañía desarrolladora. En este contexto surge la necesidad de comparar la respuesta estructural de dichos programas comerciales como el caso de SAP2000 v23.3.1, con otro programa independiente y libre de estas restricciones, en este caso se empleó PAME UNC 2D, cuyo valor no radica únicamente en su existencia o accesibilidad, sino en su capacidad de producir resultados consistentes.

Por tanto, el principal aporte de esta investigación no es validar el software por sí mismo, sino demostrar que la respuesta estructural obtenida con PAME UNC 2D es equivalente a la generada por SAP2000 v23.3.1, bajo condiciones de modelamiento estructural coherentes. Este enfoque comparativo responde a una lógica de investigación tecnológica aplicada.

Finalmente, la justificación del valor metodológico de esta investigación reside en su replicabilidad. Cualquier investigador con conocimientos en análisis estructural y programación puede aplicar el mismo procedimiento de comparación estructural en otros casos, fortaleciendo así la cultura de verificación numérica en los cálculos estructurales.

1.5. ALCANCES Y DELIMITACIÓN DE LA INVESTIGACIÓN

Alcances

La presente investigación se sitúa dentro del campo del análisis estructural computacional, específicamente en el estudio y comparación de la respuesta estructural en dos dimensiones de tres tipos de estructuras: armaduras planas, vigas y pórticos. El objetivo principal es evaluar, desde un enfoque cuantitativo, la concordancia de resultados entre dos entornos de software: el programa educativo PAME UNC 2D, desarrollado por el autor, y el software comercial SAP2000 v23.3.1, reconocido ampliamente en el ámbito profesional.

El estudio se enmarca en una investigación de tipo aplicada y tecnológica, con un alcance comparativo. Se centra en el análisis del comportamiento estructural bajo cargas estáticas lineales, evaluando desplazamientos nodales, reacciones en los apoyos y momentos flectores, según el tipo de estructura. Para cuantificar el grado de concordancia entre los resultados se aplica como medida principal el Error Porcentual Absoluto Medio (MAPE).

Asimismo, el alcance abarca el desarrollo y documentación de una herramienta informática de libre acceso que permita visualizar el proceso completo de análisis estructural matricial, desde la entrada de datos hasta la presentación de resultados, incluyendo elementos de apoyo visual como lienzos de dibujo, tablas y matrices numéricas. No obstante, el desarrollo informático no constituye el objetivo final, sino el medio para efectuar la comparación estructural bajo parámetros controlados y reproducibles.

Delimitación

Temporalmente, el estudio se desarrolló entre abril y diciembre del año 2024.

En cuanto a su delimitación temática, el presente estudio se encuentra delimitado al análisis estructural en el plano bidimensional (2D), considerando específicamente estructuras de barras sometidas a cargas estáticas. Se abordan tres tipos estructurales clásicos

y representativos: una armadura plana, una viga y un pórtico plano, los cuales han sido seleccionados intencionalmente por representar casos típicos del análisis estructural.

Los modelos analizados cumplen las siguientes condiciones comunes:

- Todas las estructuras se modelan en el plano XY para el caso de PAME UNC 2D, considerando grados de libertad de traslación en X e Y además, en el caso de vigas y pórticos, se considera rotación en torno al eje perpendicular al plano, en tanto que en SAP2000 v23.3.1, el modelado es realizado en el plano XZ.
- El análisis se realiza en el rango elástico lineal, con propiedades geométricas y mecánicas constantes (longitudes, secciones transversales y módulo elástico); el usuario de PAME UNC 2D debe verificar la consistencia de las unidades.
- Las cargas aplicadas son estáticas y concentradas, ubicadas en nodos o coordenadas específicas, según corresponda al tipo de estructura.
- Para cada modelo, se emplea la misma configuración de nodos, barras, apoyos, cargas y propiedades de material tanto en PAME UNC 2D como en SAP2000 v23.3.1, a fin de garantizar una comparación válida de los resultados estructurales.
- Los casos de estudio seleccionados (una armadura y una viga) han sido tomados de fuentes académicas reconocidas tales como: *Matrix Analysis of Structures*, de Kassimali, A. (2022) y *Matrix Structural Analysis*, de McGuire, W. et al. (2015), además de considerar un pórtico para una estructura de nave industrial, adaptado por el tesista.
- La selección de estos modelos no responde a criterios probabilísticos sino a una elección intencional basada en su representatividad y compatibilidad con los métodos de cálculo del software propuesto.

1.6. LIMITACIONES DE LA INVESTIGACIÓN

- La comparación se restringe exclusivamente a estructuras planas bidimensionales con comportamiento elástico lineal, bajo cargas estáticas. Por tanto, no se aborda el análisis tridimensional de elementos tipo placa, cascarón; tipología mixta como pórticos con elementos curvos, tampoco se consideran efectos dinámicos o térmicos.
- Asimismo, se reconoce como limitación la ausencia de verificación experimental de resultados, ya que la evaluación se realiza únicamente por comparación numérica.
- Por otra parte, tampoco se evalúan aspectos de rendimiento computacional, tiempos de procesamiento ni consumo de recursos, ya que el foco de la investigación se limita estrictamente a la comparación de los resultados estructurales.

1.7. OBJETIVOS

Con el fin de alcanzar el propósito de la presente investigación, se plantean los siguientes objetivos generales y específicos.

1.7.1. OBJETIVO GENERAL

- Comparar la respuesta estructural obtenida mediante el software educativo PAME UNC 2D con la respuesta estructural conseguida en el software comercial SAP2000 v23.3.1, en estructuras bidimensionales sometidas a cargas estáticas, a fin de evaluar el grado de concordancia entre ambos entornos de análisis estructural.

1.7.2. OBJETIVOS ESPECÍFICOS

- Modelar una armadura, una viga y un pórtico, en el contexto bidimensional, en los programas PAME UNC 2D y SAP2000 v23.3.1, bajo condiciones equivalentes de geometría, propiedades mecánicas, apoyos y cargas.
- Determinar la respuesta estructural: desplazamientos nodales, reacciones y momentos flectores, según el tipo de estructura, en el rango elástico lineal, utilizando los programas PAME UNC 2D y SAP2000 v23.3.1 bajo condiciones equivalentes de modelado.
- Calcular e interpretar la variación porcentual de los resultados estructurales obtenidos en PAME UNC 2D respecto de los resultados conseguidos en SAP2000 v23.3.1, mediante el uso del Error Porcentual Absoluto Medio (MAPE).

1.8. DESCRIPCIÓN DEL CONTENIDOS

La presente tesis se estructura en cinco capítulos, además de considerar referencias bibliográficas y anexos. Dichos componentes están organizados de manera lógica y progresiva, con el propósito de desarrollar el estudio de forma sistemática. A continuación se describe el contenido de cada capítulo.

Capítulo I. Introducción. Tiene un enfoque descriptivo, se presenta el planteamiento y la formulación del problema, así como la hipótesis, justificación de la investigación, los alcances y la delimitación, las limitaciones, los objetivos y finaliza con el acápite de la descripción del contenido de la tesis, proporcionando un panorama general de la estructura de la presente investigación. Este capítulo contextualiza el estudio dentro del campo del análisis estructural computacional aplicado a estructuras bidimensionales.

Capítulo II. Marco Teórico. Está dedicado a sustentar conceptualmente la investigación. Incluye una revisión de antecedentes internacionales, nacionales y locales relacionados con el análisis estructural, el desarrollo de software educativo y la comparación de resultados numéricos. Asimismo, se presentan las bases teóricas fundamentales, tales como el método matricial de rigidez, la programación en MATLAB, el entorno App Designer y los principios de validación numérica. También se incorpora el acápite de las definiciones de los términos básicos que se utilizan a lo largo del trabajo.

Capítulo III. Materiales y Métodos. Inicia con la ubicación geográfica y temporal, así como los materiales y software empleados. Se detalla el enfoque metodológico, la población y muestra, así como las variables. Se describe en profundidad el procedimiento aplicado, que incluye tanto el desarrollo del software educativo PAME UNC 2D como el proceso de comparación estructural con el software SAP2000 v23.3.1. Se finaliza con las técnicas e instrumentos de recolección de datos usados para tal fin.

Capítulo IV. Análisis y Discusión de Resultados. Se presentan los datos obtenidos a partir del modelamiento y análisis estructural de tres casos de estudio: una armadura, una viga y un pórtico en el contexto bidimensional. Se comparan los resultados obtenidos entre PAME UNC 2D y SAP2000 v23.3.1, en términos de desplazamientos nodales, reacciones en los apoyos y momentos flectores, según el tipo de estructura. Además, se calcula e interpreta el Error Porcentual Absoluto Medio (MAPE) como indicador cuantitativo de concordancia. Finalmente, se corrobora la hipótesis formulada en función de los resultados obtenidos.

Capítulo V. Conclusiones y Recomendaciones. Resume los principales hallazgos del estudio y en función a ello plantea propuestas concretas para futuras mejoras del software, ampliaciones del alcance metodológico y aplicaciones en contextos académicos y profesionales.

Posterior a estos capítulos, se presentan las Referencias Bibliográficas y los Anexos, que incluyen el código principal del software desarrollado, diagramas de flujo, cuestiones teóricas afines y la solución detallada de los casos estructurales modelados.

CAPÍTULO II. MARCO TEÓRICO

2.1. ANTECEDENTES TEÓRICOS

2.1.1. ANTECEDENTES INTERNACIONALES

Faizan, A. (2016) en Sakarya University (Turquía), comparó la respuesta estructural de pórticos planos utilizando el método matricial de rigidez en MATLAB y SAP2000. Evaluó desplazamientos, fuerzas internas y momentos, encontrando diferencias porcentuales promedio inferiores al 3% en esfuerzos axiales y menores al 2% en desplazamientos y rotaciones. Concluyó que el programa desarrollado en MATLAB muestra alta concordancia con SAP2000. Por su parte, Ortiz Soto, Merma Taipe y Cadena Pérez (2021), presentaron en México, en el “Primer congreso de cuerpos académicos”, un estudio orientado al análisis estructural de armaduras bidimensionales mediante la aplicación del Método Matricial de Rigidez, empleando herramientas de programación en Excel y MATLAB. Los autores implementaron un modelo compuesto por barras de acero de sección cuadrada, evaluando desplazamientos, reacciones y fuerzas axiales. Posteriormente, los resultados obtenidos fueron comparados con los generados por el software comercial SAP2000, evidenciando una variación porcentual de 0% en la respuesta estructural obtenida.

2.1.2. ANTECEDENTES NACIONALES

Díaz, F. y Guillén, A. (2020), en su investigación realizada para la Universidad Peruana de Ciencias Aplicadas, desarrollaron un proyecto denominado TrussSolver y PlaneSolver, empleando MATLAB, para el análisis estructural mediante el método matricial de la rigidez, y luego compararon la respuesta estructural usando el programa de Aslam Kassimali, consiguiendo un error relativo de 0.00 % en los cálculos de desplazamientos nodales, fuerzas internas y reacciones. Por su parte, *Rupay, M. et al. (2023)*, en su investigación sobre el Análisis estructural de armaduras planas aplicando el método

matricial de rigidez, hicieron una comparación de la respuesta estructural usando SAP2000 y Ftool, analizando los esfuerzos internos y los desplazamientos por efecto de asentamiento, obteniendo una variación mínima de 0.05% y máxima de 0.28%, concluyendo en la similitud de la respuesta estructural de ambos programas.

2.1.3. ANTECEDENTES LOCALES

Chilón T. et al. (2023), en su investigación titulada: “Impacto del SAP y MASTAN2-MATLAB en la solución de armaduras planas y espaciales de ingeniería”, presentada ante el Vicerrectorado de Investigación y Responsabilidad Social de la Universidad Nacional de Cajamarca, desarrollaron un programa denominado “ArmTiC”, y luego compararon la respuesta estructural con SAP2000, analizando los resultados obtenidos se evidencia que el mayor porcentaje de error relativo es del orden de 0.03%. En ArmTic es necesario seguir los pasos establecidos en su manual de usuario: se establece una tabla con 19 columnas y tantas filas como elementos de barra tenga la armadura. En las 12 primeras columnas se colocan las características geométricas de los elementos, así como sus propiedades físicas. El usuario quien deberá buscar la coherencia entre las unidades a emplear, recomendándose usar: L: mm; A: mm²; F: kN; E: kN/mm². Las columnas 13, 14 y 15 son para la colocación de fuerzas en los 3 ejes. En la columna 16 se ubican posibles cargas en nodos producidas por cargas sobre el elemento. Las tres últimas columnas fueron creadas para permitir el cálculo de la matriz Kff (free – free Stiffness Matrix) y Kfs (free - supported Stiffness Matrix). Por su parte, Atalaya, M. (2019) en su estudio sobre la comparación del comportamiento estructural de vigas, pórticos y armaduras usando un programa elaborado en LUA respecto a SAP 2000 concluye que la variación máxima del comportamiento estructural de vigas es de 0.27% en el análisis de deformaciones, en tanto que para el caso de vigas se logró obtener el caso ideal de variación 0% tanto en fuerzas axiales, cortantes, momentos flectores y deformaciones.

2.2. BASES TEÓRICAS

2.2.1. RESPUESTA ESTRUCTURAL DE SISTEMAS BIDIMENSIONALES

La respuesta estructural de sistemas bidimensionales describe el comportamiento físico que presentan al ser sometidas a cargas externas, considerando desplazamientos, deformaciones, reacciones en los apoyos y momentos flectores, según el tipo de estructura en un plano específico (usualmente el plano XY o XZ).

Las estructuras bidimensionales abarcan principalmente:

Armaduras planas: constituidas por barras articuladas, predominan los esfuerzos axiales.

Vigas simples y continuas: sometidas a flexión y cortante, presentan desplazamientos verticales y rotaciones como respuestas principales.

Pórticos planos: combinan esfuerzos axiales, flexión y cortante, permitiendo desplazamientos horizontales, verticales y rotación en sus nodos.

El análisis de estas estructuras implica calcular reacciones, desplazamientos nodales y momentos flectores bajo las siguientes hipótesis:

- Comportamiento elástico lineal (ley de Hooke).
- Pequeñas deformaciones.
- Ausencia de efectos no lineales (como pandeo, plastificación o grandes desplazamientos).
- Propiedades geométricas y mecánicas constantes en cada elemento.

Estas condiciones permiten formular modelos matemáticos, generalmente aplicando el método de la rigidez, que proporcionan soluciones precisas dentro del marco del análisis elástico lineal. Este enfoque constituye la base de los procedimientos computacionales utilizados en el desarrollo de esta tesis.

2.2.2. MÉTODOS MATRICIALES

De acuerdo con Hibbeler, R. (2022), existen dos enfoques principales para analizar estructuras mediante métodos matriciales. Uno de los más utilizados es el método de la rigidez, el cual se enfoca en el análisis de desplazamientos, y el otro es el método de la flexibilidad (un método de fuerza), este no es tan versátil debido a varias razones. La principal desventaja del método de la flexibilidad es que requiere un procedimiento distinto para cada caso de análisis, lo que puede hacerlo menos eficiente en comparación con el método de la rigidez. Además, con el método de la flexibilidad, los desplazamientos no se obtienen de manera directa como sucede con el método de la rigidez. Una ventaja crucial del método de la rigidez es que simplifica la formulación de las matrices necesarias para los cálculos computacionales, resultando en una mayor eficiencia en su aplicación práctica.

2.2.2.1. EL MÉTODO MATRICIAL DE RIGIDEZ (MMR)

Llamado también como método de los desplazamientos, se fundamenta en la creación y operación de matrices de rigidez de cada uno de los elementos, para ensamblarlos posteriormente en una matriz global de toda la estructura, apoyándose de los vectores de fuerzas externas y sus correspondientes vectores de desplazamiento.

Las dos principales ventajas de la notación matricial en cálculo estructural son:

- Permite emplear métodos de cálculo de un modo más preciso, compacto y generalizado.
- Facilita un método adecuado para el análisis de estructuras que permite el desarrollo de programas y aplicaciones computacionales en diversas plataformas, entre las cuales se encuentra MATLAB (Matrix Laboratory).

2.2.2.2. RESEÑA HISTÓRICA DE LOS MÉTODOS MATRICIALES

De acuerdo con Kassimali, A. (2022), los fundamentos teóricos del análisis matricial de estructuras se remontan a los trabajos de James C. Maxwell, quien en 1864 introdujo el método de las deformaciones consistentes, y de George A. Maney, quien desarrolló en 1915 el método de la pendiente-deflexión. Estos métodos clásicos son considerados precursores directos de los métodos de flexibilidad y rigidez, respectivamente. No obstante, su aplicación práctica en la época previa al uso de computadoras estaba limitada por la necesidad de resolver grandes sistemas de ecuaciones algebraicas simultáneas de forma manual, lo que dificultaba el análisis de estructuras complejas con numerosos grados de libertad.

Para sentar las bases del empleo computacional, tuvo que esperarse hasta la revolución tecnológica de la primera mitad de los años 1940 cuando aparece el prototipo de la ENIAC, gestado en la Universidad de Pensilvania por el físico John Mauchly y el ingeniero eléctrico Presper Eckert Jr., y considerada luego para fines bélicos, a solicitud del gobierno de Estados Unidos. Aquello trajo la posterior esperanza de aplicar el potencial computacional primero en la aeronáutica y posteriormente en la ingeniería civil. ENIAC fue el punto de partida para que sus sucesoras permitieran resolver grandes sistemas de ecuaciones simultáneas de forma eficiente, favoreciendo el desarrollo definitivo de los métodos matriciales de análisis estructural.

A principios de los años 1950s, la innovación de métodos cada vez más prácticos, estuvo ligada a la industria aeronáutica, y a mediados de la década de 1950, se produjeron avances teóricos cruciales que formalizaron y extendieron los métodos existentes. John H. Argyris jugó un papel fundamental al presentar una unificación formal de los métodos de fuerza y desplazamiento en el análisis estructural. Argyris publicó inicialmente una serie de artículos en "Aircraft Engineering" entre octubre de 1954 y mayo de 1955, que

posteriormente fueron compilados en un influyente libro llamado “Energy Theorems and Structural Analysis”; además enfatizó cómo la formulación matricial era ideal para la computación automática moderna, reconociendo la necesidad de abordar problemas con cientos de incógnitas. Es en esta fase inicial se estableció que el MMR podía ser extendido para analizar no solo elementos discretos (barras) sino también un "continuo", un cambio conceptual que sentó las bases de un revolucionario método generalizado.

2.2.2.3. EVOLUCIÓN DEL MÉTODO MATRICIAL DE RIGIDEZ AL MÉTODO DE ELEMENTO FINITO.

No fue sino hasta el año 1956 que sale a la luz un artículo titulado “Stiffness and deflection Analysis of Complex Structures”, bajo la autoría de M.J. Turner, Ray W. Clough, H.C. Martin y L.J. Topp. El título adaptado al español es “Análisis de rigidez y deflexión de estructuras complejas” el cual estuvo orientado particularmente en el análisis matricial de estructuras, enfocándose en discretizar la estructura en elementos uniformes más manejables, de los cuales se puede elaborar un modelo matemático para cada caso de forma matricial, rescatando las propiedades del elemento. Este documento es ampliamente considerado como el hito que marca el inicio del MEF moderno tal como se utiliza predominantemente en programas comerciales actuales.

Un momento crucial ocurrió en 1960, cuando Ray W. Clough introdujo la terminología definitiva "Método del Elemento Finito". Esta denominación proporcionó una identidad clara y unificadora para el método en evolución, que hasta entonces se conocía por varios nombres o simplemente como una extensión de los métodos matriciales.

2.2.2.4. DESARROLLO DE ESTRUCTURAS POR EL MÉTODO MATRICIAL DE LA RIGIDEZ

Para cada elemento de una estructura a analizar, se debe determinar una matriz de rigidez, inicialmente en coordenadas locales y posteriormente en el sistema global, y después se ensamblan para conformar la llamada matriz de rigidez de la estructura $[K]$. Luego de formular los vectores totales de cargas externas $\{F\}$ y de desplazamientos externos $\{U\}$, se plantea $\{F\} = [K] \{U\}$ que es la ecuación de rigidez de la estructura, la cual gobierna el comportamiento interno de la estructura idealizada. Finalmente, al resolver el sistema de ecuaciones, se obtienen primero los desplazamientos incógnita y enseguida las reacciones en los soportes. Las fuerzas pueden calcularse usando los resultados de los desplazamientos desconocidos Kassimali, A. (2022).

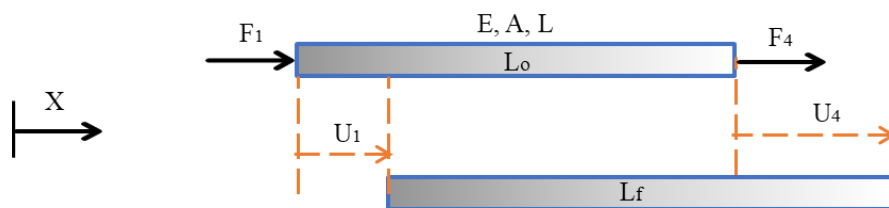
2.2.2.4.1. Matrices de rigidez por el método matricial

Como se mencionó anteriormente, es posible determinar las matrices de rigideces a partir de los conocimientos de resistencia de materiales, toda vez que este método usa principalmente elementos de barra.

➤ *Desplazamientos y fuerzas internas para el caso axial.*

Figura 1:

Relación gráfica de fuerzas y desplazamientos uniaxiales de barra



Por resistencia de materiales se sabe que:

$$\sigma = E \cdot \varepsilon$$

Donde:

σ : Esfuerzo axial

E : Módulo de elasticidad

ε : Deformación unitaria

Además:

$$\sigma = \frac{F}{A}$$

Donde:

F : Fuerza axial

A : Área de la sección transversal

También:

$$\varepsilon = \frac{\Delta L}{L}$$

Donde:

$\Delta L = U_4 - U_1$: Deformación total del elemento barra.

