



UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA
La Universidad Católica de Loja

ESCUELA DE INGENIERÍA CIVIL

**USO DE OPENSEES PARA GENERACIÓN DE SECCIONES
CON FIBRAS PARA ELEMENTOS (VIGA, COLUMNA) PARA
EL ANÁLISIS NO-LINEAL DE UNA ESTRUCTURA.**

TESIS DE GRADO PREVIA A LA
OBTENCIÓN DEL TÍTULO DE
INGENIERO CIVIL

AUTOR:

RÓMMEL VLADIMIR MERCHÁN SALINAS

DIRECTOR:

Dr. VINICIO A. SUAREZ CH.

**LOJA – ECUADOR
2010**

CERTIFICACIÓN

Dr. Vinicio Suarez Chacón
DIRECTOR DE TESIS

CERTIFICO:

Que he dirigido la tesis realizada por el Sr. Rómmel Vladimir Merchán Salinas previa a la obtención del título de Ingeniero Civil con el tema “**Uso de OpenSees para generación de secciones con fibras para elementos (viga, columna) para el análisis no-lineal de una estructura**”, la misma que tiene la suficiente validez técnica y profundidad investigativa, así como el cumplimiento de la reglamentación requerida por parte de la Escuela de Ingeniería Civil.

Loja, Noviembre de 2010

Dr. Vinicio Suárez Chacón
DIRECTOR DE TESIS

AUTORÍA

El presente proyecto de tesis “*Uso de OpenSees para generación de secciones con fibras para elementos (viga, columna) para el análisis no-lineal de una estructura*”, con cada una de sus observaciones, comentarios, análisis, diseños, verificaciones, conclusiones y recomendaciones emitidas, son de absoluta responsabilidad del autor.

Además, es necesario indicar que la información de otros autores empleada en el presente trabajo está debidamente especificada en el apartado bibliográfico.

Rómmel Vladimir Merchán Salinas

CESIÓN DE DERECHOS

Yo, Rómmel Vladimir Merchán Salinas, declaro ser autor del presente trabajo y eximo expresamente a la Universidad Técnica Particular de Loja y a sus representantes legales de posibles reclamos o acciones legales.

Adicionalmente declaro conocer y aceptar la disposición del Art. 67 del Estatuto Orgánico de la Universidad Técnica Particular de Loja que en su parte pertinente textualmente dice: *“Forman parte del patrimonio de la Universidad la propiedad intelectual de investigaciones, trabajos científicos o técnicos y tesis de grado que se realicen a través o con el apoyo financiero académico o institucional (operativo) de la Universidad”*

Rómmel Vladimir Merchán Salinas

AGRADECIMIENTO

Agradezco en primer lugar a Dios y a mis padres, quienes me dieron la vida y apoyaron en todo momento y lugar, me formaron y criaron como una persona capaz, pero sobretodo sensible.

A mis hermanos, familiares y amigos más cercanos quienes me dan la alegría de compartir y valorar las pequeñas cosas, que me han hecho crecer como ser humano.

Al Dr. Vinicio Suárez, director de esta tesis, para mi es un honor haber realizado este trabajo bajo su dirección y le estaré siempre muy agradecido porque ha dedicado su valioso tiempo a ello.

Deseo mostrar mi más sincero agradecimiento a todas aquellas personas que durante esta etapa de mi vida me supieron guiar y sin las cuales, este trabajo no habría visto la luz.

Rómmel

DEDICATORIA

A ti Dios que me diste la oportunidad de vivir.

Con mucho cariño principalmente a mis padres Rómmel y Carmen que me dieron la vida, por creer en mí y en este momento darme una carrera para mi futuro, aunque hemos pasado momentos difíciles siempre han estado brindándeme su amor, son ellos a quien les debo todo.

A mis hermanos, que han compartido sus aventuras que solo entre hermanos se pueden vivir, Viviana mi hermana menor, de carácter fuerte pero que me ha demostrado su amor inigualable, y por supuesto a Diego, el mayor, por estar siempre presente y enseñarme que con perseverancia se puede conseguir lo deseado.

A Paulina, mi novia, por ser alguien muy especial en mi vida y demostrarme que en todo momento cuento con ella. Por haber depositado su entera confianza en cada reto que se me presentaba sin dudar ni un solo momento en mi capacidad y por ayudarme a cumplir mis objetivos como persona y estudiante.