L : Longitud del elemento barra

De manera que, reemplazando convenientemente, se tiene:

$$\frac{F}{A} = E \cdot \frac{\Delta L}{L}$$

Y, por consiguiente:

$$F = \frac{EA}{L} \cdot \Delta L$$

Analizando el caso de F_4 :

$$F_4 = \frac{EA}{L} \cdot (U_4 - U_1)$$

Distribuyendo:

$$F_4 = \frac{EA}{L} \cdot U_4 - \frac{EA}{L} \cdot U_1$$

Pero por la primera condición de equilibrio:

$$\Sigma F = 0$$

$$F_1 + F_4 = 0$$

Por lo que

$$F_1 = -F_4$$

O bien:

$$F_1 = -\frac{EA}{L} \cdot (U_4 - U_1)$$

$$F_1 = \frac{EA}{L} \cdot U_1 - \frac{EA}{L} U_4$$

De modo que, en resumen:

$$F_1 = \frac{EA}{L} \cdot U_1 - \frac{EA}{L} U_4$$

$$F_4 = \frac{EA}{L} \cdot U_4 - \frac{EA}{L} \cdot U_1$$

Estas expresiones pueden tratarse de modo más compacto en forma matricial:

$$\begin{Bmatrix} F_1 \\ F_4 \end{Bmatrix} = \frac{EA}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{Bmatrix} U_1 \\ U_4 \end{Bmatrix}$$

$$\begin{Bmatrix} F_1 \\ F_4 \end{Bmatrix} = \begin{bmatrix} \frac{EA}{L} & -\frac{EA}{L} \\ -\frac{EA}{L} & \frac{EA}{L} \end{bmatrix} \begin{Bmatrix} U_1 \\ U_4 \end{Bmatrix} \quad (2.9)$$

Que es precisamente la ecuación matricial que gobierna el comportamiento de estructuras cargadas y deformadas axialmente, dentro del rango elástico lineal.

Generalizando, se tiene:

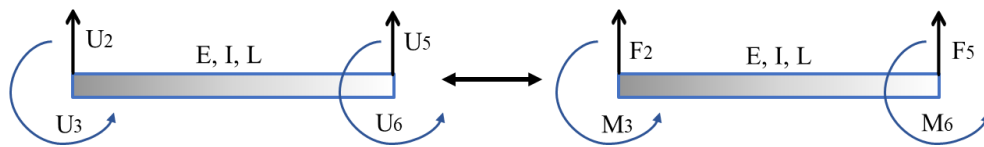
$$\{F\} = [K]\{U\} \quad (2.10)$$

➤ **Desplazamientos y fuerzas internas para el caso de vigas.**

Existe una relación única entre los desplazamientos con las fuerzas y momentos aplicados a la viga.

Figura 2:

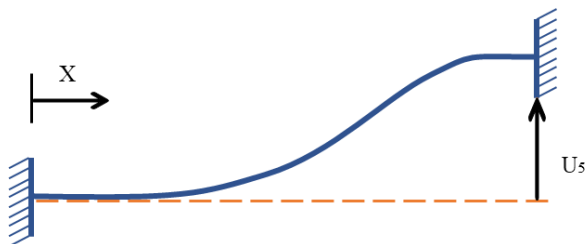
Relación gráfica entre desplazamientos y fuerzas en viga



Como caso de partida para el análisis, consideraremos solo los efectos de U_5 , y luego, de manera análoga se podrán analizar los demás casos.

Figura 3:

Deformada de la viga, por imposición de U_5



Condiciones de frontera:

Cuando la posición $x = 0$, entonces el desplazamiento $U_2 = 0$, y también lo es la pendiente de la barra, es decir el giro, así:

$$\frac{dU}{dx} = 0$$

Cuando la posición $x = L$, entonces el desplazamiento $U = U_5$, y la pendiente de la barra se mantiene como nula:

$$\frac{dU}{dx} = 0$$

Recordando la ecuación diferencial de la elástica de la viga:

$$EI \cdot \frac{d^4 U}{dx^4} = -q(x) \quad (2.11)$$

Y dado que las solicitaciones de carga son nulas, se tiene:

$$EI \cdot \frac{d^4 U}{dx^4} = 0 \quad (2.12)$$

De modo que, integrando respecto a “x” en ambos lados, la primera vez se tiene:

$$\int EI \cdot \frac{d^4 U}{dx^4} dx = \int 0 dx$$

Toda vez que E, I son características propias y constantes del elemento barra, se tiene:

$$EI \int \frac{d^4 U}{dx^4} dx = C_1$$

$$EI \frac{d^3 U}{dx^3} = C_1 \quad (2.13)$$

Volviendo a integrar respecto a x, y simplificando:

$$\int EI \frac{d^3 U}{dx^3} dx = \int C_1 dx$$

$$\int EI \frac{d^3 U}{dx^3} dx = \int C_1 dx$$

$$EI \frac{d^2 U}{dx^2} = C_1 \cdot x + C_2 \quad (2.14)$$

Repitiendo el proceso de integración:

$$\int EI \frac{d^2 U}{dx^2} dx = \int (C_1 \cdot x + C_2) dx$$

$$EI \frac{dU}{dx} = C_1 \cdot \frac{x^2}{2} + C_2 \cdot x + C_3 \quad (2.15)$$

Y con una última integración:

$$\int EI \frac{dU}{dx} dx = \int (C_1 \cdot \frac{x^2}{2} + C_2 \cdot x + C_3) dx$$

$$E.I.U = C_1 \cdot \frac{x^3}{6} + C_2 \cdot \frac{x^2}{2} + C_3 \cdot x + C_4 \quad (2.16)$$

De manera que si determinamos los valores de las constantes C_1, C_2, C_3 y C_4 , tendremos la ecuación de los desplazamientos para cada punto del elemento barra. Y para tal fin, haremos uso de las condiciones de frontera:

Condiciones de frontera para el caso $U_5 \neq 0$

Posición	Condición de frontera
$x = 0$	$U = 0$ (caso 1)
	$\frac{dU}{dx} = 0$ (caso 2)
$x = L$	$U = U_5$ (caso 3)
	$\frac{dU}{dx} = 0$ (caso 4)

Evaluando el caso 1 en la ecuación 2.16:

$$E.I.0 = C_1 \cdot \frac{0^3}{6} + C_2 \cdot \frac{0^2}{2} + C_3 \cdot 0 + C_4$$

$$C_4 = 0$$

Evaluando el caso 2 en la ecuación 2.15:

$$EI \frac{dU}{dx} = C_1 \cdot \frac{x^2}{2} + C_2 \cdot x + C_3$$

$$EI \cdot 0 = C_1 \cdot \frac{0^2}{2} + C_2 \cdot 0 + C_3$$

$$C_3 = 0$$

Evaluando el caso 3 y lo ya conocido para C_3 y C_4 , en la ecuación 2.16:

$$E \cdot I \cdot U = C_1 \cdot \frac{x^3}{6} + C_2 \cdot \frac{x^2}{2} + C_3 \cdot x + C_4$$

$$E \cdot I \cdot U_5 = C_1 \cdot \frac{L^3}{6} + C_2 \cdot \frac{L^2}{2} + 0 \cdot x + 0$$

$$E \cdot I \cdot U_5 = C_1 \cdot \frac{L^3}{6} + C_2 \cdot \frac{L^2}{2} \quad (2.17)$$

Evaluando el caso 4 en la ecuación 2.15:

$$EI \frac{dU}{dx} = C_1 \cdot \frac{x^2}{2} + C_2 \cdot x + C_3$$

$$EI \cdot 0 = C_1 \cdot \frac{L^2}{2} + C_2 \cdot L + 0$$

$$0 = C_1 \cdot \frac{L^2}{2} + C_2 \cdot L \quad (2.18)$$

Ahora bien, dos constantes ya son conocidas, C_3 y C_4 , por lo que solo quedan dos constantes por averiguar: C_1 y C_2 , las cuales serán despejadas de 2.17 y 2.18, así:

Reescribiendo C_2 de 2.18:

$$0 = C_1 \cdot \frac{L^2}{2} + C_2 \cdot L$$

$$C_2 = -C_1 \cdot \frac{L^2}{2L}$$

$$C_2 = -C_1 \cdot \frac{L}{2} \quad (2.19)$$

Ahora, reemplazando 2.19 en 2.17:

$$E.I.U_5 = C_1 \cdot \frac{L^3}{6} + C_2 \cdot \frac{L^2}{2}$$

$$E.I.U_5 = C_1 \cdot \frac{L^3}{6} + -C_1 \cdot \frac{L}{2} \cdot \frac{L^2}{2}$$

$$E.I.U_5 = C_1 \cdot \frac{L^3}{6} + -C_1 \cdot \frac{L^3}{4}$$

$$E.I.U_5 = C_1 \cdot \frac{L^3}{6} - C_1 \cdot \frac{L^3}{4}$$

$$E.I.U_5 = C_1 \cdot \left(\frac{L^3}{6} - \frac{L^3}{4} \right)$$

$$E.I.U_5 = C_1 \cdot L^3 \left(\frac{1}{6} - \frac{1}{4} \right)$$

$$E.I.U_5 = C_1 \cdot L^3 \left(-\frac{1}{12} \right)$$

$$C_1 = \frac{-12E.I.U_5}{L^3} \quad (2.20)$$

Finalmente, reemplazando (2.20) en (2.19):

$$C_2 = -C_1 \cdot \frac{L}{2}$$

$$C_2 = - \left(\frac{-12E.I.U_5}{L^3} \right) \cdot \frac{L}{2}$$

$$C_2 = \frac{6E.I.U_5}{L^2} \quad (2.21)$$

Hemos hallado las constantes C_1 , C_2 , C_3 y C_4 ; mismas que nos servirán para describir el comportamiento de la viga, tanto para cortantes, momentos flectores y deformaciones.

Del análisis de las ecuaciones diferenciales, y por resistencia de materiales, se puede obtener el comportamiento de cortantes reemplazando el valor de la constante C_1 que aparece en la ecuación 2.20, en la ecuación 2.13:

$$V(x) = EI \frac{d^3 U}{dx^3} = C_1$$

$$V(x) = \frac{-12E \cdot I}{L^3} \cdot U_5 \quad (2.22)$$

De manera análoga, se puede obtener el comportamiento de momentos flectores, reemplazando las expresiones obtenidas de C_1 y C_2 en la ecuación (2.14):

$$EI \frac{d^2 U}{dx^2} = C_1 \cdot x + C_2$$

Pero:

$$M(x) = EI \frac{d^2 U}{dx^2}$$

De manera que:

$$M(x) = C_1 \cdot x + C_2$$

$$M(x) = \frac{-12E \cdot I \cdot U_5}{L^3} \cdot x + \frac{6E \cdot I \cdot U_5}{L^2} \quad (2.23)$$

A partir de dichas funciones, se pueden graficar ambas solicitaciones (cortante y momento flector), así:

Figura 4:

Diagrama de cortantes $V(x)$

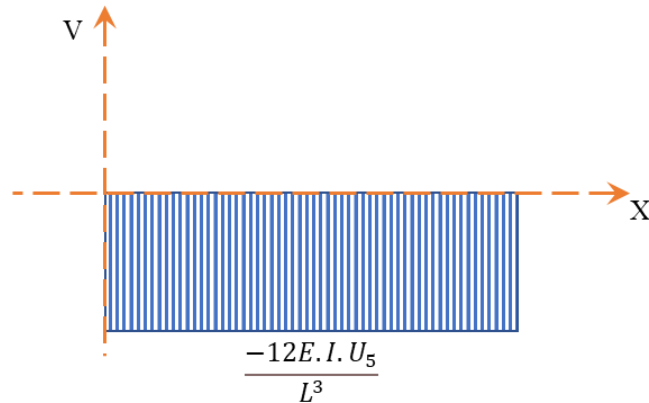
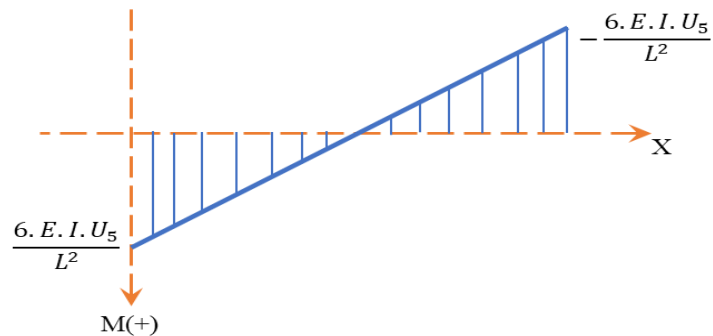


Figura 5:

Diagrama de momentos $M(x)$



Con las gráficas definidas, se pueden establecer las solicitaciones equivalentes en los extremos del elemento barra.

Considerando la siguiente convención de signos:

Figura 6:

Convención de signos para cortantes y momentos flectores en viga

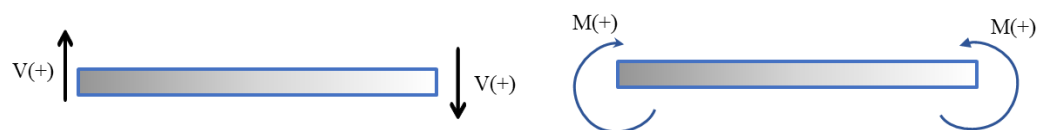
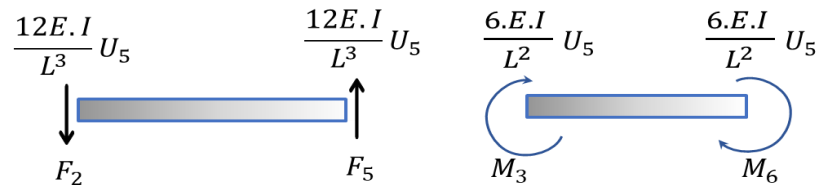


Figura 7:

Solicitaciones de cortantes y momentos en nodos extremos de viga.



Luego, se pueden identificar las sollicitaciones en cada nodo extremo de la viga, debido a la imposición de U_5 .

Solicitaciones equivalentes en los extremos de la viga debidas a U_5

Solicitación	Expresión resultante
F_2	$-\frac{12E.I}{L^3} \cdot U_5$
M_3	$-\frac{6.E.I}{L^2} \cdot U_5$
F_5	$\frac{12E.I}{L^3} \cdot U_5$
M_6	$-\frac{6.E.I}{L^2} \cdot U_5$

Se puede hacer el mismo análisis para los otros grados de libertad U_2 , U_3 y U_6 , partiendo de la ecuación de la elástica de la , para obtener las siguientes tablas:

Solicitaciones equivalentes en los extremos de la viga debidas a U_2

Solicitación	Expresión resultante
F_2	$\frac{12E.I}{L^3} \cdot U_2$
M_3	$\frac{6.E.I}{L^2} \cdot U_2$
F_5	$-\frac{12E.I}{L^3} \cdot U_2$
M_6	$\frac{6.E.I}{L^2} \cdot U_2$

Solicitaciones equivalentes en los extremos de la viga debidas a U_3

Solicitud	Expresión resultante
F_2	$\frac{6.E.I}{L^2}.U_3$
M_3	$\frac{4.E.I}{L}.U_3$
F_5	$-\frac{6.E.I}{L^2}.U_3$
M_6	$\frac{2.E.I}{L}.U_3$

Solicitaciones equivalentes en los extremos de la viga debidas a U_6

Solicitud	Expresión resultante
F_2	$\frac{6.E.I}{L^2}.U_6$
M_3	$\frac{2.E.I}{L}.U_6$
F_5	$-\frac{6.E.I}{L^2}.U_6$
M_6	$\frac{4.E.I}{L}.U_6$

Luego, por el principio de superposición se pueden acumular los efectos de las solicitudes en una única ecuación, obteniéndose:

$$F_2 = \frac{12E.I}{L^3}.U_2 + \frac{6.E.I}{L^2}.U_3 - \frac{12E.I}{L^3}.U_5 + \frac{6.E.I}{L^2}.U_6$$

$$M_3 = \frac{6.E.I}{L^2}.U_2 + \frac{4.E.I}{L}.U_3 - \frac{6.E.I}{L^2}.U_5 + \frac{2.E.I}{L}.U_6$$

$$F_5 = -\frac{12E.I}{L^3}.U_2 - \frac{6.E.I}{L^2}.U_3 + \frac{12E.I}{L^3}.U_5 - \frac{6.E.I}{L^2}.U_6$$

$$M_6 = \frac{6.E.I}{L^2}.U_2 + \frac{2.E.I}{L}.U_3 - \frac{6.E.I}{L^2}.U_5 + \frac{4.E.I}{L}.U_6$$

Estas cuatro ecuaciones se pueden ordenar convenientemente de forma matricial, así:

$$\begin{Bmatrix} F_2 \\ M_3 \\ F_5 \\ M_6 \end{Bmatrix} = \begin{bmatrix} \frac{12E.I}{L^3} & \frac{6.E.I}{L^2} & -\frac{12E.I}{L^3} & \frac{6.E.I}{L^2} \\ \frac{6.E.I}{L^2} & \frac{4.E.I}{L} & -\frac{6.E.I}{L^2} & \frac{2.E.I}{L} \\ -\frac{12E.I}{L^3} & -\frac{6.E.I}{L^2} & \frac{12E.I}{L^3} & -\frac{6.E.I}{L^2} \\ \frac{6.E.I}{L^2} & \frac{2.E.I}{L} & -\frac{6.E.I}{L^2} & \frac{4.E.I}{L} \end{bmatrix} \begin{Bmatrix} U_2 \\ U_3 \\ U_5 \\ U_6 \end{Bmatrix} \quad (2.24)$$

Siendo la matriz de rigidez de la viga:

$$[K_{viga}] = \begin{bmatrix} \frac{12E.I}{L^3} & \frac{6.E.I}{L^2} & -\frac{12E.I}{L^3} & \frac{6.E.I}{L^2} \\ \frac{6.E.I}{L^2} & \frac{4.E.I}{L} & -\frac{6.E.I}{L^2} & \frac{2.E.I}{L} \\ -\frac{12E.I}{L^3} & -\frac{6.E.I}{L^2} & \frac{12E.I}{L^3} & -\frac{6.E.I}{L^2} \\ \frac{6.E.I}{L^2} & \frac{2.E.I}{L} & -\frac{6.E.I}{L^2} & \frac{4.E.I}{L} \end{bmatrix}$$

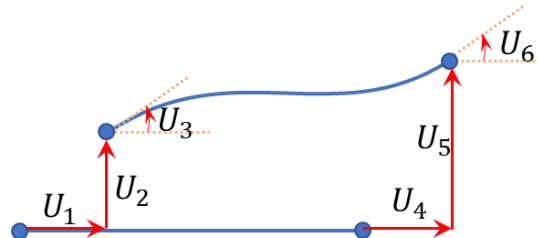
Y lo que resulta más interesante, es que se pueden superponer los efectos axiales, si lo que se desea es analizar el caso de pórticos como se verá a continuación.

➤ **Desplazamientos y fuerzas internas para el caso de pórticos.**

Como se mencionó antes, basta con agregar el caso de las solicitaciones axiales, para obtener las solicitaciones necesarias para el análisis de pórticos.

Figura 8:

Efectos de desplazamientos superpuestos para análisis de pórticos



Partiendo de la ecuación matricial que gobierna el comportamiento de las vigas:

$$\begin{Bmatrix} F_2 \\ M_3 \\ F_5 \\ M_6 \end{Bmatrix} = \begin{bmatrix} \frac{12E.I}{L^3} & \frac{6.E.I}{L^2} & -\frac{12E.I}{L^3} & \frac{6.E.I}{L^2} \\ \frac{6.E.I}{L^2} & \frac{4.E.I}{L} & -\frac{6.E.I}{L^2} & \frac{2.E.I}{L} \\ -\frac{12E.I}{L^3} & -\frac{6.E.I}{L^2} & \frac{12E.I}{L^3} & -\frac{6.E.I}{L^2} \\ \frac{6.E.I}{L^2} & \frac{2.E.I}{L} & -\frac{6.E.I}{L^2} & \frac{4.E.I}{L} \end{bmatrix} \begin{Bmatrix} U_2 \\ U_3 \\ U_5 \\ U_6 \end{Bmatrix}$$

Y agregando las ubicaciones para las consideraciones axiales:

$$\begin{Bmatrix} F_2 \\ M_3 \\ F_5 \\ M_6 \end{Bmatrix} = \begin{bmatrix} \dots & \frac{12E.I}{L^3} & \frac{6.E.I}{L^2} & \dots & -\frac{12E.I}{L^3} & \frac{6.E.I}{L^2} \\ \dots & \frac{6.E.I}{L^2} & \frac{4.E.I}{L} & \dots & -\frac{6.E.I}{L^2} & \frac{2.E.I}{L} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & -\frac{12E.I}{L^3} & -\frac{6.E.I}{L^2} & \dots & \frac{12E.I}{L^3} & -\frac{6.E.I}{L^2} \\ \dots & \frac{6.E.I}{L^2} & \frac{2.E.I}{L} & \dots & -\frac{6.E.I}{L^2} & \frac{4.E.I}{L} \end{bmatrix} \begin{Bmatrix} U_2 \\ U_3 \\ U_5 \\ U_6 \end{Bmatrix}$$

Completando las consideraciones axiales de F_1 y F_4 de la ecuación 2.9

$$\begin{Bmatrix} F_1 \\ F_2 \\ M_3 \\ F_4 \\ F_5 \\ M_6 \end{Bmatrix} = \begin{bmatrix} \frac{EA}{L} & \dots & \dots & -\frac{EA}{L} & \dots & \dots \\ \dots & \frac{12E.I}{L^3} & \frac{6.E.I}{L^2} & \dots & -\frac{12E.I}{L^3} & \frac{6.E.I}{L^2} \\ \dots & \frac{6.E.I}{L^2} & \frac{4.E.I}{L} & \dots & -\frac{6.E.I}{L^2} & \frac{2.E.I}{L} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ -\frac{EA}{L} & \dots & \dots & \frac{EA}{L} & \dots & \dots \\ \dots & -\frac{12E.I}{L^3} & -\frac{6.E.I}{L^2} & \dots & \frac{12E.I}{L^3} & -\frac{6.E.I}{L^2} \\ \dots & \frac{6.E.I}{L^2} & \frac{2.E.I}{L} & \dots & -\frac{6.E.I}{L^2} & \frac{4.E.I}{L} \end{bmatrix} \begin{Bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \\ U_5 \\ U_6 \end{Bmatrix}$$

Completando con ceros los espacios vacíos:

$$\begin{Bmatrix} F_1 \\ F_2 \\ M_3 \\ F_4 \\ F_5 \\ M_6 \end{Bmatrix} = \begin{bmatrix} \frac{EA}{L} & 0 & 0 & -\frac{EA}{L} & 0 & 0 \\ 0 & \frac{12EI}{L^3} & \frac{6EI}{L^2} & 0 & -\frac{12EI}{L^3} & \frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{4EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{2EI}{L} \\ -\frac{EA}{L} & 0 & 0 & \frac{EA}{L} & 0 & 0 \\ 0 & -\frac{12EI}{L^3} & -\frac{6EI}{L^2} & 0 & \frac{12EI}{L^3} & -\frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{2EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{4EI}{L} \end{bmatrix} \begin{Bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \\ U_5 \\ U_6 \end{Bmatrix} \quad (2.25)$$

Que precisamente es la ecuación matricial para pórticos, donde la matriz de rigidez local de cada elemento barra del pórtico es:

$$[K^{Local}] = \begin{bmatrix} \frac{EA}{L} & 0 & 0 & -\frac{EA}{L} & 0 & 0 \\ 0 & \frac{12EI}{L^3} & \frac{6EI}{L^2} & 0 & -\frac{12EI}{L^3} & \frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{4EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{2EI}{L} \\ -\frac{EA}{L} & 0 & 0 & \frac{EA}{L} & 0 & 0 \\ 0 & -\frac{12EI}{L^3} & -\frac{6EI}{L^2} & 0 & \frac{12EI}{L^3} & -\frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{2EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{4EI}{L} \end{bmatrix} \quad (2.26)$$

Luego, se puede realizar la transformación de coordenadas locales de las barras a coordenadas globales, lo cual responde a un procedimiento geométrico y/o trigonométrico.