A mis familiares más cercanos que tuvieron una palabra de apoyo para mí durante mis estudios y a mis amigos por esos momentos tan especiales que han pasado a lo largo de mi vida.

Por eso, este trabajo que me llevó alrededor de un año culminarlo es para ustedes. Recuerden que siempre los llevaré en mi corazón.

Rómmel

INDICE DE CONTENIDOS

CERTIFICACIÓN.....	I
AUTORÍA.....	II
CESIÓN DE DERECHOS.....	III
AGRADECIMIENTO	IV
DEDICATORIA	V
INDICE DE CONTENIDOS.....	VI
ÍNDICE DE FIGURAS	IX
RESUMEN	XI
CAPÍTULO I	1
INTRODUCCIÓN	1
1. INTRODUCCIÓN	1
1.1. INTRODUCCIÓN	1
1.2. JUSTIFICACIÓN.....	1
1.3. OBJETIVOS	2
1.3.1. Objetivo General	2
1.1.1. Objetivos específicos.....	2
1.2. ALCANCE Y ORGANIZACIÓN	2
CAPÍTULO II	4
GENERALIDADES DEL CURSO	4
2. GENERALIDADES DEL CURSO	4
2.1. INTRODUCCIÓN A OPENSEES	4
2.1.1. Capacidades de OpenSees	5
2.1.1.1. Modelamiento.....	5
2.1.1.2. Análisis.....	6
2.2. INTRODUCCIÓN AL LENGUAJE tcl.....	8
2.2.1. Formato de los comandos Tcl	9
2.2.2. Definición de variables y evaluación de expresiones matemáticas.....	10
2.2.3. Procedimientos	11
2.2.3.1. Condicional.....	11
2.2.3.2. Bucle for	12
2.3. PRE-PROCESAMIENTO, ANÁLISIS Y POST-PROCESAMIENTO DE DATOS.....	13
2.4. EJEMPLO. PERÍODO DE VIBRACIÓN FUNDAMENTAL	13

CAPÍTULO III	17
ANÁLISIS ELÁSTICO DE ESTRUCTURAS	17
3. ANÁLISIS ELÁSTICO DE ESTRUCTURAS EN 3D	17
3.1. EJEMPLO. ANÁLISIS ELÁSTICO 3D DE HISTORIA EN EL TIEMPO.	17
CAPÍTULO IV	29
SECCIONES EN OPENSEES.....	29
4. SECCIONES.....	29
4.1. SECCIONES CON FIBRAS	29
4.2. GENERACIÓN DE SECCIONES EN OPENSEES.....	30
4.2.1. Sección Elástica	30
4.2.2. Sección Uniaxial.....	31
4.2.3. Sección de Fibras.....	31
4.3. AGREGADOR DE SECCIONES.....	36
4.3.1. Ejemplo. Análisis Momento-Curvatura.	38
CAPÍTULO V	53
SECCIONES Y ELEMENTOS DE LONGITUD “0”	53
5. SECCIONES Y ELEMENTOS DE LONGITUD “cero”	53
5.1. INTRODUCCIÓN	53
5.2. ELEMENTOS DE LONGITUD “0”	55
5.2.1. Comando: ZERO-LENGTH ELEMENT	55
5.2.2. Comando: ZERO-LENGTH ND ELEMENT	56
5.2.3. Comando: ZERO-LENGTH SECTION ELEMENT	57
5.3. EJEMPLO. CREACIÓN DE ELEMENTOS DE LONGITUD CERO.....	58
CAPÍTULO VI	68
ANÁLISIS NO-LINEAL DE	68
ESTRUCTURAS APORTICADAS	68
6. PÓRTICOS EN EDIFICIOS Y PUENTES.....	68
6.1. INTRODUCCIÓN	68
6.2. PÓRTICOS EN EDIFICIOS	70
6.2.1. Ejemplo. Análisis pushover 3D de un pórtico de 3 pisos con 2 vanos en ambas direcciones.	70
6.3. PÓRTICOS EN PUENTES	97
6.3.1. Definición y elementos de un puente	97
6.4. EJEMPLO. ANÁLISIS NO LINEAL DE UN PUENTE DE 3 VANOS Y LUZ 90M.	98