MATRIZ DE TRANSFORMACIÓN GENERALIZADA:

$$T = \begin{bmatrix} \cos(\alpha) & \sin(\alpha) & 0 & 0 & 0 & 0 \\ -\sin(\alpha) & \cos(\alpha) & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos(\alpha) & \sin(\alpha) & 0 \\ 0 & 0 & 0 & -\sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

TRANSPUESTA DE MATRIZ DE TRANSFORMACIÓN GENERALIZADA:

$$T^T = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 & 0 & 0 & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos(\alpha) & -\sin(\alpha) & 0 \\ 0 & 0 & 0 & \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

MATRIZ DE RIGIDEZ EN COORDENADAS GLOBALES

$$[K] = [T][k][T]^T$$

MATRIZ DE RIGIDEZ ENSAMBLADA Y ORDENADA

$$[K] = \begin{bmatrix} K_{LL} & K_{LR} \\ K_{RL} & K_{RR} \end{bmatrix}$$

CÁLCULO DE DESPLAZAMIENTOS

De la formulación del método matricial se tiene que:

$$\{F\} = [K]\{U\}$$
$$\begin{Bmatrix} F_L \\ - \\ F_R \end{Bmatrix} = \begin{bmatrix} K_{LL} & K_{LR} \\ K_{RL} & K_{RR} \end{bmatrix} \begin{Bmatrix} U_L \\ - \\ U_R \end{Bmatrix}$$

Como los desplazamientos en los grados de libertad restringidos en este caso son nulo (no hay asentamientos), entonces la matriz equivalente está dada por:

$$\begin{Bmatrix} F_L \\ - \\ F_R \end{Bmatrix} = \begin{bmatrix} K_{LL} & K_{LR} \\ K_{RL} & K_{RR} \end{bmatrix} \begin{Bmatrix} U_L \\ - \\ 0 \end{Bmatrix}$$

Podemos transformar esta ecuación matricial, en dos nuevas ecuaciones:

$$\{F_L\} = [K_{LL}]\{U_L\} \quad \dots(1)$$

$$\{F_R\} = [K_{RL}]\{U_L\} \quad \dots(2)$$

$$\text{Si } [A][X] = [B]$$

$$\text{Entonces: } [X] = [A]^{-1}[B]$$

Aplicando esta propiedad fundamental de álgebra matricial a la ecuación (1), obtenemos:

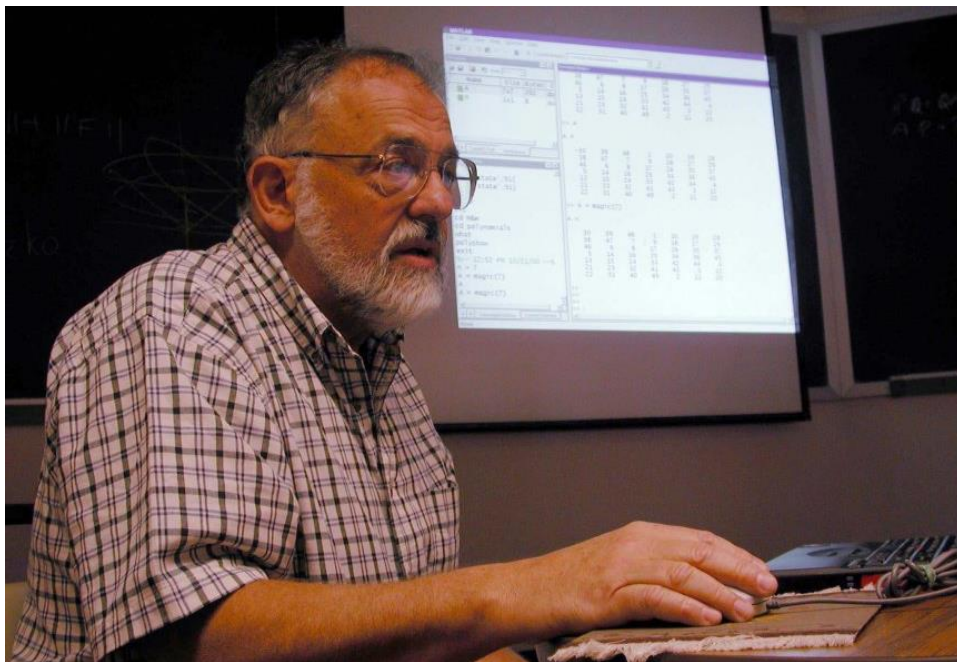
$$\{U_L\} = [K_{LL}]^{-1}\{F_L\}$$

2.2.3. MATLAB

Acrónimo de Matrix Laboratory. Es un entorno computacional de alto nivel, diseñado principalmente para cálculos numéricos, desarrollo de algoritmos, modelado, simulación, elaboración de gráficas y programación científica. Desarrollado inicialmente por Cleve Moler en la década de 1980, MATLAB ha evolucionado hasta convertirse en una herramienta ampliamente utilizada en ingeniería, matemáticas aplicadas, física y otras disciplinas científicas, especialmente en las que requieren manipulación intensiva de matrices y vectores. Posteriormente se cofundó Mathworks para continuar con su desarrollo.

Figura 9:

Cleve Moler, desarrollador principal de MATLAB.



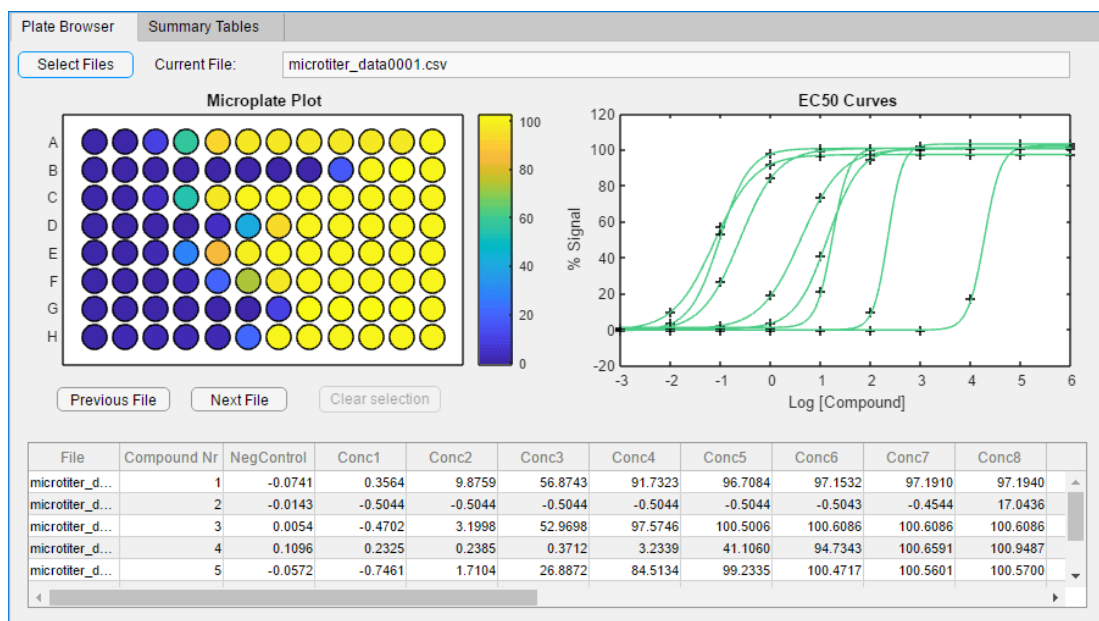
Fuente: nhigham.com. Moler, mostrando uso de Matlab en el año 2000.

El núcleo de MATLAB está basado en el álgebra matricial, lo que lo convierte en un recurso idóneo para el análisis estructural mediante el Método Matricial de Rigidez. Sus funciones nativas permiten realizar operaciones de suma, multiplicación, inversión y descomposición de matrices de forma eficiente.

En MATLAB se pueden desarrollar aplicaciones que requieran interfaces gráficas conocidas como GUI (Graphic User Interface), mediante el complemento de GUIDE, y en los últimos años, Mathworks ha creado una plataforma que brinda mayor versatilidad en cuanto a sus herramientas, se trata de App Designer, o diseñador de aplicaciones, el cual es un entorno de desarrollo interactivo, específicamente para el desarrollo de aplicaciones con una interfaz gráfica.

Figura 10:

Modelo de interfaz creada en App Designer



Fuente: <https://la.mathworks.com/products/matlab/app-designer.html>

La principal ventaja de este laboratorio de matrices de MATLAB es el empleo de un grupo de herramientas gráficas desde una librería de componentes o toolboxes, los cuales permiten una mejor interacción con el usuario.

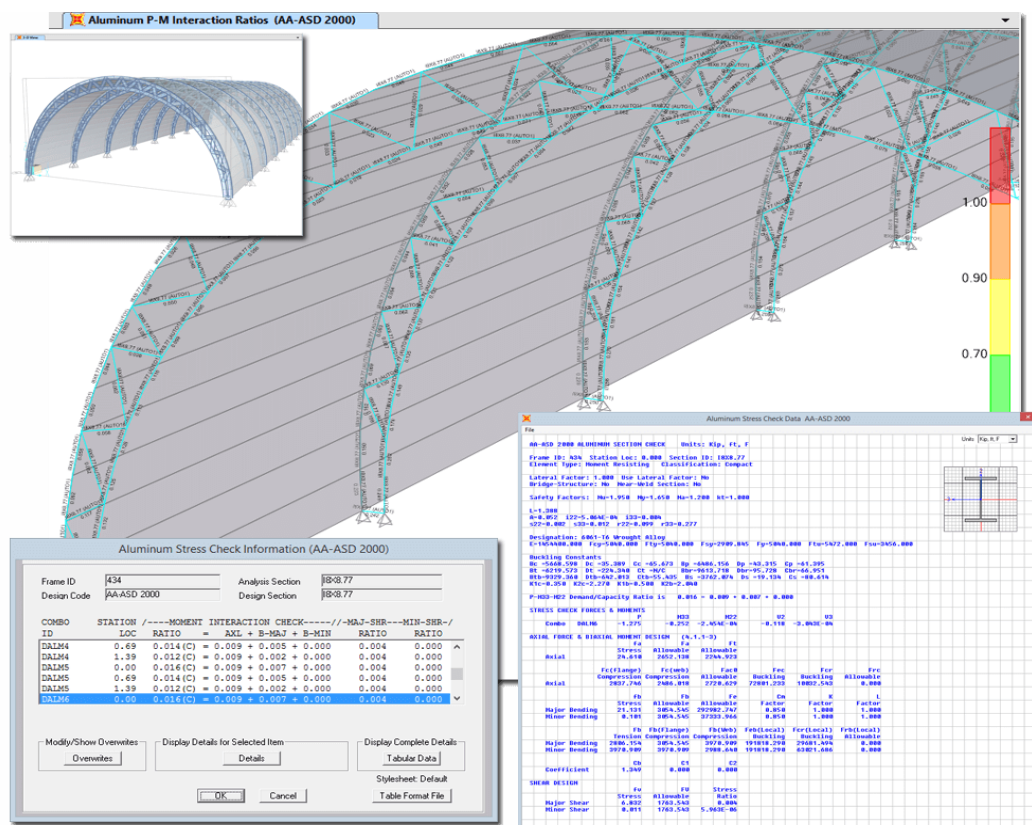
En el entorno de MATLAB, se considera a una matriz como un arreglo rectangular de números. Las expresiones que presentan un único componente se conocen como escalares, mientras que las matrices que poseen solo una columna o una sola fila se conocen como vectores.

2.2.4. SAP2000 V23.3.1

Acónimo de Structural Analysis Program, es un software de Computers and Structures, Inc. (CSI) que, si bien es cierto, se fundamenta en el método de elementos finitos, también es capaz de adaptarse al método matricial de rigidez. Cuenta con una interfaz gráfica 3D. Está diseñado para realizar la modelación, análisis y dimensionamiento de una amplia gama de problemas relacionados con la ingeniería estructural. Su flexibilidad para analizar distintos tipos de estructuras, junto con su capacidad de cálculo, hacen de SAP2000 V23.3.1 una herramienta indispensable para muchos ingenieros. Gracias a su versatilidad, permite modelar estructuras de diversa índole, como puentes, edificios, estadios, presas, entre otros. Desde modelos estáticos sencillos en 2D hasta estructuras complejas de gran tamaño que requieren análisis no lineales avanzados, SAP2000 V23.3.1 se presenta como una herramienta eficiente y altamente productiva para los ingenieros estructurales.

Figura 11:

Ejemplo de análisis de estructuras de aluminio en SAP2000 V23.3.1



2.3. DEFINICIÓN BÁSICA DE TÉRMINOS

2.3.1. COMPARACIÓN: Acción de examinar dos o más objetos, procesos o resultados con el fin de identificar semejanzas y diferencias. (RAE, 2024).

2.3.2. ESTRUCTURA. Conjunto de elementos que resisten las solicitaciones de cargas y otros agentes externos. Su finalidad es soportar y transmitir estas acciones hacia los apoyos de manera segura y eficiente, conservando sus propiedades físicas y mecánicas a lo largo del tiempo y asegurando su estabilidad y funcionalidad. (Navarro, C., 2021).

2.3.2.1. ESTRUCTURA ARTICULADA. Estructura formada por barras unidas mediante nudos articulados, los cuales impiden desplazamientos relativos entre barras que concurren a él. Dichos enlaces no permiten transmitir momentos flectores de unas piezas a otras, las barras trabajan fundamentalmente a solicitaciones axiales (Cervera, M. et al., 2015)

2.3.2.2. ESTRUCTURA RETICULADA. Estructura formada por piezas prismáticas unidas entre sí por nudos rígidos. Los nudos rígidos impiden los desplazamientos y los giros relativos de las barras que concurren a él. Dado que este enlace sí transmite momentos de una pieza a la otra, las barras trabajan fundamentalmente a flexión (Cervera, M. et al., 2015).

2.3.3. MAPE (ERROR PORCENTUAL ABSOLUTO MEDIO). Del inglés: Mean Absolute Percentage Error, es un indicador estadístico usado para evaluar la precisión de un modelo al comparar sus resultados con valores de referencia. MAPE expresa el error en porcentaje, calculando la media de los errores relativos entre los valores observados y los estimados.

$$MAPE = \frac{100}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right|$$

Donde y_i representa el valor de referencia y $\hat{y}_i$ el valor estimado.

2.3.4. MATLAB. Surge de la abreviación de Matrix Laboratory, que adaptado al español significa laboratorio de matrices, está considerado como un lenguaje de alto desempeño y destaca su gran potencia para el trabajo con vectores y matrices.

2.3.5. MODELAMIENTO ESTRUCTURAL. Idealización, representación virtual o matemática de aspectos parciales de una estructura física, pero con condiciones aceptables, a fin de predecir reacciones, esfuerzos y deformaciones; razón por la cual deben recopilar información relevante de la estructura, como su geometría y propiedades inherentes. (Navarro, C., 2021).

2.3.6. PAME UNC 2D. Acrónimo de Programa de Análisis Matricial de Estructuras, desarrollado como parte de la presente tesis, para optar por el título de ingeniero civil, en la Universidad Nacional de Cajamarca. El programa en cuestión aborda la simplificación bidimensional de estructuras, de ahí que finalice con “2D”.

2.3.7. RESPUESTA ESTRUCTURAL. Conjunto de resultados de una estructura cuando está sometida a cargas. Incluye desplazamientos, deformaciones, esfuerzos y reacciones en los apoyos.

2.3.8. SAP2000 V23.3.1. Surge de la abreviación de Structural Analysis Program 2000, es un software desarrollado por Computers and Structures, Inc. (CSI). Aunque está basado en el método de elementos finitos, también puede aplicar el método matricial de rigidez.

2.3.9 SOFTWARE EDUCATIVO. Programa informático elaborado con fines netamente académicos. Tiene como finalidad ayudar a comprender mejor los conceptos teóricos y aplicarlos en el ámbito práctico, a través de simulaciones y entorno virtual.

CAPÍTULO III. MATERIALES Y MÉTODOS

3.1. UBICACIÓN GEOGRÁFICA Y TEMPORAL

La presente investigación se realizó en la ciudad de Cajamarca, perteneciente al distrito, provincia y región del mismo nombre, entre los meses de abril y diciembre del 2024.

Figura 12:

Cajamarca – nivel de departamento, contextualizado en el mapa de Perú.



Fuente: Plataforma Nacional de Datos Georreferenciados Geo Perú. (2025).

Figura 13:

Provincia de Cajamarca, en mapa del departamento de Cajamarca



Fuente: paintmaps.com. (2025).

Figura 14:

Distrito de Cajamarca, dentro de la provincia del mismo nombre.



Fuente: mef.gob.pe. (2025)

3.2. MATERIALES Y SOFTWARE

3.2.1. MATERIALES

- Laptop personal con capacidad de procesamiento suficiente para la ejecución de los programas y visualización de resultados.
- Libros físicos y digitales de análisis matricial de estructuras.

3.2.2. SOFTWARE

- PAME UNC 2D, diseñado por el tesista, basado en el método matricial de rigidez para estructuras bidimensionales.
- SAP2000 en su versión 23.3.1, utilizado como referente de comparación de los resultados estructurales obtenidos.
- Microsoft Excel para el procesamiento estadístico de los datos (MAPE).

3.3. METODOLOGÍA

3.3.1. Enfoque

El presente estudio adopta un enfoque cuantitativo, dado que analiza, procesa y compara resultados numéricos obtenidos a partir de simulaciones computacionales de estructuras bidimensionales, mediante la determinación de la variación porcentual.

3.3.2. Diseño

El diseño de la investigación es no experimental, comparativo, ya que no se manipulan directamente variables independientes, sino que se observan y comparan los resultados estructurales generados por dos programas computacionales (PAME UNC 2D y SAP2000 V23.3.1), aplicando los mismos parámetros de entrada en ambos sistemas.

3.3.3. Tipo

La investigación es de tipo aplicada, pues tiene como propósito evaluar la precisión de un software educativo desarrollado bajo el enfoque del análisis matricial de estructuras, en comparación con un programa comercial consolidado, con el fin de aportar herramientas de apoyo al proceso formativo de profesionales en ingeniería estructural.

3.3.4. Nivel

El nivel de la investigación es descriptivo-comparativo. Se describen los resultados obtenidos por cada programa de análisis para estructuras específicas, y posteriormente se efectúa una comparación sistemática entre las respuestas estructurales generadas, utilizando indicadores cuantitativos de error relativo.

3.3.5. Método

Se emplea el método no experimental de tipo comparativa, basado en la simulación numérica computacional con el método matricial de rigidez aplicado a estructuras bidimensionales. Las simulaciones fueron ejecutadas bajo las mismas condiciones de carga, geometría y propiedades mecánicas, con el objetivo de contrastar la consistencia de los resultados entre el software educativo y el programa de referencia.

3.4. POBLACIÓN DE ESTUDIO

El universo de estudio corresponde a estructuras bidimensionales susceptibles de análisis mediante el método matricial de rigidez y el método de elementos finitos para estructuras de elementos tipo barra, incluyendo tipologías de cerchas, vigas y pórticos.

3.5. MUESTRA

Se realizó un muestreo no probabilístico intencional. La muestra seleccionada está compuesta por los tres tipos de estructuras bidimensionales:

- **Cercha:** Corresponde a un ejercicio extraído del libro "Matrix Structural Analysis", de McGuire, W. et al. (2015).
- **Viga:** Se utilizó un ejercicio del libro "Matrix Analysis of Structures", de Aslam Kassimali (2022), específicamente el ejemplo 5.6.
- **Pórtico:** Es un ejercicio de marco para una nave industrial con una carga de impacto generada por el polipasto, adaptado por el tesista.

3.6. VARIABLES DE ESTUDIO

3.6.1. VARIABLE INDEPENDIENTE

- Entornos de los programas de análisis estructural: PAME UNC 2D y SAP2000 v23.3.1.

Fundamento: el entorno de cada software es el conjunto de características predefinidas de una plataforma informática diseñada para resolver problemas de análisis estructural.

Definición operacional:

La variable independiente está constituida por dos entornos computacionales:

- PAME UNC 2D.
- SAP2000 v23.3.1.

3.6.2. VARIABLE DEPENDIENTE

- Respuesta estructural.

Fundamento: La respuesta estructural depende del tipo de estructura, geometría y cargas aplicadas.

Definición operacional:

En este estudio, la respuesta estructural se mide en términos de:

- Desplazamientos nodales (en mm).
- Reacciones en los apoyos (en kN).
- Momentos flectores (según el tipo de estructura, kN-m).
- Variación porcentual promedio entre los resultados obtenidos en PAME UNC 2D y SAP2000 v23.3.1, según la ecuación del Error Porcentual Absoluto Medio (MAPE).

3.7. PROCEDIMIENTO

Para lograr el objetivo principal de comparar la respuesta estructural de estructuras bidimensionales, obtenidas en PAME UNC 2D y SAP2000 V23.3.1, se consideran dos escenarios: el primero, en el que se analizan los tres modelos utilizando solamente PAME UNC 2D, desde el ingreso de datos, procesamiento, hasta la obtención de resultados; y el segundo, en el que se analizan los mismos tres modelos utilizando únicamente SAP2000 V23.3.1, con las mismas directrices del caso anterior, replicando las propiedades geométricas, mecánicas y condiciones de carga. Esta es la base para la comparación de la respuesta estructural vista en el análisis y discusión de resultados contemplados en el próximo capítulo.

La comparación se enfocó en los siguientes parámetros de respuesta estructural:

- Reacciones
- Desplazamientos nodales
- Momentos flectores (según el tipo de estructura).
- Variaciones porcentuales entre los resultados de ambos programas.

3.7.1. DESARROLLO DE CASOS EN PAME UNC 2D

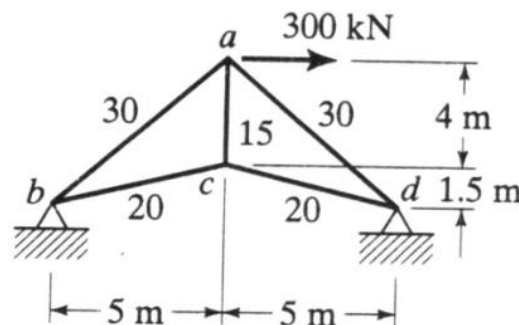
CASO 1: ARMADURA

Se analiza el caso del ejercicio 3.1(c) del libro “Matrix Structural Analysis”, de McGuire, W. et al. (2015), presente en la investigación de Chilón T. et al. (2023).

Enunciado: Determine los desplazamientos, reacciones y fuerzas en las barras para la armadura mostrada. Las áreas de la sección transversal ($mm^2 \times 10^3$) están mostrados en cada barra. Considere $E=200,000$ MPa.

Figura 15:

Ejercicio 3.1 de “Matrix Structural Analysis”



Fuente: (McGuire, W. et al. 2015) & (Chilón T. et al., 2023)

Solución con PAME UNC 2D:

El procedimiento completo de este caso, se adjunta en el Anexo B del presente documento. Por cuestiones prácticas, se describe la síntesis de los pasos, contemplando el ingreso de datos, la obtención de los grados de libertad libres, matrices de rigidez en coordenadas globales y locales, hasta llegar a la respuesta estructural de la cercha, que contempla: reacciones de apoyos, esfuerzos axiales y desplazamientos nodales.

Resumen de pasos:

a) Archivo>>Nuevo>> “ARMADURA”>>Información general.

Tabla 1:

Información general de inicio de ejercicio analizado

Identificador	información
Título del proyecto	Tesis_ejemplo_1
Número de nodos	4
Número de barras	5

b) Ingreso de coordenadas y geometría de barras:

Tabla 2:

Ingreso de coordenadas nodales

Nodo	Abscisa (mm)	Ordenada (mm)
1	5000	5500
2	5000	1500
3	0	0
4	10000	0

Tabla 3:

Ingreso de geometría de las barras

Barra	Nodo Inicial	Nodo Final
1	3	1
2	4	1
3	2	1
4	3	2
5	4	2

c) Generación del módulo de elasticidad. $E=200,000$ MPa, pero en nuestras unidades consistentes se usará KN/mm^2 , por lo que $E = 200 \text{ KN}/\text{mm}^2$

d) Geometría de la sección del material.

Tabla 4:

Secciones típicas usadas en el modelo de cercha

ID	ÁREA (mm^2)
A_1	15000
A_2	20000
A_3	30000

e) Asignación del material a cada elemento barra.

Para el ejemplo, todas las barras tienen el mismo módulo de elasticidad.

$$E = 200 \text{ KN}/\text{mm}^2$$

f) Asignación de las secciones correspondientes a cada barra.

Tabla 5:

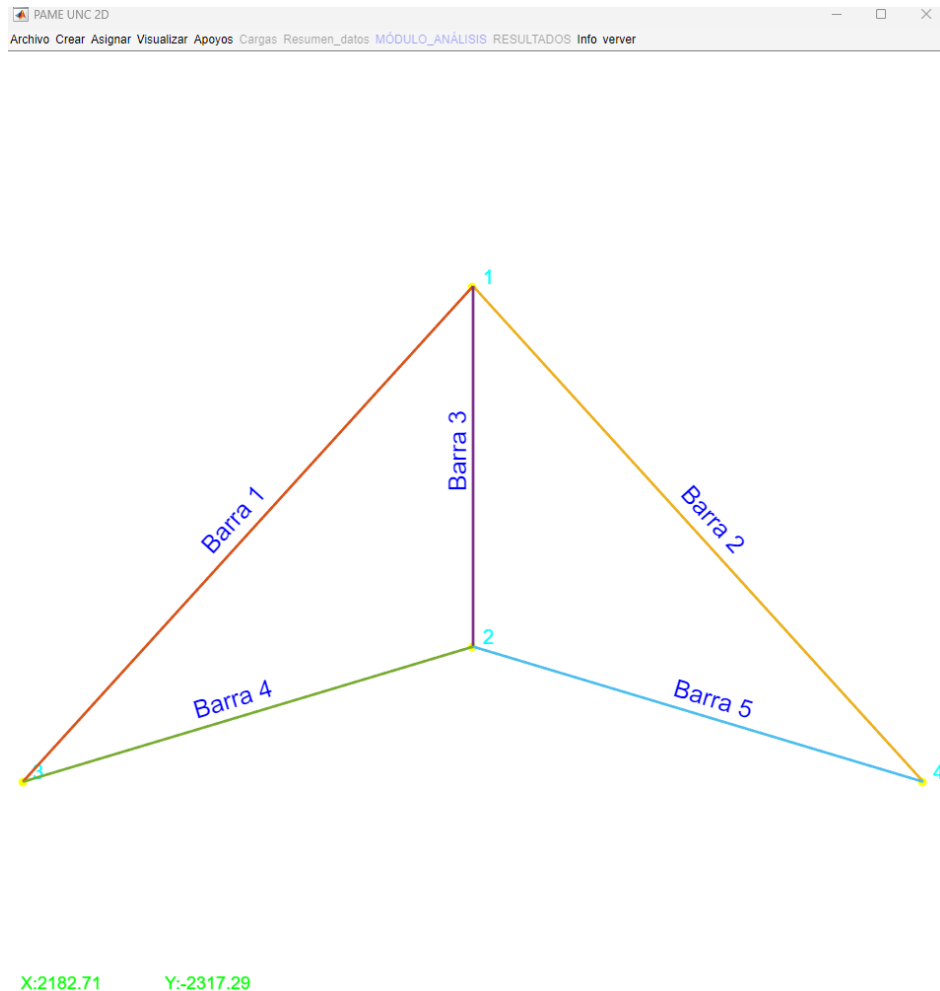
Secciones asignadas a cada barra

N°BARRA	SECCIÓN (mm ²)
BARRA 1:	A3=30000
BARRA 2:	A3=30000
BARRA 3:	A1=15000
BARRA 4:	A2=20000
BARRA 5:	A2=20000

g) Visualización de la configuración de la estructura.

Figura 16:

Visualización de la armadura en el lienzo principal

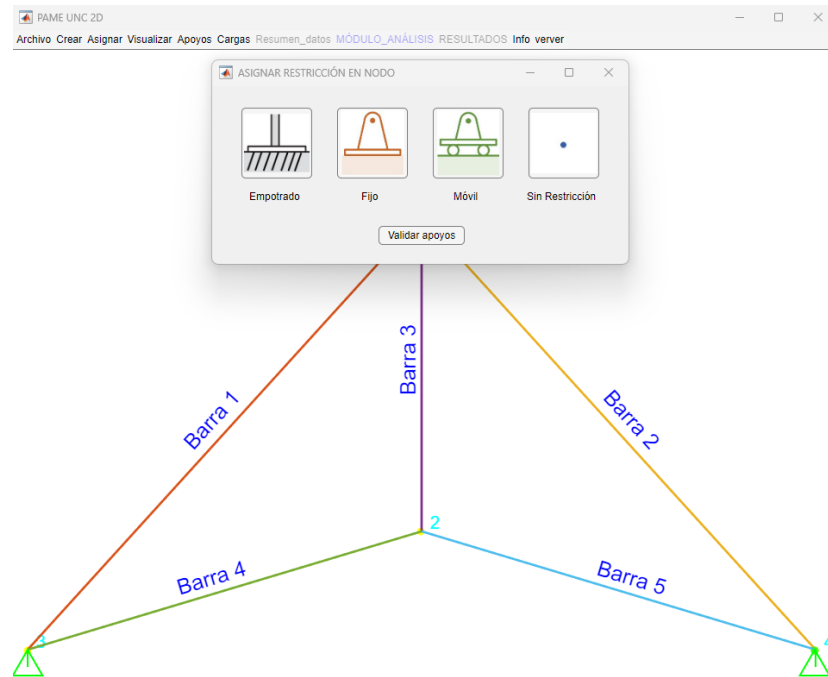


h) Asignación de apoyos:

Apoyos >> Dibujar apoyos

Figura 17:

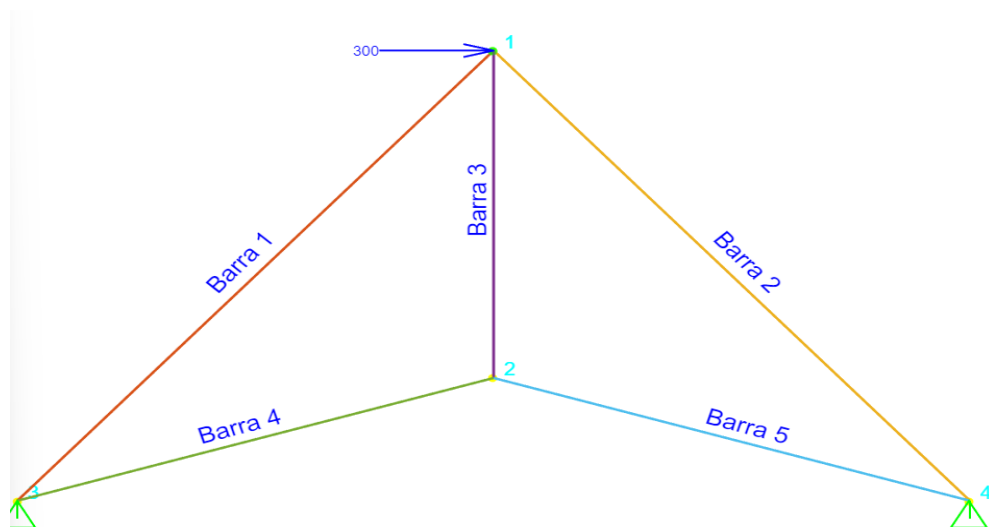
Asignación y validación interactiva de apoyos



i) Asignación de cargas nodales

Figura 18:

Asignación de carga en la estructura sobre el lienzo principal



j) Módulo de análisis:

DETERMINACION DE LOS GRADOS DE LIBERTAD LIBRES (G.D.L.L)

Tipo de estructura	:	Armadura
Número de coordenadas por nodo (NCJT)	:	2
Número de nodos (NJ)	:	4
Número de apoyos	:	2

Tabla 6:

Restricciones en nodos para caso cercha

Nodo	Restricción X	Restricción Y
3	1	1
4	1	1

Nota. “1” indica que el grado está restringido.

Número de restricciones (NR): 4.

Grados de libertad libres:

$$GDLL = (NCJT) \times (NJ) - NR.$$

$$GDLL=4$$

Tabla 7:

ID de ejes locales para cada nodo

ID.COORD	NODO
1	1
2	1
3	2
4	2
5	3
6	3
7	4
8	4

Interpretación: como cada nodo tiene dos grados de libertad, habrán 8 ID. COORD (“x” local y “y” local)

Tabla 8:*Matrices de rigidez para cada barra de la cercha.*

NÚMERO DE BARRA		MATRIZ DE RIGIDEZ			
BARRA 1		365.2522	401.7774	-365.2522	-401.7774
		401.7774	441.9552	-401.7774	-441.9552
		-365.2522	-401.7774	365.2522	401.7774
		-401.7774	-441.9552	401.7774	441.9552
BARRA 2		365.2522	-401.7774	-365.2522	401.7774
		-401.7774	441.9552	401.7774	-441.9552
		-365.2522	401.7774	365.2522	-401.7774
		401.7774	-441.9552	-401.7774	441.9552
BARRA 3		0	0	0	0
		0	750	0	-750
		0	0	0	0
		0	-750	0	750
BARRA 4		702.9918	210.8975	-702.9918	-210.8975
		210.8975	63.2693	-210.8975	-63.2693
		-702.9918	-210.8975	702.9918	210.8975
		-210.8975	-63.2693	210.8975	63.2693
BARRA 5		702.9918	-210.8975	-702.9918	210.8975
		-210.8975	63.2693	210.8975	-63.2693
		-702.9918	210.8975	702.9918	-210.8975
		210.8975	-63.2693	-210.8975	63.2693

Tabla 9:*Matriz de rigidez global ensamblada para los GDLL*

$K_{global}(gdll)$				
	1	2	3	4
1	730.5044	0	0	0
2	0	1633.9	0	-750
3	0	0	1406	0
4	0	-750	0	876.5385

k) Módulo de resultados

Tabla 10:

Desplazamientos nodales de armadura en PAME UNC 2D

GDL_{LIBRES}	{U}
1	4.1068E-01
2	0
3	0
4	0

Tabla 11:

Resultados de desplazamientos nodales por eje

Nodo N°	Desplaz. X	Desplaz. Y
1	4.1068E-01	0
2	0	0
3	0	0
4	0	0

Tabla 12:

Resultados de fuerzas axiales para cada barra

Barra N°	Fuerza axial (Qa)
1	-2,2299E+02
2	2,2299E+02
3	0
4	0
5	0

Tabla 13:

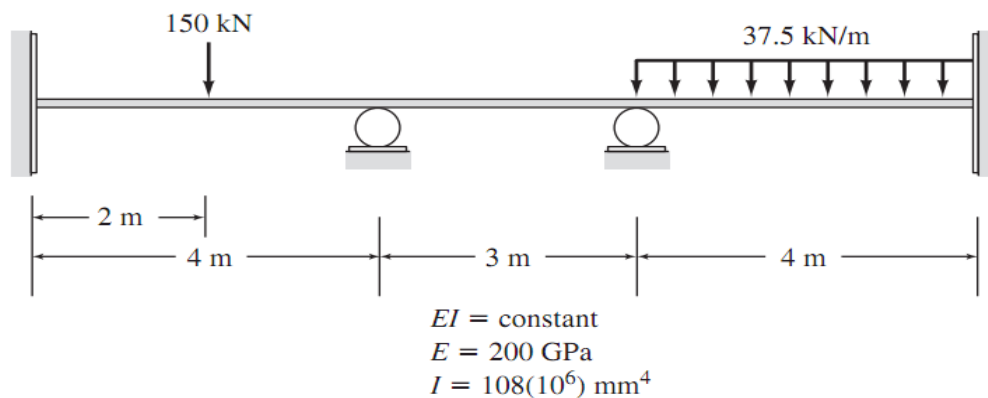
Resultados de reacciones en los apoyos

NODO N°	RX (KN)	RY (KN)
3	-1,5000E+02	-1,6500E+02
4	-1,5000E+02	1,6500E+02

CASO 2: VIGA

Se analizó el caso del ejercicio 5.6 tomado del libro *Matrix Analysis of Structures*, de Kassimali, A. (2022).

Enunciado: Determinar los desplazamientos nodales y reacciones en los apoyos (fuerza y momento) para la viga continua de tres tramos que se muestra en la Figura, utilizando el método de rigidez matricial.



Solución con PAME UNC 2D:

a) Archivo>>Nuevo>>“VIGA”>>VIGA_EJ_5_6_PAME_UNC_2D_VIORT_FINA
L.xlsx

Se establecen los nodos (5), barras (4) y los grados de libertad (15), empezando de izquierda a derecha, considerando tres grados de libertad, para el formato general.

b) **Propiedades geométricas**

$$\text{NÚMERO DE GRADOS DE LIBERTAD} = 5 * 3 = 15$$

Nota: Se han considerado 3 grados de libertad por cada nodo, para los casos en los que se tengan cargas horizontales o inclinadas en las barras, de no ser así, se pueden considerar solo dos grados de libertad (desplazamiento vertical y giro).

Figura 19:

grados de libertad de la viga

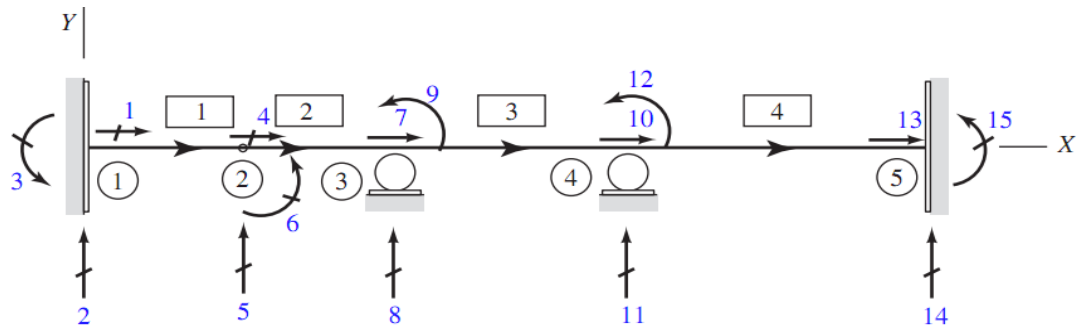


Tabla 14:

Ingreso de coordenadas nodales

Nodo	Abscisa (m)
1	0
2	2
3	4
4	7
5	11

Tabla 15:

Ingreso de geometría de las barras

Barra	Nodo Inicial	Nodo Final
1	1	2
2	2	3
3	3	4
4	4	5

c) Propiedades del material:

Tabla 16:

Propiedades de material para cada barra

PROPIEDAD	NÚMERO DE BARRA			
	BARRA 1	BARRA 2	BARRA 3	BARRA 4
E (KN/m ²)	200000000	200000000	200000000	200000000
I (m ⁴)	0.000108	0.000108	0.000108	0.000108
A (mm ²)	0	0	0	0
L (mm)	2	2	3	4

Nota: no se considera el área, pues la matriz de rigidez para este caso, solo depende de E, I, L.

d) Matrices de rigidez

Tabla 17:

Matrices de rigidez locales y globales

NÚMERO DE BARRA	K_LOCAL = K_GLOBAL					
	ux	uy	uz	ux	uy	uz
BARRA 1	0	0	0	0	0	0
	0	32400	32400	0	-32400	32400
	0	32400	43200	0	-32400	21600
	0	0	0	0	0	0
	0	-32400	-32400	0	32400	-32400
	0	32400	21600	0	-32400	43200
BARRA 2	0	0	0	0	0	0
	0	32400	32400	0	-32400	32400
	0	32400	43200	0	-32400	21600
	0	0	0	0	0	0
	0	-32400	-32400	0	32400	-32400
	0	32400	21600	0	-32400	43200
BARRA 3	0	0	0	0	0	0
	0	9600	14400	0	-9600	14400
	0	14400	28800	0	-14400	14400
	0	0	0	0	0	0
	0	-9600	-14400	0	9600	-14400
	0	14400	14400	0	-14400	28800
BARRA 4	0	0	0	0	0	0
	0	4050	8100	0	-4050	8100
	0	8100	21600	0	-8100	10800
	0	0	0	0	0	0
	0	-4050	-8100	0	4050	-8100
	0	8100	10800	0	-8100	21600

Nota: las matrices locales y globales coinciden, pues el ángulo de giro es nulo.

Tabla 18:*Matriz de rigidez ensamblada, caso de viga*

GDL	GDL													
	4	5	6	9	12	1	2	3	8	11	13	14	15	
4	0	0	0	0	0	0	0	0	0	0	0	0	0	
5	0	64800	0	32400	0	0	-32400	-32400	-32400	0	0	0	0	
6	0	0	86400	21600	0	0	32400	21600	-32400	0	0	0	0	
9	0	32400	21600	72000	14400	0	0	0	-18000	-14400	0	0	0	
12	0	0	0	14400	50400	0	0	0	14400	-6300	0	-8100	10800	
1	0	0	0	0	0	0	0	0	0	0	0	0	0	
2	0	-32400	32400	0	0	0	32400	32400	0	0	0	0	0	
3	0	-32400	21600	0	0	0	32400	43200	0	0	0	0	0	
8	0	-32400	-32400	-18000	14400	0	0	0	42000	-9600	0	0	0	
11	0	0	0	-14400	-6300	0	0	0	-9600	13650	0	-4050	8100	
13	0	0	0	0	0	0	0	0	0	0	0	0	0	
14	0	0	0	0	-8100	0	0	0	0	-4050	0	4050	-8100	
15	0	0	0	0	10800	0	0	0	0	8100	0	-8100	21600	

Tabla 19:*Matriz de rigidez: [KLL]*

SUBMATRIZ	GDLL			
	5	6	9	12
[KLL]=	64800	0	32400	0
	0	86400	21600	0
	32400	21600	72000	14400
	0	0	14400	50400

Tabla 20:*Inversa de [KLL]*

SUBMATRIZ	GDLL			
	5	6	9	12
[KLL] -1 =	64800	0	32400	0
	0	86400	21600	0
	32400	21600	72000	14400
	0	0	14400	50400

e) Cálculo de desplazamientos

$$\{F\} = [K]\{U\}$$

$$\begin{Bmatrix} F_L \\ - \\ F_R \end{Bmatrix} = \begin{bmatrix} K_{LL} & K_{LR} \\ K_{RL} & K_{RR} \end{bmatrix} \begin{Bmatrix} U_L \\ - \\ U_R \end{Bmatrix}$$

Como los desplazamientos en los grados de libertad restringidos son nulos, entonces la matriz equivalente está dada por:

$$\begin{Bmatrix} F_L \\ - \\ F_R \end{Bmatrix} = \begin{bmatrix} K_{LL} & K_{LR} \\ K_{RL} & K_{RR} \end{bmatrix} \begin{Bmatrix} U_L \\ - \\ 0 \end{Bmatrix}$$

Podemos transformar esta ecuación matricial, en dos nuevas ecuaciones:

$$\{F_L\} = [K_{LL}]\{U_L\} \dots (1)$$

$$\{F_R\} = [K_{RL}]\{U_L\} - \{F_{ext}\} \dots (2)$$

$$Si [A][X] = [B]$$

$$Entonces: [X] = [A]^{-1}[B]$$

Aplicando esta propiedad fundamental de álgebra matricial a la ecuación (1), obtenemos:

$$\{U_L\} = [K_{LL}]^{-1}\{F_L\}$$

Siendo:

$$\{F_L\} = \begin{Bmatrix} -150 \\ 0 \\ 0 \\ -50 \end{Bmatrix}$$

$$\{U_L\} = \begin{bmatrix} 64800 & 0 & 32400 & 0 \\ 0 & 86400 & 21600 & 0 \\ 32400 & 21600 & 72000 & 14400 \\ 0 & 0 & 14400 & 50400 \end{bmatrix} \begin{Bmatrix} -150 \\ 0 \\ 0 \\ -50 \end{Bmatrix} = \begin{Bmatrix} -0.003279 \\ -0.000482 \\ 0.001929 \\ -0.001543 \end{Bmatrix}$$

Obteniéndose la siguiente tabla:

Tabla 21:

Desplazamientos nodales en los GDLL de la viga

GDLL	Desplazamiento	Unidad
5	-0.003279	m
6	-0.000482	rad
9	0.001929	rad
12	-0.001543	rad

f) Cálculo de reacciones $\{F_R\}$:

Tabla 22:

Matriz $[K_{RL}]$

GDL - GDLL	GDLL			
	5	6	9	12
1	0	0	0	0
2	-32400	32400	0	0
3	-32400	21600	0	0
8	-32400	-32400	-18000	14400
11	0	0	-14400	-6300
13	0	0	0	0
14	0	0	0	-8100
15	0	0	0	10800

$$[KRL]\{U_L\} = \begin{Bmatrix} 0 \\ 90.625 \\ 95.83333 \\ 64.93056 \\ -18.05556 \\ 0 \\ 12.5 \\ -16.66667 \end{Bmatrix}$$

$$\{F_{ext}\} = \begin{Bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -75 \\ 0 \\ -75 \\ 50 \end{Bmatrix}$$

Usando: $\{F_R\} = [K_{RL}]\{U_L\} - \{F_{ext}\}$

$$\{F_{ext}\} = \begin{Bmatrix} 0 \\ 90.625 \\ 95.83333 \\ 64.93056 \\ 56.94444 \\ 0 \\ 87.5 \\ -66.66667 \end{Bmatrix}$$

Finalmente se tiene:

Tabla 23:

Reacciones en los GDLR de la viga

GDLR	REACCIONES	Unidades
2	90.625	KN
3	95.8333	KN-m
8	64.931	KN
11	56.944	KN
14	87.5	KN
15	-66.6667	KN-m

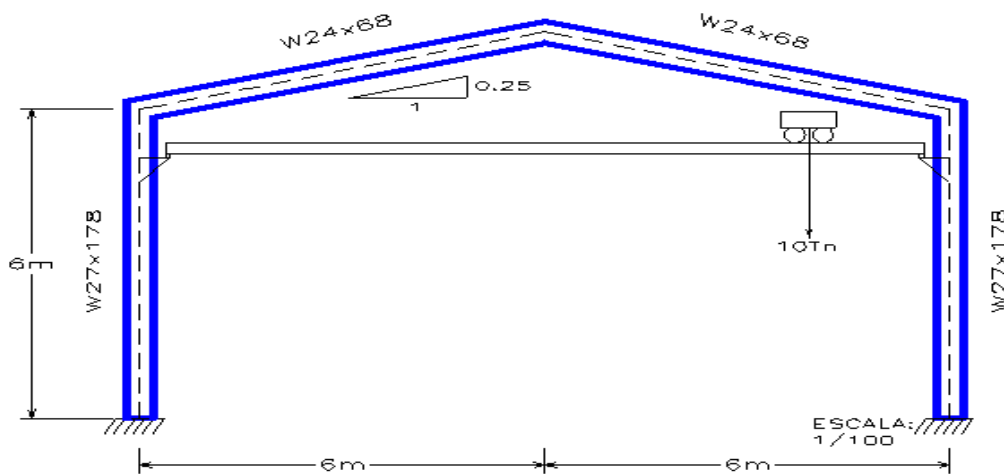
CASO 3: PÓRTICO

Se analiza un caso práctico para pórticos, adaptado por el tesista:

Enunciado: La estructura mostrada es un pórtico que soporta un puente grúa con efectos de anclajes despreciable. En un instante $\Delta t = 0.5 \text{ seg}$, el polipasto impacta con el extremo final de la viga puente. Calcular los **desplazamientos y reacciones que genera el impacto del polipasto en la estructura**. La velocidad del polipasto al momento del impacto es de 10 km/h. El peso del polipasto es de 10 Tn. La estructura del pórtico es a base de acero estructural ASTM A36 ($E = 20\,000\,000 \text{ Tn/m}^2$). Las columnas del pórtico son perfiles laminados W27x178 y las vigas W24x68. Desprecie el peso de la viga puente. Para el cálculo del impacto puede emplear la expresión del impulso en función de la cantidad de movimiento: $\Delta t \cdot F = m \cdot V$, donde Δt : instante de tiempo; **F**: fuerza de impacto; **m**: masa; **V**: velocidad. Use $g = 9.81 \text{ m/s}^2$.