CAPÍTULO VII	122
ANÁLISIS INELÁSTICO BAJO	122
CARGAS DINÁMICAS.....	122
7. ANÁLISIS DINÁMICOS.....	122
7.1. INTRODUCCIÓN	122
7.2. ANÁLISIS DINÁMICOS EN OPENSEES.....	123
7.3. EJEMPLO. ANÁLISIS DINÁMICOS DE UN PÓRTICO EN 2D.....	126
CAPÍTULO VIII	150
OPENSEES EN PARALELO	150
8. OPENSEES EN PARALELO.....	150
8.1. INTRODUCCIÓN	150
8.2. COMPUTADORES EN PARALELO.....	150
8.3. MODELOS DE PROGRAMACIÓN EN PARALELO	152
8.4. INTERPRETES DE OPENSEES EN PARALELO	153
8.5. OPENSEES SP	153
8.5.1. ¿Qué ocurre cuando se ejecuta OpenSeesSP?.....	153
8.6. COMANDOS MODIFICADOS	155
8.7. PASOS PARA LA EJECUCIÓN EN PARALELO	155
CONCLUSIONES:	157
RECOMENDACIONES:	158
BIBLIOGRAFÍA:.....	159

ÍNDICE DE FIGURAS

FIGURA 1 – Componentes para el análisis en OpenSees	8
FIGURA 2 – Interpretador Tcl de OpenSees	9
FIGURA 3 – Columna a modelar	14
FIGURA 4 – Geometría del pórtico 3D	19
FIGURA 5 – Definición de los extremos rígidos.....	23
FIGURA 6 – Definición del sistema local de coordenadas.....	25
FIGURA 7 – Análisis de resultados (Desplazamiento vs Tiempo).....	27
FIGURA 8 – Análisis de resultados (Deriva vs Tiempo)	28
FIGURA 9 – Esfuerzo Deformación del concreto.....	30
FIGURA 10 – Sección circular con fibras.....	32
FIGURA 11 – Comando fibra.....	33
FIGURA 12 – Características del comando “patch quad”	34
FIGURA 13 – Características del comando “patch circ”	35
FIGURA 14 – Características del comando “layer straight”	35
FIGURA 15 – Características del comando “layer circ”	36
FIGURA 16 – Agregador de secciones 1	37
FIGURA 17 – Agregador de secciones 2	37
FIGURA 18 – Gráfica momento – curvatura sección rectangular con Opensees	47
FIGURA 19 – Gráfica Momento – Curvatura sección circular con Opensees.....	47
Figura 20 – Entorno de RC-Analysis para sección rectangular	48
FIGURA 21 – Gráfica momento – curvatura sección rectangular con RC-Analysis	49
Figura 22 – Entorno de RC-Analysis para sección circular	¡Error! Marcador no definido.
FIGURA 23 – Gráfica Momento – Curvatura sección circular obtenida con RC-Analysis.....	51
Figura 24 - Comparación Momento-Curvatura sección rectangular	51
Figura 25 - Comparación Momento-Curvatura sección circular	52
FIGURA 26 – Regiones inelásticas previstas en columnas y pared.	54
FIGURA 27 – Ubicación de los elementos y secciones de longitud cero.....	55
FIGURA 28 – Geometría del modelo	59
FIGURA 29 – Ejecución del ejemplo6.tcl	64
FIGURA 30 – Historia de desplazamiento nudo 4 en dirección X.....	64
FIGURA 31 – Momento en Viga extremo i	65
FIGURA 32 – Cortante en columna 3 extremo i	66
FIGURA 33 – Histéresis del material del elemento 6	66
FIGURA 34 – Solicitaciones horizontales y verticales en una estructura	69
FIGURA 35 – Vista en planta del modelo	71
FIGURA 36 – Elevación plano XZ del modelo	72
FIGURA 37 – Vista 3D del modelo	72
FIGURA 38 – Sección transversal del elemento	80
FIGURA 39 – Orientación de los elementos	81
FIGURA 40 – Vector paralelo al plano XZ	81
FIGURA 41 – elemento beamWithHinges	82
FIGURA 42 - Puntos de integración	83
FIGURA 43 – Gráfica Pushover	92

FIGURA 44 – Esfuerzo en fibra superior izquierda	93
FIGURA 45 – Esfuerzo en fibra inferior izquierda.....	93
FIGURA 46 – Esfuerzo en fibra superior derecha.....	94
FIGURA 47 – Esfuerzo en fibra inferior derecha	94
FIGURA 48 – Momento en el extremo izquierdo	95
FIGURA 49 – Curvatura en el extremo izquierdo	95
FIGURA 50 – Momento en el extremo derecho.....	96
FIGURA 51 – Curvatura en el extremo derecho	96
FIGURA 52 – Elementos de un puente	98
FIGURA 53 – Elevación plano XZ del modelo	99
FIGURA 54 – Nudos y elementos	99
FIGURA 55 – Vista 3D del modelo	100
FIGURA 56 – Efectos P-Delta	106
FIGURA 57 – Registro de aceleración en dirección X	111
FIGURA 58 – Registro de aceleración en dirección Y	112
FIGURA 59 – Historia de desplazamiento en dirección X del nudo 2.....	116
FIGURA 60 – Historia de desplazamiento en dirección Y del nudo 2.....	116
FIGURA 61 – Historia de desplazamiento en dirección X del nudo 9.....	117
FIGURA 62 – Historia de desplazamiento en dirección Y del nudo 9.....	117
FIGURA 63 – Fuerza axial Fx en el nudo i de la Pila 3.....	118
FIGURA 64 – Cortante Fy en el nudo i de la Pila 3.....	119
FIGURA 65 – Momento Mz en el nudo i de la Pila 3	119
FIGURA 66 – Deformación en extremo i de cada sección de la Pila 4	120
FIGURA 67 – Deformación en extremo j de cada sección de la Pila 4	120
FIGURA 68 – Curvatura en extremo i de cada sección de la Pila 4	121
FIGURA 69 – Curvatura en extremo j de cada sección de la Pila 4	121
FIGURA 70 – Análisis dinámico bajo excitación sísmica uniforme.....	123
FIGURA 71 – Análisis dinámico bajo excitación sísmica bidireccional.....	124
FIGURA 72 – Análisis dinámico bajo excitación sísmica múltiple	124
FIGURA 73 – Análisis dinámico bajo excitación senoidal.....	125
FIGURA 74 – Análisis dinámico bajo excitación senoidal múltiple	125
FIGURA 75 – Geometría del modelo	126
FIGURA 76 – Historia de desplazamiento en dirección X nudo 3, análisis 1	147
FIGURA 77 – Historia de desplazamiento en dirección X nudo 3, análisis 2	147
FIGURA 78 – Historia de desplazamiento en dirección X nudo 3, análisis 3	148
FIGURA 79 – Historia de desplazamiento en dirección X nudo 3, análisis 3	148
FIGURA 80 – Historia de desplazamiento en dirección X nudo 3, análisis 3	149
FIGURA 81 – Comparación de desplazamientos entre elementos tipo 1 y tipo 2	149
FIGURA 82 – Microp procesadores de memoria compartida	151
FIGURA 83 – Memoria distribuida MPPs.....	151
FIGURA 84 – Memoria distribuida SMPs – Hybrid Systems	152
FIGURA 85 – Sistemas Cluster	152
FIGURA 86 – Interprete OpenSeesSP antes del análisis.....	154
FIGURA 87 – Interprete OpenSeesSP después del análisis	154

RESUMEN

En la actualidad se hace indispensable el uso de software para el diseño, simulación y análisis de cualquier tipo de obra civil, y más aún cuando se trata de construcciones de gran envergadura en donde surge la necesidad de lograr que la simulación sea en lo posible tan cercana a la realidad para asegurar la permanencia de la obra. Por esta entre otras razones nace OpenSees, un software de código abierto, gratuito, creado para simular el comportamiento de sistemas estructurales y geotécnicos bajo el ataque de sismos.

Por ello el Laboratorio de Simulación Computarizada de la UCG ha adoptado OpenSees como su principal herramienta de simulación, y ha creído conveniente la creación de un Curso Básico de OpenSees, de tipo virtual, disponible en <http://eva.utpl.edu.ec/vlee/>, para que estudiantes de habla hispana se capaciten en el manejo y aplicación de esta herramienta para su uso académico en proyectos de investigación, y porque no en proyectos reales.

Así, este trabajo está orientado a mejorar y completar el Curso, para lo cual se seguirá la misma estructura. Por tanto este documento presenta varios ejemplos en código Tcl, explicados paso a paso de modelos de estructuras en 2D y 3D desde una columna en cantiliver hasta un pórtico de un puente, sobre los cuales se ejecutarán varios tipos de simulaciones tales como: Análisis estático elástico, análisis modal, análisis no-lineal estático y análisis no-lineal dinámico. Posterior a ello se analizarán los modelos e interpretarán los resultados mediante gráficas obtenidas al procesarlos con ayuda de Excel.