Figura 20:

CASO3. Ejercicio para pórticos



SOLUCIÓN:

Tabla 24:

Propiedades de secciones^(*):

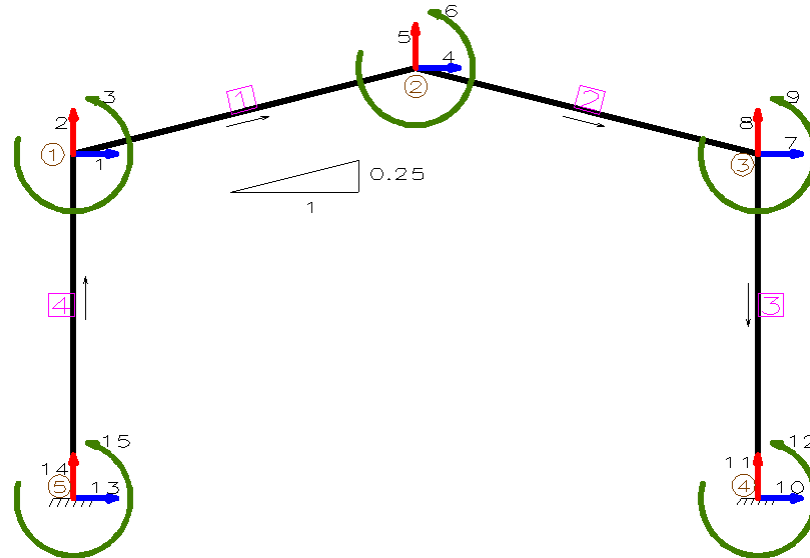
TIPO DE SECCIÓN	A (in ²)	I (in ⁴)	A (m ²)	I (m ⁴)
W24x68	20.1	1830	0.012968	0.0007617
W27x178	52.3	6990	0.033742	0.0029095

(*)Se obtiene de tablas, o importando los valores desde SAP 2000, plantilla "AISC PRO"

DEFINE >> SECTION PROPERTIES >> FRAME SECTION>>IMPORT NEW PROPERTY >> STEEL, I/WIDE FLANGE >> AISC.PRO (UBICADA EN CARPETA DE INSTALACIÓN DEL PROGRAMA) >> & SELECCIONAR

EL PERFIL DESEADO (W24X68, W27X178). LUEGO SE OBSERVAN SUS PROPIEDADES (A, I) TENIENDO EN CUENTA LAS UNIDADES PREFIJADAS EN EL PROGRAMA.

1) ENUMERACIÓN DE BARRAS, NUDOS Y GRADOS DE LIBERTAD



2) CÁLCULO DE LA FUERZA DE IMPACTO

$$\Delta t \cdot F = m \cdot V$$

$$(0.5s) \times F = \left(\frac{10 \text{ tf}}{9.81 \frac{m}{s^2}} \right) \times \left(10 \frac{km}{h} \right)$$

Realizando conversiones idóneas:

$$(0.5s) \times F = \left(\frac{10 \text{ tf}}{9.81 \frac{m}{s^2}} \times \frac{9.81 \times 10^3 N}{1 \text{ tf}} \times \frac{1 kg \times \frac{m}{s^2}}{1 N} \right) \times \left(10 \frac{km}{h} \times \frac{1 h}{3600 s} \times \frac{1000 m}{1 km} \right)$$

$$(0.5s) \times F = \left(\frac{10 \boxed{tf}}{9.81 \boxed{\frac{m}{s^2}}} \times \frac{9.81 \times 10^3 \boxed{N}}{1 \boxed{tf}} \times \frac{1 kg \times \boxed{\frac{m}{s^2}}}{1 \boxed{N}} \right) \times \left(10 \frac{\boxed{km}}{\boxed{h}} \times \frac{1 \boxed{h}}{3600 s} \times \frac{1000 m}{1 \boxed{km}} \right)$$

$$\rightarrow F = \frac{\left(\frac{9.81 \times 10^4}{9.81} kg \right) \times \left(\frac{10 \times 1000 m}{3600 s} \right)}{0.5 s}$$

$$\rightarrow F = \frac{10^8}{1800} kg \times \frac{m}{s^2}$$

$$\rightarrow F = \frac{10^6}{18} N$$

$$\rightarrow F = \frac{10^6}{18} \boxed{N} \times \frac{1tf}{9.81 \times 10^3 \boxed{N}}$$

$$\therefore F = \frac{10^3}{18 \times 9.81} tf \approx 5.66tf$$

3) CARACTERÍSTICAS DE LAS BARRAS:

Tabla 25:

Propiedades de las barras que conforman el pórtico

Barra	Area	M. de inercia	Mod. Elasticidad	Longitud	Angulo entre x y X	
	A (m ²)	I (m ⁴)	E (Tn/m ²)	L (m)	α (°)	α (rad)
1	0.012968	0.000762	20,000,000	6.184658	-14.0362	-0.24498
2	0.012968	0.000762	20,000,000	6.184658	14.03624	0.244979
3	0.033742	0.002909	20,000,000	6	90	1.570796
4	0.033742	0.002909	20,000,000	6	270	4.712389

4) INGRESO DE INFORMACIÓN EN PAME UNC 2D → VIORTI

Vamos a: Archivo>>Nuevo>>PÓRTICO>>hoja interactiva VIORTI, o presionamos directamente la combinación de teclas “ctrl”+”N”

5) MATRIZ DE RIGIDEZ GENERALIZADA EN COORDENADAS LOCALES:

$$[k] = \begin{bmatrix} \frac{EA}{L} & 0 & 0 & -\frac{EA}{L} & 0 & 0 \\ 0 & \frac{12EI}{L^3} & \frac{6EI}{L^2} & 0 & -\frac{12EI}{L^3} & \frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{4EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{2EI}{L} \\ -\frac{EA}{L} & 0 & 0 & \frac{EA}{L} & 0 & 0 \\ 0 & -\frac{12EI}{L^3} & -\frac{6EI}{L^2} & 0 & \frac{12EI}{L^3} & -\frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{2EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{4EI}{L} \end{bmatrix}$$

5.1) MATRICES DE RIGIDEZ EN COORDENADAS LOCALES:

	1	2	3	4	5	6	
$[k_1] =$	41935.11	0	0	-41935.11	0	0	1
	0	772.77	2389.66	0	-772.77	2389.66	2
	0	2389.66	9852.81	0	-2389.66	4926.41	3
	-41935.11	0	0	41935.11	0	0	4
	0	-772.77	-2389.66	0	772.77	-2389.66	5
	0	2389.66	4926.41	0	-2389.66	9852.81	6

	4	5	6	7	8	9	
$[k_2] =$	41935.11	0	0	-41935.11	0	0	4
	0	772.77	2389.66	0	-772.77	2389.66	5
	0	2389.66	9852.81	0	-2389.66	4926.41	6
	-41935.11	0	0	41935.11	0	0	7
	0	-772.77	-2389.66	0	772.77	-2389.66	8
	0	2389.66	4926.41	0	-2389.66	9852.81	9

	7	8	9	10	11	12	
$[k_3] =$	112472.89	0	0	-112472.89	0	0	7
	0	3232.73	9698.19	0	-3232.73	9698.19	8
	0	9698.19	38792.77	0	-9698.19	19396.38	9
	-112472.89	0	0	112472.89	0	0	10
	0	-3232.73	-9698.19	0	3232.73	-9698.19	11
	0	9698.19	19396.38	0	-9698.19	38792.77	12

	13	14	15	1	2	3	
$[k_4] =$	112472.89	0	0	-112472.89	0	0	13
	0	3232.73	9698.19	0	-3232.73	9698.19	14
	0	9698.19	38792.77	0	-9698.19	19396.38	15
	-112472.89	0	0	112472.89	0	0	1
	0	-3232.73	-9698.19	0	3232.73	-9698.19	2
	0	9698.19	19396.38	0	-9698.19	38792.77	3

6) MATRIZ DE TRANSFORMACIÓN GENERALIZADA:

$$[T] = \begin{bmatrix} \cos(\alpha) & \text{sen}(\alpha) & 0 & 0 & 0 & 0 \\ -\text{sen}(\alpha) & \cos(\alpha) & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos(\alpha) & \text{sen}(\alpha) & 0 \\ 0 & 0 & 0 & -\text{sen}(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

6.1) MATRICES DE TRANSFORMACIÓN INDIVIDUAL:

		Ángulo: -14.0362°		Seno: -0.2425		Coseno: 0.9701	
[T ₁] =	0.9701	-0.2425	0	0	0	0	
	0.2425	0.9701	0	0	0	0	
	0	0	1	0	0	0	
	0	0	0	0.9701	-0.2425	0	
	0	0	0	0.2425	0.9701	0	
	0	0	0	0	0	1	

		Ángulo: 14.0362°		Seno: 0.2425		Coseno: 0.9701	
[T ₂] =	0.9701	0.2425	0	0	0	0	
	-0.2425	0.9701	0	0	0	0	
	0	0	1	0	0	0	
	0	0	0	0.9701	0.2425	0	
	0	0	0	-0.2425	0.9701	0	
	0	0	0	0	0	1	

		Ángulo: 90°		Seno: 1		Coseno: 0	
[T ₃] =	0	1	0	0	0	0	0
	-1	0	0	0	0	0	0
	0	0	1	0	0	0	0
	0	0	0	0	1	0	0
	0	0	0	-1	0	0	0
	0	0	0	0	0	0	1

		Ángulo: 270°		Seno: -1		Coseno: 0	
[T ₄] =	0	-1	0	0	0	0	0
	1	0	0	0	0	0	0
	0	0	1	0	0	0	0
	0	0	0	0	-1	0	0
	0	0	0	1	0	0	0
	0	0	0	0	0	0	1

7) TRANSPUESTA DE MATRIZ DE TRANSFORMACIÓN GENERALIZADA:

$$T^T = \begin{bmatrix} \cos(\alpha) & -\text{sen}(\alpha) & 0 & 0 & 0 & 0 \\ \text{sen}(\alpha) & \cos(\alpha) & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos(\alpha) & -\text{sen}(\alpha) & 0 \\ 0 & 0 & 0 & \text{sen}(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Tabla 26:

Transpuestas de matrices de transformación para pórtico

$[T_1]^T =$	0.9701	0.2425	0	0	0	0
	-0.2425	0.9701	0	0	0	0
	0	0	1	0	0	0
	0	0	0	0.9701	0.2425	0
	0	0	0	-0.2425	0.9701	0
	0	0	0	0	0	1
$[T_2]^T =$	0.9701	-0.2425	0	0	0	0
	0.2425	0.9701	0	0	0	0
	0	0	1	0	0	0
	0	0	0	0.9701	-0.2425	0
	0	0	0	0.2425	0.9701	0
	0	0	0	0	0	1
$[T_3]^T =$	0	-1	0	0	0	0
	1	0	0	0	0	0
	0	0	1	0	0	0
	0	0	0	0	-1	0
	0	0	0	1	0	0
	0	0	0	0	0	1
$[T_4]^T =$	0	1	0	0	0	0
	-1	0	0	0	0	0
	0	0	1	0	0	0
	0	0	0	0	1	0
	0	0	0	-1	0	0
	0	0	0	0	0	1

8) MATRIZ DE RIGIDEZ EN COORDENADAS GLOBALES

$$[K] = [T][k][T]^T$$

8.1) MATRIZ DE RIGIDEZ EN COORDENADAS GLOBALES PARA CADA BARRA:

	1	2	3	4	5	6	
$[K_1] =$	39513.79	9685.26	-579.58	-39513.79	-9685.26	-579.58	1
	9685.26	3194.08	2318.31	-9685.26	-3194.08	2318.31	2
	-579.58	2318.31	9852.81	579.58	-2318.31	4926.41	3
	-39513.79	-9685.26	579.58	39513.79	9685.26	579.58	4
	-9685.26	-3194.08	-2318.31	9685.26	3194.08	-2318.31	5
	-579.58	2318.31	4926.41	579.58	-2318.31	9852.81	6

	4	5	6	7	8	9	
$[K_2] =$	39513.79	-9685.26	579.58	-39513.79	9685.26	579.58	4
	-9685.26	3194.08	2318.31	9685.26	-3194.08	2318.31	5
	579.58	2318.31	9852.81	-579.58	-2318.31	4926.41	6
	-39513.79	9685.26	-579.58	39513.79	-9685.26	-579.58	7
	9685.26	-3194.08	-2318.31	-9685.26	3194.08	-2318.31	8
	579.58	2318.31	4926.41	-579.58	-2318.31	9852.81	9

	7	8	9	10	11	12	
$[K_3] =$	3232.73	0	9698.19	-3232.73	0	9698.19	7
	0	112472.89	0	0	-112472.89	0	8
	9698.19	0	38792.77	-9698.19	0	19396.38	9
	-3232.73	0	-9698.19	3232.73	0	-9698.19	10
	0	-112472.89	0	0	112472.89	0	11
	9698.19	0	19396.38	-9698.19	0	38792.77	12

	13	14	15	1	2	3	
$[K_4] =$	3232.73	0	-9698.19	-3232.73	0	-9698.19	13
	0	112472.89	0	0	-112472.89	0	14
	-9698.19	0	38792.77	9698.19	0	19396.38	15
	-3232.73	0	9698.19	3232.73	0	9698.19	1
	0	-112472.89	0	0	112472.89	0	2
	-9698.19	0	19396.38	9698.19	0	38792.77	3

9) ENSAMBLAJE DE MATRIZ EN COORDENADAS GLOBALES

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	42746.52	9685.26	9118.62	-39513.79	-9685.26	-579.58	0	0	0	0	0	0	-3232.73	0	9698.19
2	9685.26	115666.98	2318.31	-9685.26	-3194.08	2318.31	0	0	0	0	0	0	0	-112472.89	0
3	9118.62	2318.31	48645.58	579.58	-2318.31	4926.41	0	0	0	0	0	0	-9698.19	0	19396.38
4	-39513.79	-9685.26	579.58	79027.59	0	1159.15	-39513.79	9685.26	579.58	0	0	0	0	0	0
5	-9685.26	-3194.08	-2318.31	0	6388.17	0	9685.26	-3194.08	2318.31	0	0	0	0	0	0
6	-579.58	2318.31	4926.41	1159.15	0	19705.63	-579.58	-2318.31	4926.41	0	0	0	0	0	0
7	0	0	0	-39513.79	9685.26	-579.58	42746.52	-9685.26	9118.62	-3232.73	0	9698.19	0	0	0
8	0	0	0	9685.26	-3194.08	-2318.31	-9685.26	115666.98	-2318.31	0	-112472.89	0	0	0	0
9	0	0	0	579.58	2318.31	4926.41	9118.62	-2318.31	48645.58	-9698.19	0	19396.38	0	0	0
10	0	0	0	0	0	0	-3232.73	0	-9698.19	3232.73	0	-9698.19	0	0	0
11	0	0	0	0	0	0	0	-112472.89	0	0	112472.89	0	0	0	0
12	0	0	0	0	0	0	9698.19	0	19396.38	-9698.19	0	38792.77	0	0	0
13	-3232.73	0	-9698.19	0	0	0	0	0	0	0	0	0	3232.73	0	-9698.19
14	0	-112472.89	0	0	0	0	0	0	0	0	0	0	0	112472.89	0
15	9698.19	0	19396.38	0	0	0	0	0	0	0	0	0	-9698.19	0	38792.77

De las bases teóricas, se sabe que, para poder determinar el vector de desplazamientos, se necesita de $[K_{LL}]^{-1}$ y $\{F_L\}$

Tabla 27:

Inversa de la sub matriz de rigidez $[K_{LL}]$

	1	2	3	4	5	6	7	8	9	
	0.000455889	9.60067E-07	-8.95362E-05	0.000419138	0.000112904	4.36534E-05	0.000380619	-9.60077E-07	-8.61881E-05	1
	9.60067E-07	8.88643E-06	-3.20025E-07	2.06722E-06	4.44549E-06	-9.50035E-07	9.60077E-07	4.59876E-09	-3.20025E-07	2
	-8.95362E-05	-3.20025E-07	4.00086E-05	-8.81567E-05	2.70551E-06	-1.45511E-05	-8.61881E-05	3.20025E-07	1.85662E-05	3
	0.000419138	2.06722E-06	-8.81567E-05	0.000432964	-6.64185E-19	4.27789E-05	0.000419138	-2.06722E-06	-8.81567E-05	4
$[K_{LL}]^{-1} =$	0.000112904	4.44549E-06	2.70551E-06	-7.2652E-19	0.000505301	-9.0815E-20	-0.000112904	4.44549E-06	-2.70551E-06	5
	4.36534E-05	-9.50035E-07	-1.45511E-05	4.27789E-05	-6.72035E-20	5.82975E-05	4.36534E-05	9.50035E-07	-1.45511E-05	6
	0.000380619	9.60077E-07	-8.61881E-05	0.000419138	-0.000112904	4.36534E-05	0.000455889	-9.60067E-07	-8.95362E-05	7
	-9.60077E-07	4.59876E-09	3.20025E-07	-2.06722E-06	4.44549E-06	9.50035E-07	-9.60067E-07	8.88643E-06	3.20025E-07	8
	-8.61881E-05	-3.20025E-07	1.85662E-05	-8.81567E-05	-2.70551E-06	-1.45511E-05	-8.95362E-05	3.20025E-07	4.00086E-05	9

Vector de fuerzas:

$$\{F_L\} = \begin{Bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 5.66 \text{ tf} \\ 0 \\ 0 \\ 0 \\ 0 \end{Bmatrix} \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \\ 12 \\ 15 \end{matrix}$$

Multiplicando estos arreglos matriciales, se obtienen los desplazamientos:

Tabla 28:

Desplazamientos $\{U_L\}$ del pórtico en los GDLL.

GDLL	DESPLAZAMIENTOS	UNIDADES
1	0.002154	m
2	5.434E-06	m
3	-0.000488	rad
4	0.002372	m
5	-0.000639	m
6	0.000247	rad
7	0.00258	m
8	-5.434E-06	m
9	-0.00051	rad

Con los desplazamientos obtenidos, y haciendo uso de la ecuación (2) de las bases teóricas, se obtienen las fuerzas en los grados de libertad restringidos (reacciones).

10) CÁLCULO DE LAS REACCIONES $\{F_R\}$:

Tabla 29:

Sub matriz $[KRL]$

	1	2	3	4	5	6	7	8	9	
$[KRL] =$	0	0	0	0	0	0	-3232.73	6.69E-12	-9698.192	10
	0	0	0	0	0	0	0.00	-1.12E+05	-5.94E-13	11
	0	0	0	0	0	0	9698.19	5.94E-13	19396.384	12
	-3232.731	2.00753E-11	-9698.192	0	0	0	0.00	0	0	13
	2.008E-11	-112472.893	-1.78E-12	0	0	0	0.00	0	0	14
	9698.1922	1.78226E-12	19396.384	0	0	0	0.00	0	0	15

Multiplicando esta matriz, con la matriz de desplazamientos del paso anterior, se obtiene la matriz de fuerzas en los G.L.R (reacciones en apoyos):

Tabla 30:

Reacciones $\{FR\}$ en los GDLR.