El documento termina presentando en su último Capítulo unas breves generalidades sobre el uso de OpenSees en paralelo y de sus intérpretes para trabajar en este entorno: OpenSeesSP y OpenSeesMP; de los cuales solamente el primero se estudiará en este trabajo.

El trabajo concluye con la actualización y carga del contenido de este documento al Entorno Virtual de Aprendizaje “EVA” del Laboratorio Virtual de Ingeniería Sísmica (<http://eva.utpl.edu.ec/vlee/>) donde se encuentra disponible el Curso Básico de OpenSees.

CAPÍTULO I

INTRODUCCIÓN

1. INTRODUCCIÓN

1.1.INTRODUCCIÓN

Este trabajo de investigación consiste en mejorar y completar el Curso Virtual Básico de OpenSees, el cual está dirigido a los estudiantes y profesionales de Ingeniería Civil que deseen aprender a manejar el software OpenSees. El curso contiene 13 unidades de estudio que deben ser cubiertas en orden sucesivo. Cada unidad contiene información escrita de acuerdo al tema a tratar, ejemplos explicados paso a paso, foros y otros recursos para uso de las personas que accedan al curso. Adicionalmente, estarán cargados cuestionarios y tareas que las personas matriculadas en el curso deberán realizar y obtener una calificación satisfactoria en cada unidad para su aprobación. El tiempo requerido por los estudiantes para culminar el curso varía en función de su dedicación y conocimientos previos, sin embargo se ha estimado un tiempo promedio de 80 horas.

1.2.JUSTIFICACIÓN

OpenSees es el Sistema Abierto para Simulación en Ingeniería Sísmica desarrollado en el Pacific Earthquake Research Center (PEER) (<http://peer.berkeley.edu/>) de la Universidad de California en Berkeley, USA. OpenSees es un software de código abierto, gratuito, de uso académico, especialmente creado para simular el comportamiento de sistemas estructurales y geotécnicos bajo el ataque de sismos. Esta

herramienta es muy potente, por esta razón es usada en las mejores universidades del mundo en investigaciones en el campo de la Ingeniería Sísmica. Así mismo el Laboratorio de Simulación Computarizada de la UCG ha adoptado OpesSees como su principal herramienta de simulación.

Con estos antecedentes, este trabajo adquiere mucha importancia ya que posibilita el que estudiantes de habla hispana se capaciten en el manejo y aplicación de esta herramienta para su uso académico en proyectos de investigación, y porque no en proyectos reales.

1.3.OBJETIVOS

1.3.1. Objetivo General

Actualizar y completar el Curso Virtual Básico de OpenSees, disponible en el Laboratorio Virtual de Ingeniería Sísmica (<http://eva.utpl.edu.ec/vlee/>) de la UCG de la UTPL, mediante la creación de archivos de entrada en código Tcl.

Creación de archivos de entrada Tcl que servirán para completar el Curso Virtual Básico de OpenSees, disponible en el Laboratorio Virtual de Ingeniería Sísmica (<http://eva.utpl.edu.ec/vlee/>) de la UCG de la UTPL.

1.1.1. Objetivos específicos

- Idealizar y generar modelos de estructuras reticulares (columnas, vigas, pórticos planos, edificios 3D, puentes) utilizando correctamente los comandos tcl de OpenSees.
- Ejecutar varios tipos de simulaciones tales como: Análisis estático elástico, análisis modal, análisis no-lineal estático y análisis no-lineal dinámico sobre los modelos generados.
- Analizar los modelos creados usando correctamente los comandos tcl de OpenSees.
- Pos-procesamiento e interpretación correcta de los resultados de la simulación.
- Actualizar los cuestionarios y contenido en general que ya se encuentra cargado del Curso Básico de OpenSees.

1.2.ALCANCE Y ORGANIZACIÓN

Con la realización de esta investigación se busca completar el Curso Virtual Básico de OpenSees, mediante la elaboración de ejemplos explicados paso a paso de modelos de estructuras basados en comandos Tcl, sobre los que se ejecutarán varios tipos de simulaciones, y por último, se analizará e interpretará los resultados.