GDLR	REACCIONES	UNIDADES
10	-3.4267	Tonf
11	0.6112	Tonf
12	15.19497	Tonf-m
13	-2.2333	Tonf
14	-0.6112	Tonf
15	11.43083	Tonf-m

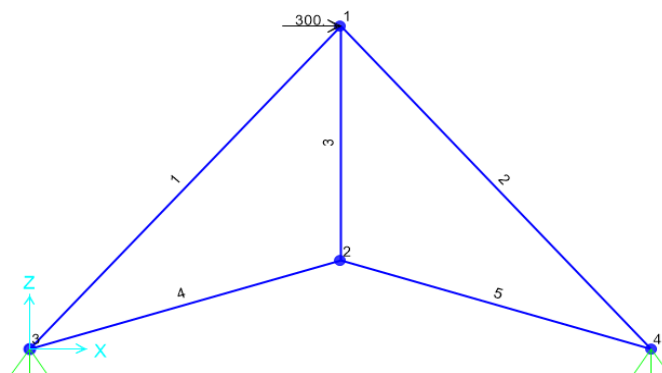
3.7.2. DESARROLLO DE CASOS EN SAP2000 V23.3.1

CASO 1: ARMADURA

Se crea el modelo de la armadura, considerando que los momentos deben ser liberados, recordando que la respuesta estructural depende del módulo de elasticidad (E), área de la sección transversal (A), y longitud de las barras (L).

Figura 21:

Geometría y carga nodal aplicada en la armadura, en SAP2000 V23.3.1.



Una vez ingresada la geometría de la armadura, y las propiedades de los materiales, así como la asignación de carga, se procede a ejecutar el modelo y revisar los resultados.

Tabla 31:

Desplazamientos nodales de la armadura en SAP2000 V23.3.1

Joint Displacements					
Joint Text	OutputCase Text	CaseType Text	U1 mm	U2 mm	U3 mm
1	CARGA1	LinStatic	0.41068	0	0
2	CARGA1	LinStatic	0	0	0
3	CARGA1	LinStatic	0	0	0
4	CARGA1	LinStatic	0	0	0

Tabla 32:

Reacciones en los apoyos de la armadura según SAP2000 V23.3.1

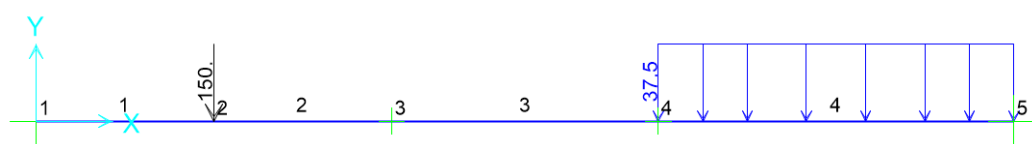
Joint Reactions					
Joint Text	OutputCase Text	CaseType Text	F1 KN	F2 KN	F3 KN
3	CARGA1	LinStatic	-150	0	-165
4	CARGA1	LinStatic	-150	0	165

CASO 2: VIGA

Se crea el modelo de viga, considerando que los ejes principales deben ser “X” e “Y”, lo que trae por consiguiente que el eje “Z” quedará apuntando hacia afuera del documento (consideración de la regla de la mano derecha o su equivalente formal: sistema cartesiano ortogonal), pues de haber conservado los ejes “X” y “Z” como principales, los momentos tendrían el signo cambiado.

Figura 22:

Geometría y cargas asignadas a viga en SAP2000 V23.3.1.



Una vez ingresada la geometría de la viga, y las propiedades de los materiales, así como la asignación de cargas, se procede a ejecutar el modelo y revisar los resultados.

Tabla 33:

Desplazamientos nodales de viga en SAP2000 V23.3.1

Joint Displacements								
Joint Text	OutputCase Text	CaseType Text	U1 m	U2 m	U3 m	R1 Radians	R2 Radians	R3 Radians
1	CARGAS_V	LinStatic	0	0	0	0	0	0
2	CARGAS_V	LinStatic	0	-0.003279	0	0	0	-0.000482
3	CARGAS_V	LinStatic	0	0	0	0	0	0.001929
4	CARGAS_V	LinStatic	0	0	0	0	0	-0.001543
5	CARGAS_V	LinStatic	0	0	0	0	0	0

Tabla 34:

Reacciones en los apoyos de la viga según SAP2000 V23.3.1

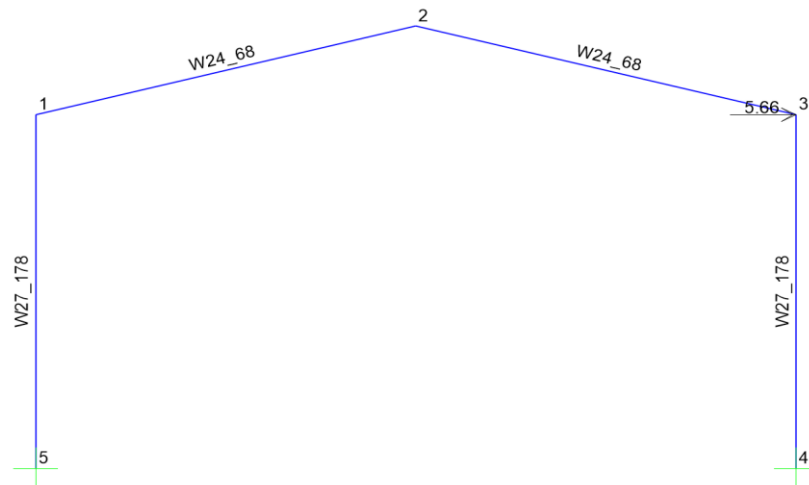
Joint Reactions								
Joint Text	OutputCase Text	CaseType Text	F1 KN	F2 KN	F3 KN	M1 KN-m	M2 KN-m	M3 KN-m
1	CARGAS_V	LinStatic	0	90.625	0	0	0	95.8333
3	CARGAS_V	LinStatic	0	64.931	0	0	0	0
4	CARGAS_V	LinStatic	0	56.944	0	0	0	0
5	CARGAS_V	LinStatic	0	87.5	0	0	0	-66.6667

CASO 3: PÓRTICO

Se crea el modelo de pórtico, considerando los parámetros de Módulo de elasticidad (E), momento de inercia (I), área de la sección transversal (A), longitudes de las barras (L).

Figura 23:

Geometría y carga del pórtico en SAP2000 V23.3.1



Una vez ingresada la geometría del pórtico, y las propiedades de los materiales, así como la asignación de la carga, se procede a ejecutar el modelo y revisar los resultados.

Tabla 35:

Desplazamientos nodales del pórtico en SAP2000 V23.3.1

Joint Displacements					
Joint Text	OutputCase Text	CaseType Text	U1 m	U2 m	R3 Radians
1	PORTI_CARGA	LinStatic	0.002154	0.000005434	-0.000488
2	PORTI_CARGA	LinStatic	0.002372	-0.000639	0.000247
3	PORTI_CARGA	LinStatic	0.00258	-0.000005434	-0.000507
4	PORTI_CARGA	LinStatic	0	0	0
5	PORTI_CARGA	LinStatic	0	0	0

Tabla 36:

Reacciones en los apoyos del pórtico según SAP2000 V23.3.1

TABLE: Joint Reactions					
Joint Text	OutputCase Text	CaseType Text	F1 Tonf	F2 Tonf	M3 Tonf-m
4	PORTI_CARGA	LinStatic	-3.4267	0.6112	15.19497
5	PORTI_CARGA	LinStatic	-2.2333	-0.6112	11.43083

3.8. TÉCNICAS E INSTRUMENTOS DE RECOLECCIÓN DE INFORMACIÓN

3.8.1. Técnicas

- Adaptación y diseño de modelos estructurales: Se utilizaron ejemplos estructurales extraídos de textos académicos y se desarrolló un modelo adicional adaptado por el tesista. Esta técnica permite la replicación de casos de estudio y su aplicación en el software de análisis estructural.
- Recopilación sistematizada de ejemplos: Los ejemplos analizados provienen de libros especializados como "Matrix Structural Analysis" de McGuire et al. (2015) y "Matrix Analysis of Structures" de Kassimali (2022). La selección de casos sigue criterios de representatividad para evaluar estructuras de cerchas, vigas y pórticos.

3.8.2. Instrumentos

- Software MATLAB R2022a: Utilizado en su entorno App Designer para el desarrollo del software PAME UNC 2D. Este lenguaje de programación permitió modelar, procesar y analizar las estructuras con base en el método matricial de rigidez.
- Software SAP2000 V23.3.1: Programa comercial ampliamente validado y utilizado como referencia para comparar los resultados estructurales obtenidos.
- Equipo de cómputo: Se empleó una computadora con especificaciones adecuadas para la ejecución de simulaciones y procesamiento de datos.

CAPÍTULO IV. ANÁLISIS Y DISCUSIÓN DE RESULTADOS

4.1. ANÁLISIS DE RESULTADOS

De acuerdo con los objetivos trazados en esta investigación, se muestran los resultados de la respuesta estructural analizada con PAME UNC 2D y SAP2000 V23.3.1, así como la variación porcentual de los resultados obtenidos en ambos programas, a través del cálculo del Error Porcentual Absoluto Medio (MAPE).

4.1.1. CASO DE ARMADURA:

Tabla 37:

Comparación de desplazamientos de armadura.

NODO	PAME UNC 2D	SAP2000 V23.3.1	VARIACIÓN PORCENTUAL
	UX (mm)	UX (mm)	
1	0.41068	0.41068	0%
2	0	0	0%
MAPE = 0%			

Tabla 38:

Comparación de reacciones en eje "X", para armadura.

NODO	PAME UNC 2D	SAP2000 V23.3.1	VARIACIÓN PORCENTUAL
	RX (kN)	RX (kN)	
3	-1.50E+02	-150	0%
4	-1.50E+02	-150	0%
MAPE = 0%			

Tabla 39:

Comparación de reacciones en el eje "Y", para la armadura.

NODO	PAME UNC 2D	SAP2000 V23.3.1	VARIACIÓN PORCENTUAL
	RY (kN)	RY (kN)	
3	-1.65E+02	-165	0%
4	1.65E+02	165	0%
MAPE = 0%			

4.1.2. CASO DE VIGA:

Tabla 40:

Comparación de los desplazamientos verticales de la viga.

NODO	U _y (m)		VARIACIÓN PORCENTUAL
	PAME UNC 2D	SAP2000 V23.3.1	
1	0	0	0%
2	-0.003279	-0.003279	0%
3	0	0	0%
4	0	0	0%
5	0	0	0%
MAPE = 0%			

Tabla 41:

Comparación de los giros nodales de la viga.

NODO	RZ (rad)		VARIACIÓN PORCENTUAL
	PAME UNC 2D	SAP2000 V23.3.1	
1	0	0	0%
2	-0.000482	-0.000482	0%
3	0.001929	0.001929	0%
4	-0.001543	-0.001543	0%
5	0	0	0%
MAPE = 0%			

Tabla 42:

Comparación de las reacciones en viga.

NODO	F2 (kN)		VARIACIÓN PORCENTUAL	M3 (kN-m)		VARIACIÓN PORCENTUAL
	PAME UNC 2D	SAP2000 V23.3.1		PAME UNC 2D	SAP2000 V23.3.1	
1	90.625	90.625	0%	95.833	95.833	0%
3	64.931	64.931	0%	0	0	0%
4	56.944	56.944	0%	0	0	0%
5	87.5	87.5	0%	-66.667	-66.667	0%
MAPE = 0%						

4.1.3. CASO DEL PÓRTICO:

Tabla 43:

Comparación de los desplazamientos horizontales en el pórtico.

Nodo (GDLL)	U ₁ (m)		VARIACIÓN PORCENTUAL
	PAME UNC 2D	SAP2000 V23.3.1	
1 (1)	0.002154	0.002154	0%
2 (4)	0.002372	0.002372	0%
3 (7)	0.00258	0.00258	0%
			MAPE = 0%

Tabla 44:

Comparación de los desplazamientos verticales en el pórtico.

Nodo (GDLL)	U ₂ (m)		VARIACIÓN PORCENTUAL
	PAME UNC 2D	SAP2000 V23.3.1	
1 (2)	5.434E-06	5.434E-06	0%
2 (5)	-0.000639	-0.000639	0%
3 (8)	-5.434E-06	-5.434E-06	0%
			MAPE = 0%

Tabla 45:

Comparación de las rotaciones nodales en el pórtico.

Nodo (GDLL)	R ₃ (rad)		VARIACIÓN PORCENTUAL
	PAME UNC 2D	SAP2000 V23.3.1	
1 (3)	-0.000488	-0.000488	0%
2 (6)	0.000247	0.000247	0%
3 (9)	-0.000507	-0.000507	0%
			MAPE = 0%

Tabla 46:

Comparación de las reacciones en el eje "X" para el pórtico.

NODO (GDLR)	RX (Tonf)		VARIACIÓN PORCENTUAL
	PAME UNC 2D	SAP2000 V23.3.1	
4 (10)	-3.4267	-3.4267	0%
5 (13)	-2.2333	-2.2333	0%
			MAPE = 0%

Tabla 47:*Comparación de las reacciones en el eje "Y" para el pórtico.*

NODO (GDLR)	RY (Tonf)		VARIACIÓN PORCENTUAL
	PAME UNC 2D	SAP2000 V23.3.1	
4 (11)	0.6112	0.6112	0%
5 (14)	-0.6112	-0.6112	0%
			MAPE = 0%

Tabla 48:*Comparación de los momentos alrededor de "Z" para el pórtico.*

NODO (GDLR)	M3 (Tonf-m)		VARIACIÓN PORCENTUAL
	PAME UNC 2D	SAP2000 V23.3.1	
4 (12)	15.19497	15.19497	0%
5 (15)	11.43083	11.43083	0%
			MAPE = 0%

4.2. DISCUSIÓN DE RESULTADOS

Al realizar la comparación de la respuesta estructural de los modelos bidimensionales analizados, considerando los parámetros de reacciones en los apoyos, desplazamientos nodales y momentos flectores, según el tipo de estructura, por medio de los programas PAME UNC 2D y SAP2000 V23.3.1, se evidencia que la variación porcentual para los tres casos: armadura, viga y pórtico, es del orden de 0%, lo cual confirma la concordancia total de los resultados obtenidos en ambos programas.

En este caso, la verificación se realizó aplicando el Error Porcentual Absoluto Medio (MAPE), que arrojó valores iguales a cero en todos los ítems evaluados, lo que representa una evidencia estadística sólida.

4.3. CORROBORACIÓN DE HIPÓTESIS

De acuerdo con el enfoque cuantitativo adoptado, la hipótesis formulada en el capítulo I ha sido sometida a prueba mediante el análisis comparativo de la respuesta estructural obtenida con los programas PAME UNC 2D y SAP2000 V23.3.1, bajo condiciones de modelado equivalentes. La hipótesis planteada fue:

“La variación porcentual promedio en la respuesta estructural obtenida mediante el software educativo PAME UNC 2D es inferior al 5% respecto a la respuesta generada por el software SAP2000 V23.3.1, para estructuras bidimensionales como armaduras, vigas y pórticos.”

Los resultados obtenidos muestran que las variaciones porcentuales fueron 0%, tanto en desplazamientos nodales, como en reacciones de apoyo y momentos flectores, según el tipo de estructura. De modo que la concordancia fue total en todos los parámetros evaluados y por tanto la hipótesis queda plenamente corroborada

CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

Finalizada la investigación, cuyo objetivo principal fue comparar la respuesta estructural obtenida mediante el software educativo PAME UNC 2D con aquella generada por el software comercial SAP2000 V23.3.1, en el análisis de estructuras bidimensionales, y con base en los objetivos planteados y los resultados obtenidos, se concluye lo siguiente:

- Se logró comparar la respuesta estructural usando PAME UNC 2D y SAP2000 v23.3.1, para los casos de una armadura, viga y pórtico, sometidos a cargas estáticas.
- Se consiguió modelar correctamente en ambos entornos, con parámetros idénticos de geometría, propiedades mecánicas, apoyos y cargas. Esta equivalencia metodológica permitió establecer una base objetiva para la comparación estructural.
- Se consiguió que los valores de desplazamientos nodales, reacciones en los apoyos y momentos flectores, según el tipo de estructura, fueran idénticos en ambos programas, lo que evidencia que el método matricial de rigidez implementado en PAME UNC 2D produce resultados estructuralmente consistentes respecto a SAP2000 V23.3.1.
- El uso del Error Porcentual Absoluto Medio (MAPE) como métrica de comparación arrojó un valor de 0% en todos los modelos, consolidando la evidencia cuantitativa de que no existe discrepancia entre los resultados de PAME UNC 2D y SAP2000 v23.3.1 en el rango elástico lineal estático.

5.2. RECOMENDACIONES

A partir de los resultados obtenidos en la presente investigación, se formulan las siguientes recomendaciones para futuras investigaciones:

— Se recomienda ampliar el alcance de la investigación, incorporando condiciones más complejas, tales como apoyos elásticos, apoyos rotados, o combinaciones de desplazamientos restringidos parcialmente, a fin de evaluar con mayor precisión el comportamiento comparativo de los programas frente a escenarios más representativos de la realidad estructural.

— Se sugiere considerar en futuros análisis de estructuras: geometría variable y la incorporación de efectos térmicos, que puedan afectar la distribución de desplazamientos y esfuerzos, permitiendo así un análisis comparativo más amplio en la práctica profesional.

— Finalmente, se propone fortalecer el proceso de validación de los resultados a través del incremento en la variedad de configuraciones estructurales estudiadas, con más número de elementos y condiciones de carga, lo cual permitiría robustecer las conclusiones sobre la precisión del software educativo en relación al software comercial.

REFERENCIAS BIBLIOGRÁFICAS

- Atalaya, M. (2019). *Comparación del comportamiento estructural de vigas, pórticos y armaduras usando un programa elaborado en LUA respecto a SAP 2000* [Universidad Nacional de Cajamarca].
<https://repositorio.unc.edu.pe/handle/20.500.14074/2864>
- Cervera, M. et al. (2015). *Análisis matricial de estructuras*. CIMNE.
- Chilón T. et al. (2023). *Impacto del SAP y MASTAN2-Matlab en la solución de armaduras planas y espaciales de ingeniería*. [Estudio de investigación].
Universidad Nacional de Cajamarca.
- Díaz, F. & Guillén, A. (2020). Modelo computacional para el análisis matricial de estructuras reticulares. *Universidad Peruana de Ciencias Aplicadas (UPC)*.
<https://repositorioacademico.upc.edu.pe/handle/10757/648845>
- Faizan, A. (2016). *Analysis of plane frame structure with MATLAB and SAP2000 programs*.
- Hibbeler, R. (2022). *ANÁLISIS ESTRUCTURAL* (Novena edición). Pearson.
- Joyanes, L. (2020). *Fundamentos de programación. Algoritmos, estructura de datos y objetos* (5ta ed.). Mc Graw Hill.
- Kassimali, A. (2022). *Matrix Analysis of Structures* (Tercera edición). Cengage.
- McGuire, W., W., Gallagher, R., & Ziemian, R. (2015). *Matrix Structural Analysis*, 2nd Edition. *Faculty Books*. <https://digitalcommons.bucknell.edu/books/7>
- Moore, H. (2022). *MATLAB® for Engineers* (6th ed.). Pearson Education. Inc.
- Navarro, C. (2021). *Introducción al análisis de estructuras*. Universidad Carlos III de Madrid.
- Ortiz Soto, D., Merma Taipe, H., & Cadena Pérez, I. (2022). *Análisis estructural de armaduras en dos dimensiones con el Método Matricial de Rigidez empleando*

- programación en Excel y MATLAB. En *Avances de Investigación en Ingeniería Aplicada* (1ª edición, pp. 329–343). Instituto Tecnológico Superior de los Ríos.
- Rupay Vargas, M., Monago Tarazona, I., Napoleón Rodríguez, L., Rojas Bermudo, C. & Teraccaya Samaniego, K. (2023) *Análisis estructural de armaduras planas aplicando el método matricial de rigidez*. *Yotantsipanko*, 3(1), 74 -83.
<https://doi.org/10.54288/yotantsipanko.v3i1.31>
- Turner, M. J., Clough, R. W., Martin, H. C., & Topp, L. J. (1956). Stiffness and deflection analysis of complex structures. *Journal of the Aeronautical Sciences*, 23(9), 805–823.

ANEXOS

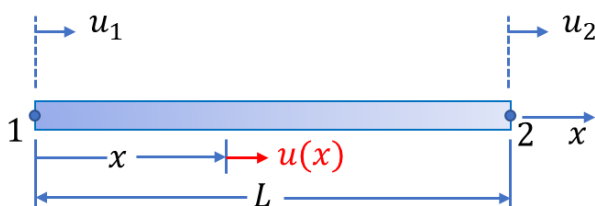
ANEXO A. RELACIÓN ENTRE EL MÉTODO DE ELEMENTO FINITO Y EL MÉTODO MATRICIAL DE RIGIDEZ.

la matriz de rigidez $[K]$ se puede obtener mediante el método de elementos finitos y el método matricial de rigidez. El primero utiliza un proceso matemático de discretización del dominio, llegando a una solución generalmente aproximada mediante las funciones de forma, aunque, para el caso de barras, la solución es exacta y por ende coincide con el método matricial de rigidez. La versatilidad del MEF radica en que se pueden analizar elementos de placa, membrana y elementos tridimensionales en general. SAP2000 V23.3.1 usa elementos finitos como método predominante, considerando 6 grados de libertad (GDL) por nodo, haciendo un total de 12 GDL por ejemplo para el análisis de estructuras aporticadas; sin embargo, es posible adecuar el entorno de trabajo, para estructuras bidimensionales, esto es, analizar a la estructura en tan solo dos ejes ortogonales que típicamente forman el plano XZ, lo cual se traduce en la simplificación de 3 GDL en cada nodo extremo de barra, haciendo un total de 6 GDL para los nodos de interés de cada elemento idealizado, coincidiendo finalmente en el caso simplificado de estructuras bidimensionales compuestos de elementos tipo barra.

A continuación, se analiza el caso del elemento barra con desplazamiento uniaxial por el método de elementos finitos.

Figura 24:

Diagrama del elemento barra con su función de desplazamiento $u(x)$



Considerando el comportamiento elástico de la barra libre de apoyos, y tomando el marco de referencia local a través del eje horizontal, se denota el desplazamiento longitudinal $u(x)$ para cualquier posición, siendo $u_1 = u(x = 0)$ y $u_2 = u(x = L)$, habiéndose generado así la variable continua de campo $u(x)$, misma que se puede aproximar en función de los desplazamientos nodales u_1 y u_2 .

Es preciso emplear las funciones de interpolación $N_1(x)$ y $N_2(x)$, también llamadas funcione de forma, quedando la función de desplazamientos como sigue:

$$u(x) = N_1(x)u_1 + N_2(x)u_2 \quad (2.1)$$

Dicha ecuación muestra que es posible encontrar el desplazamiento en cada punto “x” a lo largo de toda la barra de longitud “L”, para lo cual se deben calcular las funciones de forma, siendo necesario que los valores de frontera, es decir los desplazamientos nodales sean satisfechos por la discretización, así pues:

$$u(x = 0) = u_1 ; \quad u(x = L) = u_2 \quad (2.2)$$

Reemplazando (2.2) en (2.1), para el caso $x = 0$, se tiene:

$$u(0) = N_1(0)u_1 + N_2(0)u_2 = u_1$$

Con $u_2 \neq 0$, se tiene:

$$\rightarrow N_1(0) = 1 \wedge N_2(0) = 0$$

Reemplazando (2.2) en (2.1), para el caso $x = L$, se tiene:

$$u(L) = N_1(L)u_1 + N_2(L)u_2 = u_2$$

Con $u_1 \neq 0$, se tiene:

$$\rightarrow N_1(L) = 0 \wedge N_2(L) = 1$$

Para este caso de desplazamiento uniaxial, las formas más sencillas para los elementos de forma son polinómicas de tipo lineal, así:

$$N_1(x) = a_0 + a_1x \quad (2.3)$$

$$N_2(x) = b_0 + b_1x \quad (2.4)$$

Pero nótese que $N_1(0) = 1$, y evaluando dicho caso en (2.3), se tiene:

$$N_1(0) = a_0 + a_1 \cdot (0)$$

$$1 = a_0 + \cancel{a_1 \cdot (0)}$$

$$\text{De donde, } a_0 = 1$$

Nótese ahora, que $N_1(L) = 0$, y evaluando dicho caso en (2.3), se tiene:

$$N_1(L) = a_0 + a_1 \cdot (L)$$

Y como $a_0 = 1$:

$$0 = 1 + a_1 \cdot (L)$$

Despejando a_1 :

$$a_1 = -\frac{1}{L}$$

Ahora analizamos el caso $N_2(0) = 0$ en (2.4):

$$N_2(x) = b_0 + b_1x$$

$$N_2(0) = b_0 + b_1 \cdot 0$$

$$0 = b_0 + \cancel{b_1 \cdot 0}$$

De donde:

$$b_0 = 0$$

Ahora analizamos el caso $N_2(L) = 1$ en (2.4):

$$N_2(x) = b_0 + b_1x$$

$$N_2(L) = b_0 + b_1 \cdot L$$

$$1 = 0 + b_1L$$

$$b_1 = \frac{1}{L}$$

Reemplazando los valores obtenidos en las funciones de forma, se tiene:

$$N_1(x) = a_0 + a_1x$$

$$N_1(x) = 1 + \frac{-1}{L}x$$

$$N_1(x) = 1 - \frac{x}{L}$$

$$N_2(x) = b_0 + b_1x$$

$$N_2(x) = 0 + \frac{1}{L}x$$

$$N_2(x) = \frac{x}{L}$$

Y reemplazando las funciones de forma en la ecuación general de desplazamientos escrita en (2.1), se tiene:

$$u(x) = N_1(x)u_1 + N_2(x)u_2$$

$$u(x) = \left(1 - \frac{x}{L}\right)u_1 + \left(\frac{x}{L}\right)u_2 \quad (2.5)$$

Esta última ecuación puede expresarse en forma matricial como:

$$u(x) = [N_1(x) \ N_2(x)] \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} = [N]\{u\} \quad (2.6)$$

En dicha ecuación:

$[N]$: Matriz de funciones de forma o interpolación (vector fila).

$\{u\}$: Vector columna de desplazamientos uniaxiales en los nodos.

Para poder definir los desplazamientos para cualquier valor de “x”, con exactitud, es necesario determinar una relación entre los desplazamientos nodales u_1 y u_2 , así como las fuerzas aplicadas en dichos nodos.

De mecánica de materiales, se conoce la relación entre los parámetros de deformación, mediante:

$$\delta = \frac{PL}{EA}$$

Donde:

δ : Deformación uniaxial.

P : Carga axial

L : Longitud del elemento barra

E : Módulo de elasticidad

A : Área de la sección transversal

Despejando “P”:

$$P = \frac{EA}{L} \delta$$

Pero por la ley de Hooke para elementos elásticos, se tiene:

$$P = k \cdot \delta$$

Donde:

k : Constante de elasticidad.

Despejando k :

$$k = \frac{P}{\delta}$$

Pero $\frac{P}{\delta}$ también es $\frac{EA}{L}$, de modo que:

$$k = \frac{EA}{L}$$

Además,

$$\delta = u_2 - u_1$$

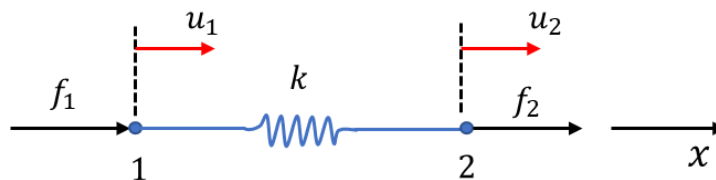
Así que reemplazando k y δ en la ley de Hooke para el análisis de cargas en la barra con desplazamientos uniaxiales:

$$P = \frac{EA}{L} \cdot (u_2 - u_1) \quad (2.7)$$

Ahora, incluyendo el caso de cargas en el elemento, considerando su comportamiento elástico lineal:

Figura 25:

Fuerzas y desplazamientos nodales



Siendo:

$$P = \begin{Bmatrix} f_1 \\ f_2 \end{Bmatrix}$$

Y considerando que el elemento está traccionado como caso positivo de carga, y considerando a f_2 como la carga positiva, quiere decir que f_1 debe tener el mismo módulo, pero sentido contrario, por lo que según aparece el gráfico, debe tener un coeficiente negativo, por lo que:

$$f_1 = -\frac{EA}{L} \cdot (u_2 - u_1)$$

$$f_2 = \frac{EA}{L} \cdot (u_2 - u_1)$$

Así, la ecuación (2.7) se convierte en:

$$\begin{aligned} \begin{Bmatrix} f_1 \\ f_2 \end{Bmatrix} &= \frac{EA}{L} \cdot \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} \\ \begin{Bmatrix} f_1 \\ f_2 \end{Bmatrix} &= \begin{bmatrix} \frac{EA}{L} & -\frac{EA}{L} \\ -\frac{EA}{L} & \frac{EA}{L} \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} \end{aligned} \quad (2.8)$$

Donde:

$$\begin{bmatrix} \frac{EA}{L} & -\frac{EA}{L} \\ -\frac{EA}{L} & \frac{EA}{L} \end{bmatrix} = [\mathbf{K}]$$

De manera similar se puede realizar el estudio comparativo de la relación del método matricial de rigidez con el método del elemento finito para los casos de vigas y pórticos.

ANEXO B. ALGORITMOS Y DIAGRAMAS DE FLUJO

Joyanes, L. (2020), refiere que un algoritmo es un conjunto finito de pasos que involucra operaciones computacionales elementales que conducen a la solución de un problema específico, y dado que la secuencia de pasos textuales puede no ser tan clara para los usuarios, se realizan diagramas a los cuales se les conoce típicamente como diagramas de flujo o *flowcharts*.

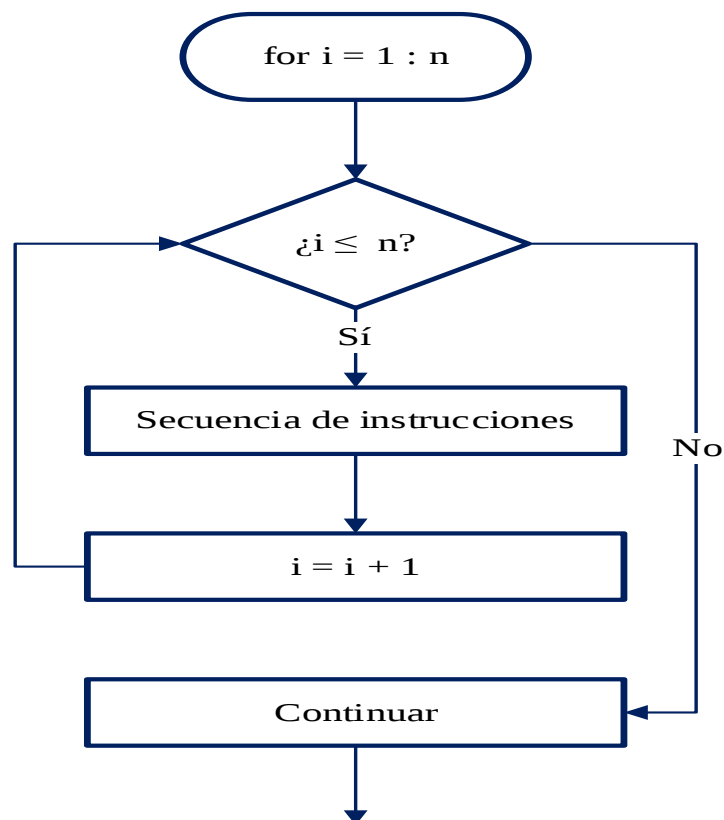
Sentencias condicionales

Ciclo *for*:

El ciclo ***for*** en MATLAB empieza con una variable de control ***i*** que parte con un valor inicial, generalmente 1, cada vez que se cumple la condición $i \leq n$, se ejecuta un bloque de instrucciones. Posteriormente, la variable ***i*** se incrementa en 1. Si después de este incremento la condición sigue siendo verdadera, el ciclo se repite. Si en algún momento ***i*** excede el valor de ***n***, el ciclo termina y el programa continúa con las instrucciones siguientes fuera del ciclo.

Figura 26:

Flujograma para ciclo “for”



En MATLAB, se tendría una estructura versátil como sigue:

```
%% Estructura del ciclo “for”
```

```
% Fijar valor para “n”  
for  $i=1:n$   
% Secuencia de instrucciones  
end
```

Ejemplo:

```
%% Estructura del ciclo “for”
```

```
n=6;  
for  $i=1:n$   
disp( $i^2$ )  
end
```

En el command window se obtendrá:

```
Command window
```

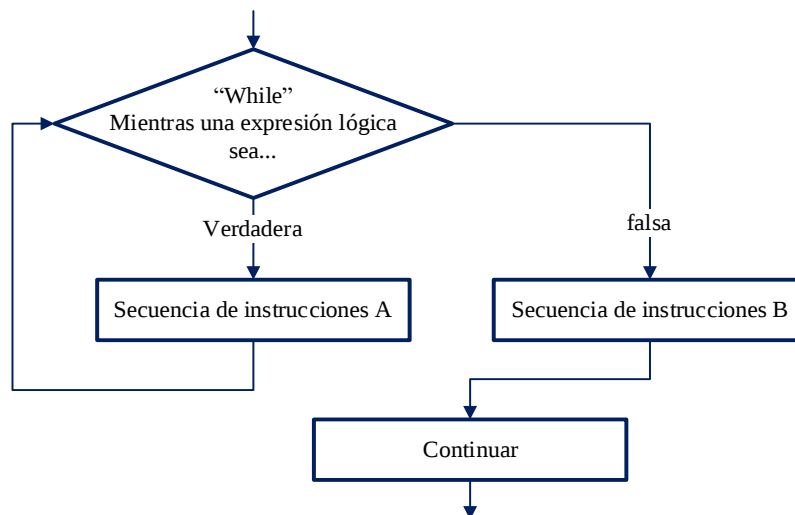
```
1  
4  
9  
16  
25  
36
```

Ciclo *While*

es una estructura de control que permite repetir una secuencia de instrucciones mientras se cumpla una condición lógica, es decir, si la expresión es verdadera, se ejecuta el bloque de instrucciones contenido dentro del ciclo (representado como "Secuencia de instrucciones A" en el diagrama). Después de cada iteración, la condición se vuelve a evaluar. Si sigue siendo verdadera, el ciclo se repite. Si en algún momento la expresión se vuelve falsa, el ciclo se interrumpe y se ejecuta la "Secuencia de instrucciones B" (que representa lo que sucede fuera del ciclo, una vez terminado). Luego, el programa sale del bucle y continúa con el resto de las instrucciones.

Figura 27:

Flujograma para ciclo "While"



En MATLAB, se tendría una estructura versátil como sigue:

```
% Script de estructura para el ciclo while
```

```
while condición_lógica
```

```
% Secuencia de instrucciones A
```

```
end
```

```
% Secuencia de instrucciones B
```

Ejemplo:

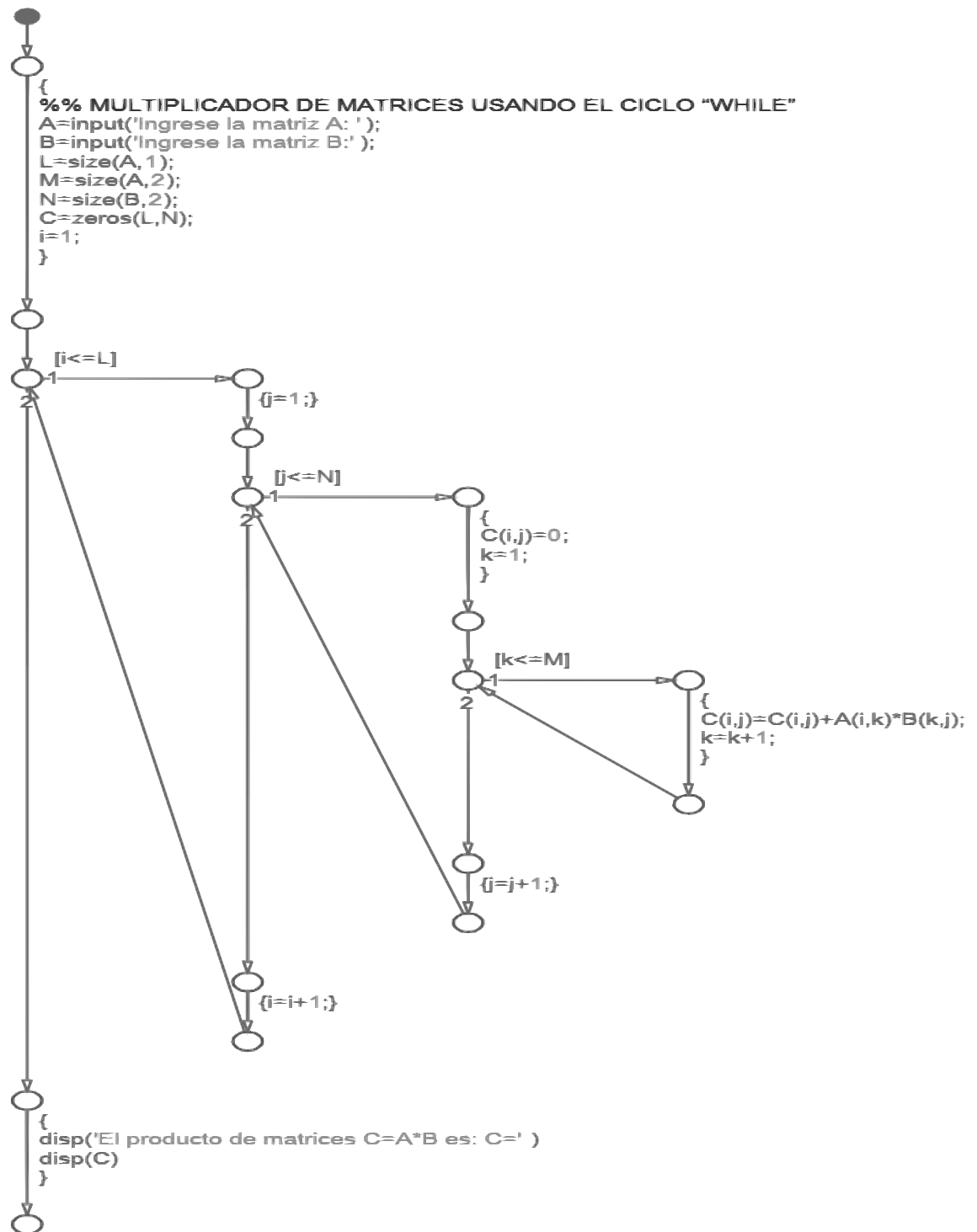
```
%% MULTIPLICADOR DE MATRICES USANDO EL CICLO
“WHILE”
A=input('Ingrese la matriz A: ');
B=input('Ingrese la matriz B:');
L=size(A,1);
M=size(A,2);
N=size(B,2);
i=1;
while i<=L
    j=1;
    while j<=N
        C(i,j)=0;
        k=1;
        while k<=M
            C(i,j)=C(i,j)+A(i,k)*B(k,j);
            k=k+1;
        end
        j=j+1;
    end
    i=i+1;
end
disp(C)
```

El ejemplo muestra la versatilidad del uso del ciclo while para multiplicar matrices y, aunque MATLAB tiene la capacidad de multiplicar las matrices sin la necesidad del código, este nos sirve para entender los fundamentos lógicos y matemáticos de dicha operación.

En MATLAB es posible obtener directamente los diagramas de flujo, para lo cual se pueden seguir los pasos del Anexo B. El resultado es como sigue:

Figura 28:

Flujograma del multiplicador de matrices en Flowchart de MATLAB.



Se muestra el empleo del ciclo while, de manera anidada. Las etiquetas “1” y “2” hacen referencia a la veracidad o falsedad, respectivamente, de la expresión lógica a analizar.

Aunque las líneas de código cumplen el propósito de mostrar el potencial del ciclo while empleado en MATLAB, este puede interpretar la multiplicación de matrices sin la necesidad de un extenso código. Tan solo bastará con establecer las matrices a multiplicar (“A” y “B”), y utilizar el símbolo de multiplicación para obtener el resultado de dicha operación, es decir, la matriz C.

Ejemplo:

Multiplicación de matrices en MATLAB.

Código MATLAB	Descripción
>> A=[2 3 5; 4 7 9];	Se define una matriz de 2x3 llamada A.
>> B=[8 6; 10 12; 1 6];	Se define una matriz de 3x2 llamada B.
>> C=A*B	Se realiza la multiplicación de matrices A y B, almacenando el resultado en la matriz C.
C =	Se muestra la matriz C resultante de la multiplicación.
51 78	
111 162	

Recordemos que, en el caso de la multiplicación de matrices, el número de columnas de la matriz A debe coincidir con el número de filas de la matriz B, es decir, si A es una matriz de dimensión $m \times n$ (m filas y n columnas) y B es una matriz de dimensión $n \times p$ (n filas y p columnas), entonces la multiplicación de $A \times B$ es posible y da como resultado una matriz de dimensión $m \times p$.

En el aspecto de la programación, conviene advertir al usuario sobre esta propiedad, para no incurrir en este tipo de errores.

Error por dimensiones incompatibles en multiplicación de matrices.

Código MATLAB	Descripción
>> A=[2 3 5; 4 7 9];	Se define una matriz de 2x3 llamada A.
>> B=[8 6 10; 12 1 6];	Se define una matriz de 2x3 llamada B.
>> Z=A*B	Se intenta realizar la multiplicación de matrices A y B, almacenando el resultado en la matriz Z.
Error using * Incorrect dimensions for matrix multiplication. Check that the number of columns in the first matrix matches the number of rows in the second matrix. To perform elementwise multiplication, use '.*'.	Se produce un error debido a que las dimensiones de las matrices A y B no son compatibles para realizar la multiplicación de matrices. La multiplicación de matrices solo se puede realizar si el número de columnas de la primera matriz es igual al número de filas de la segunda matriz. En este caso, la matriz A tiene 3 columnas, pero la matriz B tiene solo 2 filas. El mensaje de error sugiere utilizar el operador .* si se desea realizar una multiplicación elemento a elemento.

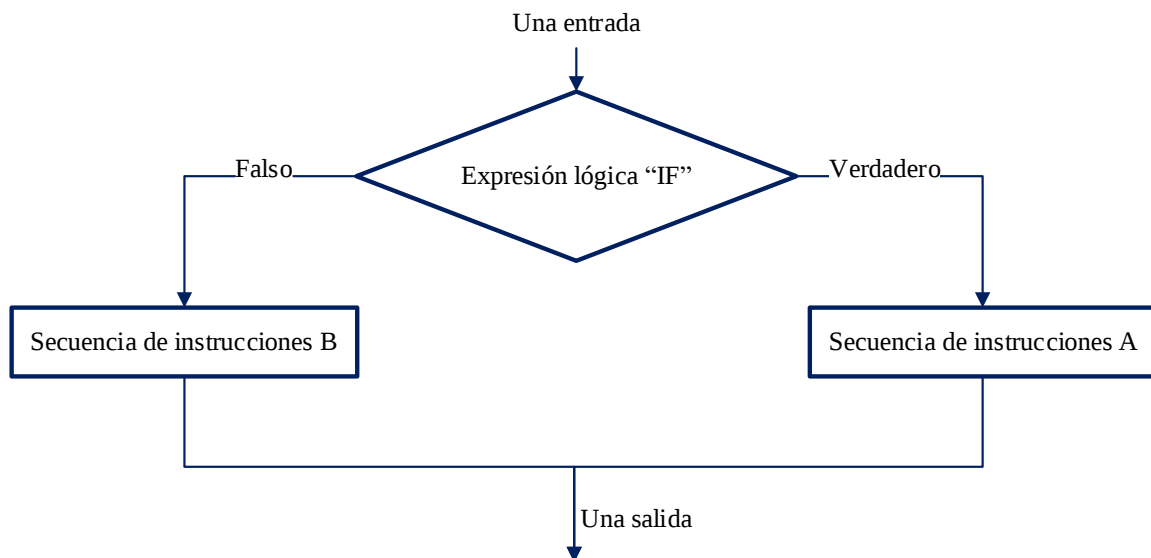
Lo anterior se puede resolver, usando un ciclo “if” que advierta sobre dicha propiedad al usuario de un determinado programa.

Ciclo if

Este ciclo necesita una entrada con expresión lógica, la cual es evaluada y, dependiendo de su valor de verdad se ejecutan las secuencias de instrucciones; si es verdadero, se sigue la secuencia de instrucciones A, en cambio, si el valor de verdad de la expresión lógica es falso se sigue la secuencia de instrucciones B. Al finalizar la evaluación de ambos casos, existe una salida.

Figura 29:

Diagrama de flujo del ciclo IF



Ejemplo:

% UBICACIÓN DE LA TABLA EN LA VENTANA DE ÁREAS

```
cant_areas=app.CantidaddeseccionestpicasEditField.Value;
```

```
if cant_areas<=4
```

```
    h=28.8+cant_areas*22.4;                % Altura de la tabla.
```

```
    esqui_inf_izq=119.6-cant_areas*22.4;    % Coordenada inferior izquierda.
```

```
    app.UITable_Areas.Position = [97 esqui_inf_izq 129 h]; %Posición de tabla.
```

```
else
```

```
    app.UITable_Areas.Position = [97 30 129 119.6];
```

```
end
```

Descripción de los elementos del código para la ventana de áreas.

Elemento del código	Descripción detallada
% UBICACIÓN DE LA TABLA EN LA VENTANA DE ÁREAS	Comentario descriptivo: Indica el propósito de este bloque de código, señalando que se trata de determinar la posición y dimensiones de una tabla dentro de una ventana.
cant_areas = app.CantidaddeseccionestpicasEditField.Value;	Asignación de valor: Obtiene el valor numérico ingresado por el usuario en campo CantidaddeseccionestpicasEditField. Este valor representa la cantidad de áreas o secciones típicas existentes. El valor obtenido se almacena en la variable cant_areas para su uso posterior en los cálculos.
if cant_areas <= 4	Condición: Evalúa si el valor almacenado en cant_areas es menor o igual a 4. Esta condición bifurca el flujo del programa y ejecuta diferentes bloques de código según el resultado de la comparación.
h = 28.8 + cant_areas * 22.4;	Cálculo de la altura: Si la condición anterior es verdadera (hay 4 o menos áreas), se calcula la altura total de la tabla en píxeles. La altura se determina sumando un valor base de 28.8 píxeles y un incremento de 22.4 píxeles por cada área adicional. Esta ecuación asegura que la tabla tenga suficiente espacio para mostrar todas las áreas. El resultado se almacena en la variable h.
esqui_inf_izq = 119.6 - cant_areas * 22.4;	Cálculo de la posición vertical: Calcula la posición vertical inicial de la esquina inferior izquierda de la ventana. Esta posición se ajusta en función del número de áreas, de modo que la tabla se desplace a medida que aumenta el número de áreas. El resultado se almacena en la variable esqui_inf_izq.
app.UITable_Areas.Position = [97 esqui_inf_izq 129 h];	Asignación de posición y tamaño: Establece las propiedades de posición y tamaño de la tabla en la interfaz gráfica. Los valores dentro de los corchetes representan: 97: Posición horizontal fija (en píxeles) de la esquina inferior izquierda de la tabla; esqui_inf_izq: Posición vertical calculada dinámicamente (en píxeles) de la esquina inferior izquierda de la tabla; 129: Ancho fijo de la tabla (en píxeles); h: Altura de la tabla calculada previamente (en píxeles).
else	Condición alternativa: Si la condición inicial (cant_areas <= 4) es falsa (es decir, hay más de 4 áreas), se ejecuta este bloque de código.
app.UITable_Areas.Position = [97 30 129 119.6];	Asignación de posición y tamaño fija: Establece una posición y tamaño fijos para la tabla cuando hay más de 4 áreas. Estos valores están predefinidos y no se calculan dinámicamente.
end	Fin del condicional: Marca el final del bloque if-else, indicando que el programa continúa con la siguiente instrucción después de este bloque.

Se muestra una propuesta de una parte del código de PAME UNC 2D para la pestaña de propiedades de sección, específicamente para las áreas de las secciones transversales de los elementos. Se inicia recopilando el dato para la variable “cant_areas”, el cual se obtendrá de la ventana: “PROPIEDADES_DE_SECCION_AREA”, cuya estructura se detallará en el apartado dirigido a App Designer de MATLAB.

Luego de obtener ese dato, se emplea el ciclo if, para determinar si cant_areas es menor o igual a cuatro; de ser cierto, se establece la altura de la tabla como:
$$h=28.8+cant_areas*22.4,$$

además de determinar las coordenadas para la esquina inferior izquierda de la tabla, usando: $esqui_inf_izq=119.6-cant_areas*22.4$, para luego obtener la posición de la tabla. De ser falso el caso analizado, se establecerá una posición fija para la tabla:
`app.UITable_Areas.Position = [97 30 129 119.6];`

un ejemplo más simplificado, respecto al error mostrado en el ejemplo de la multiplicación de matrices, y, optimizando las líneas de código para la multiplicación de matrices es como sigue:

```
%% MULTIPLICADOR DE MATRICES EFICIENTE
% Solicitar al usuario las matrices
A = input('Ingrese la matriz A: ');
B = input('Ingrese la matriz B: ');

% Verificar las dimensiones
if size(A, 2) == size(B, 1)
    % Multiplicar las matrices
    C = A * B;
    disp('El resultado de la multiplicación de matrices es:');
    disp(C);
else
    error('El número de columnas de A debe ser igual al número de filas de B. ');
end
```

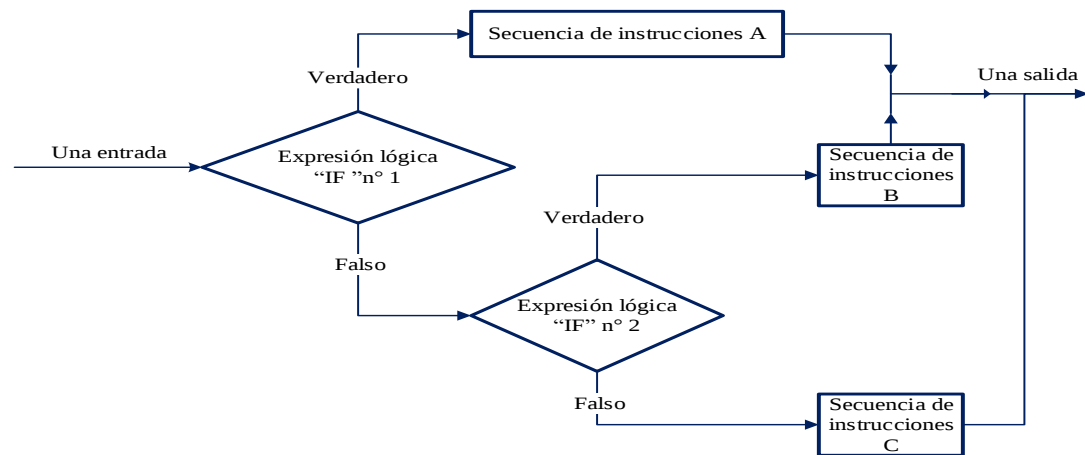
Descripción y función de elementos de código para multiplicar matrices.

Código	Descripción	Función
% MULTIPLICADOR DE MATRICES EFICIENTE	Comentario que indica el propósito del código.	Documentación; no tiene efecto en la ejecución.
A = input('Ingrese la matriz A: ');	Solicita al usuario que introduzca los elementos de la matriz A desde la línea de comandos.	Lee la entrada del usuario y la asigna a la variable A.
B = input('Ingrese la matriz B: ');	Solicita al usuario que introduzca los elementos de la matriz B desde la línea de comandos.	Lee la entrada del usuario y la asigna a la variable B.
if size(A, 2) == size(B, 1)	Comprueba si el número de columnas de la matriz A (size(A, 2)) es igual al número de filas de la matriz B (size(B, 1)). Esta es la condición necesaria para que la multiplicación de matrices sea válida.	Control de flujo; determina si se puede realizar la multiplicación. size(A,2) devuelve el número de columnas de A. size(B,1) devuelve el número de filas de B.
C = A * B;	Realiza la multiplicación de las matrices A y B. El resultado se almacena en la matriz C.	Realiza la operación de multiplicación de matrices.
disp('El resultado de la multiplicación de matrices es:');	Muestra un mensaje en la consola indicando que se va a mostrar el resultado.	Presentación; mejora la legibilidad de la salida.
disp(C);	Muestra el resultado de la multiplicación (matriz C) en la consola.	Presentación; muestra el resultado del cálculo.
else	Bloque de código que se ejecuta si la condición del if es falsa (es decir, si las dimensiones de las matrices no son compatibles para la multiplicación).	Control de flujo; define el comportamiento en caso de error.
error('El número de columnas de A debe ser igual al número de filas de B.');	Muestra un mensaje de error en la consola y detiene la ejecución del programa, informando al usuario de la condición de error.	Manejo de errores; informa al usuario sobre un error en la entrada.
end	Cierra los bloques if/else.	Control de flujo; delimita el final de la estructura condicional.

Al igual que los demás ciclos antes mencionados, el ciclo “*if*” tiene la posibilidad de anidarse en la secuencia de código, tal como se verifica en el posterior diagrama deflujo.

Figura 30:

Ciclo IF anidado.



Se inicia con una entrada de datos con una expresión lógica “IF” n°1 a evaluar; si la expresión resulta ser verdadera, se procede con la secuencia de instrucciones A, de ser falsa se procede con el análisis de la nueva expresión lógica “IF” n°2, de resultar verdadera, se procede al desarrollo de la secuencia de instrucciones B, y en caso contrario, se procede con la secuencia de instrucciones C, para finalmente salir del bucle.

Código genérico de ciclo if anidado en MATLAB:

```
% Esquema de un ciclo IF anidado  
% información de entrada  
entrada = ...;  
% ciclo if principal  
if (Expresión lógica n°1) % Reemplaza con tu expresión lógica 1  
    % Si la primera condición es verdadera, ejecuta las instrucciones A  
else  
    % Si la primera condición es falsa, evalúa la segunda condición  
    if (Expresión lógica n°2)  
        % Si la segunda condición es verdadera, ejecuta las instrucciones B  
    else  
        % Si ninguna de las condiciones es verdadera, ejecutar C.  
    end  
end  
% Una salida (opcional, si se desea mostrar un resultado final)  
resultado = ...; % Cálculo del resultado.  
disp(['El resultado es: ', num2str(resultado)]);
```

ANEXO C. CÓDIGO PRINCIPAL DE PAME UNC 2D

```
classdef PAME < matlab.apps.AppBase
    % Properties that correspond to app components
    properties (Access = public)
        PAMEUNC2DUIFigure          matlab.ui.Figure
        ArchivoMenu                 matlab.ui.container.Menu
        NuevoMenu                   matlab.ui.container.Menu
        AbrirMenu                   matlab.ui.container.Menu
        GuardarMenu                 matlab.ui.container.Menu
        OpcionesMenu                matlab.ui.container.Menu
        SalirMenu                   matlab.ui.container.Menu
        CrearMenu                   matlab.ui.container.Menu
        MaterialeMenu              matlab.ui.container.Menu
        SeccinAMenu                 matlab.ui.container.Menu
        AsignarMenu                 matlab.ui.container.Menu
        AsignarMaterialMenu         matlab.ui.container.Menu
        AsignarSeccinMenu          matlab.ui.container.Menu
        VisualizarMenu             matlab.ui.container.Menu
        VernodosMenu               matlab.ui.container.Menu
        VersolodibujodenodosMenu   matlab.ui.container.Menu
        VerIDdenodosMenu           matlab.ui.container.Menu
        VerbarrasMenu              matlab.ui.container.Menu
        VerbarrasSNMenu            matlab.ui.container.Menu
        VerIDdebarrasMenu          matlab.ui.container.Menu
        VermaterialesMenu          matlab.ui.container.Menu
        verseccionesMenu           matlab.ui.container.Menu
        ApoyosMenu                 matlab.ui.container.Menu
        DibujarapoyosMenu          matlab.ui.container.Menu
        CargasMenu                 matlab.ui.container.Menu
        CargaNodalMenu             matlab.ui.container.Menu
        CargadistribuidaMenu       matlab.ui.container.Menu
        Resumen_datosMenu          matlab.ui.container.Menu
        DatospdfMenu               matlab.ui.container.Menu
        MDULO_ANLISISMenu          matlab.ui.container.Menu
        GDLLMenu                   matlab.ui.container.Menu
        IDNROCOORDNODOMenu        matlab.ui.container.Menu
    end
end
```

```

GKMATRIZPORBARRAMenu          matlab.ui.container.Menu
KMATRIZENSAMBLADAMenu          matlab.ui.container.Menu
FVECTORCARGASNODMenu           matlab.ui.container.Menu
RESULTADOSMenu                 matlab.ui.container.Menu
UVECTORDEDESPLAZAMenu          matlab.ui.container.Menu
FUERZASAXIALESMenu            matlab.ui.container.Menu
REACCIONESNAPOYOSMenu         matlab.ui.container.Menu
GENERARREPORTERESULTADOSMenu  matlab.ui.container.Menu
InfoMenu                       matlab.ui.container.Menu
AyudaMenu                      matlab.ui.container.Menu
VerManualdeUsuarioMenu         matlab.ui.container.Menu
AcercadeMenu                   matlab.ui.container.Menu
ververMenu                     matlab.ui.container.Menu
verdiagramMenu                 matlab.ui.container.Menu
YLabel                         matlab.ui.control.Label
XLabel                         matlab.ui.control.Label
UIAxes_modelofinal             matlab.ui.control.UIAxes
ContextMenu                    matlab.ui.container.ContextMenu
UniformizarcoloresdebarrasMenu matlab.ui.container.Menu
FondoblancoMenu                matlab.ui.container.Menu
FondonegroMenu                 matlab.ui.container.Menu
end

```

```

% Callbacks that handle component events

```

```

methods (Access = private)

```

```

    % Code that executes after component creation

```

```

    function startupFcn(app)

```

```

    %LIMPIAR EL AXES LLAMADO app.UIAxes_modelofinal

```

```

        cla(app.UIAxes_modelofinal); % Cada vez que se
        abre el programa PAME 2D UNC se borra el UIaxes.

```

```

        clearvars -global; % Cada vez que se abre el
        programa, se borran las variables que se hayan podido generar en
        anteriores ejecuciones.

```

```

        app.UIAxes_modelofinal.BackgroundColor='black';

```

```

        app.UIAxes_modelofinal.Color='black';

```

```

end

```

```

% Menu selected function: NuevoMenu
function NuevoMenuSelected(app, event)
    clearvars -global; % Cada vez que se crea un nuevo
modelo, borrará todas las variables consignadas como variables.
    cla(app.UIAxes_modelofinal); % Cada vez que se abre
un nuevo modelo, se borra el UIAxes.
    Nuevo; %Abre la otra ventana de app designer
llamada "Nuevo"
    %% IDENTIFICADORES DE PESTAÑAS DEL MENU:
    global ID_ABRIR_MENU ID_GUARDAR_MENU...
        ID_CREAR_MENU ID_SECCIN_MENU...
        ID_ASIGNAR_MENU ID_ASIGNAR_SECCIN_MENU...
        ID_VISUALIZAR_MENU ID_VER_ID_DE_NODOS...
        ID_CARGAS_MENU ID_CARGA_DISTRIBUIDA...
        ID_RESUMEN_DATOS_MENU...
        ID_MODULO_ANALISIS_MENU
ID_ABRIR_MENU=app.AbrirMenu;
ID_GUARDAR_MENU=app.GuardarMenu;

ID_CREAR_MENU=app.CrearMenu;
ID_SECCIN_MENU=app.SeccinAMenu;

ID_ASIGNAR_MENU=app.AsignarMenu;
ID_ASIGNAR_SECCIN_MENU=app.AsignarSeccinMenu;

ID_VISUALIZAR_MENU=app.VisualizarMenu;
ID_VER_ID_DE_NODOS=app.VerIDdenodosMenu;

ID_CARGAS_MENU=app.CargasMenu;
ID_CARGA_DISTRIBUIDA=app.CargadistribuidaMenu;

ID_RESUMEN_DATOS_MENU=app.Resumen_datosMenu;

ID_MODULO_ANALISIS_MENU=app.MDULO_ANLISISMenu;
end

```

```

% Menu selected function: VersolodibujodenodosMenu
function VersolodibujodenodosMenuSelected(app, event)
    global dibu UIAxes_de_PAME ... % global, porque se usará luego, al
    activar la pestaña de asignar apoyo
    alto_y
    app.UIAxes_modelofinal.DataAspectRatio=[1 1 1];
    UIAxes_de_PAME=app.UIAxes_modelofinal;
    global coordenadas_x coordenadas_y % vectores columna de las coordenadas
    dibu=plot(app.UIAxes_modelofinal,coordenadas_x,coordenadas_y,'.','Mark
    erSize",20,"Color",'yellow'); % Asignamos a la variable "dibu" los datos
    ploteados.

    %% Límites del UIAxes en el eje X:
    der=app.UIAxes_modelofinal.XLim(2);
    izq=app.UIAxes_modelofinal.XLim(1);

    %% Límites del UIAxes en eje Y:
    y_superior=app.UIAxes_modelofinal.YLim(2);
    y_inferior=app.UIAxes_modelofinal.YLim(1);
    alto_y=y_superior-y_inferior;

    %% ACTIVAR LA SIGUIENTE PESTAÑA: VER ID DE NODOS:
    global ID_VER_ID_DE_NODOS
    ID_VER_ID_DE_NODOS.Enable='on'; % Como este componente está en esta
    ventana, en su reemplazo se puede usar: app.VerIDdenodosMenu.Enable='on'
    end

% Menu selected function: VerIDdenodosMenu
function VerIDdenodosMenuSelected(app, event)
    global coordenadas_x coordenadas_y ... vectores columna de las
    coordenadas
    offset_graf % El desfase usado.
    % PARA OFFSET CON PITAGORAZO (EN FUNCIÓN DE LA LONGITUD DE LA
    % PRIMERA BARRA):
    a=coordenadas_x(2)-coordenadas_x(1);
    b=coordenadas_y(2)-coordenadas_y(1);
    long_primer_elem=sqrt(a^2+b^2);
    % FIN DE PITAGORAZO

```

```

        offset_graf=long_primer_elem/35; % El offset será la 35-ava parte de la
longitud de la primera barra

    % ETIQUETA DE CADA NODO:

    for i=1:size(coordenadas_x) % Normal detecta el primer parámetro (n) del
vector tamaño (n x 1)

        etiqueta_nodo=num2str(i);

        text(app.UIAxes_modelofinal,coordenadas_x(i)+offset_graf,coordenadas_y
(i)+offset_graf,etiqueta_nodo,'Color','c','FontSize',20);

    end

    %% VER SIGUIENTE PESTAÑA: VER BARRAS:

    app.VerbarrasMenu.Enable='on';

    end

    % Menu selected function: VerbarrasMenu

    function VerbarrasMenuSelected(app, event)

    end

    % Menu selected function: UniformizarcoloresdebarrasMenu

    function UniformizarcoloresdebarrasMenuSelected(app, event)

    cla(app.UIAxes_modelofinal);

    global coordenadas_x coordenadas_y ...

    nro_barras ...

    barras_i_f ...

    coordenadas_x_y ...

    for i=1:nro_barras

        JB=barras_i_f(i,1); %Aquí ya se puede usar con paréntesis (i,1), porque
ya es matriz y no celda. Lo mismo en siguientes:

        JE=barras_i_f(i,2);

        XB=coordenadas_x_y(JB,1);

        YB=coordenadas_x_y(JB,2);

        XE=coordenadas_x_y(JE,1);

        YE=coordenadas_x_y(JE,2);

        vect_x=[XB XE]';

        vect_y=[YB YE]';

        plot(app.UIAxes_modelofinal,vect_x,vect_y,"LineWidth",2,"Color",'green
');

        hold(app.UIAxes_modelofinal,'on');

    end

```

```

        plot(app.UIAxes_modelofinal, coordenadas_x, coordenadas_y, '.', "MarkerSize", 20, "Color", 'yellow');
        hold(app.UIAxes_modelofinal, 'off');
    end

    % Menu selected function: MaterialeMenu
    function MaterialeMenuSelected(app, event)
        MATERIAL;
    end

    % Menu selected function: VerManualdeUsuarioMenu
    function VerManualdeUsuarioMenuSelected(app, event)
        % open('Ayuda_Manual_de_usuario_PAME.pdf'); % Mejorado a winopen

        winopen('Ayuda_Manual_de_usuario_PAME.pdf');
        % Agregar archivo del manual (pdf) al compilar. Fuente:
        https://la.mathworks.com/matlabcentral/answers/489203-how-can-i-open-a-pdf-in-a-compiled-application

    end

    % Menu selected function: AcercadeMenu
    function AcercadeMenuSelected(app, event)
        Acerca_De;
    end

    % Menu selected function: SalirMenu
    function SalirMenuSelected(app, event)
        close(app.PAMEUNC2DUIFigure); % Cierra la aplicación principal.
        %TAMBIÉN PODEMOS USAR app.delete:
        %app.delete;
    end

    % Menu selected function: VerbarrasSNMenu
    function VerbarrasSNMenuSelected(app, event)
        global nro_barras ...
        barras_i_f ...
        coordenadas_x_y ...
        JB JE XB YB XE YE vect_x vect_y

```

```

hold(app.UIAxes_modelofinal,'on');
for i=1:nro_barras
JB=barras_i_f(i,1);
JE=barras_i_f(i,2);
XB=coordenadas_x_y(JB,1);
YB=coordenadas_x_y(JB,2);
XE=coordenadas_x_y(JE,1);
YE=coordenadas_x_y(JE,2);
vect_x=[XB XE]';
vect_y=[YB YE]';
plot(app.UIAxes_modelofinal,vect_x,vect_y,"LineWidth",2);
hold(app.UIAxes_modelofinal,'on');
end
hold(app.UIAxes_modelofinal,'off');
%% ACTIVAR SIGUIENTE PESTAÑA: VER ID DE BARRAS:
app.VerIDdebarrasMenu.Enable="on";
end

% Menu selected function: VerIDdebarrasMenu
function VerIDdebarrasMenuSelected(app, event)
% ETIQUETA ID DE CADA BARRA:
global nro_barras barras_i_f ... %Cantidad de barras y matriz de
conectores de barras
coordenadas_x_y offset_graf
for i=1:nro_barras
indice_nodo_i=barras_i_f(i,1);
indice_nodo_j=barras_i_f(i,2);
% Coordenadas del nodo inicial:
abs_i=coordenadas_x_y(indice_nodo_i,1);
ord_i=coordenadas_x_y(indice_nodo_i,2);
% Coordenadas del nodo final:
abs_f=coordenadas_x_y(indice_nodo_j,1);
ord_f=coordenadas_x_y(indice_nodo_j,2);
% Longitud de la i-ésima barra por pitagorazo:
L_barra_i=sqrt((ord_f - ord_i)^2 + (abs_f - abs_i)^2); %Usar para la
matriz donde necesita longitud de elementos
% Ángulo de giro:

```

```

    angulo_de_giro=rad2deg(atan((ord_f - ord_i)/(abs_f - abs_i)));
    %Abscisa de punto medio:
    abs_pto_medio=(abs_f + abs_i)/2;
    %Ordenada de punto medio:
    ord_pto_medio=(ord_f + ord_i)/2;
    % Nombre de la barra:
    id_bar=num2str(i);
    nombre_barra=strjoin(['Barra'},id_bar]);
    if angulo_de_giro>0
        text(app.UIAxes_modelofinal,abs_pto_medio-
1.5*offset_graf,ord_pto_medio+1.5*offset_graf,...
        nombre_barra,...
        'HorizontalAlignment',"center",...
        "Color",'b',...
        "FontSize",25,...
        'Rotation',angulo_de_giro);
    else
        text(app.UIAxes_modelofinal,abs_pto_medio+1.5*offset_graf,ord_pto_medio+1.5*offset_graf,...
        nombre_barra,...
        'HorizontalAlignment',"center",...
        "Color",'b',...
        "FontSize",25,...
        'Rotation',angulo_de_giro);
    end
end
end
% [...] Solicitar código completo a: olympicmaths20@gmail.com

```