La estructura del documento se ha dispuesto en 8 capítulos descritos a continuación: en el Capítulo uno, se presenta la descripción general del proyecto. En el Capítulo dos, se presenta lo que sería la introducción misma del Curso Virtual, además a través de un pequeño ejemplo se muestra la estructura y pasos a seguir para crear los archivos de entrada en código Tcl. En el Capítulo tres, se detalla más a fondo lo indicado en el ejemplo del capítulo anterior mediante el análisis elástico de historia en el tiempo de una estructura en 3D, formada por elementos elásticos tanto para las vigas como para las columnas. En el Capítulo cuatro, se muestra las posibilidades que tiene OpenSees para la creación de secciones con fibras y se presenta un ejemplo en el que se realiza un análisis momento-curvatura sobre una sección de fibras de una columna rectangular y una circular. En el Capítulo cinco, se indica como crear secciones y elementos de longitud cero y a través de un ejemplo de un pórtico en 2D sencillo como usarlos. En el Capítulo seis, se presenta una introducción referente a los análisis inelásticos de estructuras aporticadas en edificios y puentes, a continuación se presentan dos ejemplos, el primero, un pórtico 3D de un edificio en el que se realizará un análisis pushover, y el segundo un pórtico 3D de un puente, estas estructuras a diferencia de los ejemplos anteriores serán creadas con elementos inelásticos. El Capítulo siete continúa con los análisis inelásticos, pero esta vez dinámicos, se indica los cinco posibles análisis dinámicos que se pueden simular en OpenSees, y se realiza un ejemplo en el que se muestra la aplicación de todos ellos. Por último, en el Capítulo ocho se presentan unas breves generalidades sobre el uso de OpenSees en computadores en paralelo, específicamente a OpenSees SP.

El trabajo concluye con la actualización y carga del contenido de este documento al Entorno Virtual de Aprendizaje “EVA” del Laboratorio Virtual de Ingeniería Sísmica (<http://eva.utpl.edu.ec/vlee>) donde se encuentra disponible el Curso Básico de OpenSees. Adicional a esto se creará tareas y cuestionarios para cada uno de los ejemplos presentados, con los que se calificará a los estudiantes que tomen el curso.

CAPÍTULO II

GENERALIDADES DEL CURSO

2. GENERALIDADES DEL CURSO

2.1. INTRODUCCIÓN A OPENSEES

En la actualidad se hace indispensable el uso de software para el diseño, simulación y análisis de cualquier tipo de obra civil, y aún más cuando se trata de construcciones de gran envergadura en donde surge la necesidad de lograr que la simulación sea en lo posible tan cercana a la realidad para asegurar la permanencia de la obra. Por esta entre otras razones nace OpenSees, un software de código abierto, gratuito, creado para simular el comportamiento de sistemas estructurales y geotécnicos bajo el ataque de sismos.

La mayor Fortaleza de OpenSees radica en su capacidad para simular el comportamiento real de los materiales, secciones y elementos que componen las estructuras. Materiales como el concreto, acero, madera, suelos se comportan de manera no lineal – inelástica cuando se deforman. Para lograr esto, OpenSees brinda la posibilidad de creación de secciones de vigas o columnas, que pueden ser de tres tipos: Elástica, Uniaxial y de Fibras, donde esta última es sin duda una de las mejores características de OpenSees. Entre algunas de las ventajas de utilizar este tipo de sección se tiene:

- La incorporación automática de la interacción entre carga axial y momento, es decir la sección cambia instantáneamente su resistencia a flexión de acuerdo a la carga axial que se le aplica.
- El modelo captura de forma explícita los efectos de flexión biaxial.
- Puede tener cualquier forma y estar formada por fibras de varios materiales.
- Es posible monitorear la respuesta esfuerzo-deformación de cada fibra en la sección.

Se puede analizar estructuras y elementos geotécnicos en niveles como:

- Fuerza deformación del elemento.
- Análisis momento curvatura en secciones.
- Esfuerzos en fibras.

Las simulaciones de OpenSees pueden ejecutarse en sistemas de computación en paralelo para reducir el tiempo de simulación de grandes modelos o para la ejecución de estudios paramétricos.

OpenSees es muy flexible y permite que un mismo problema se resuelva de distintas maneras. El realismo de la simulación dependerá de la calidad del modelo y de las opciones seleccionadas para el análisis.

Por estas capacidades avanzadas para la modelación, análisis y tiempo de ejecución de la respuesta inelástica de sistemas utilizando una amplia gama de modelos de materiales, elementos y algoritmos de solución, se lo considera como una herramienta muy potente dentro de las investigaciones en el campo de la Ingeniería Sísmica.

2.1.1. Capacidades de OpenSees

Las capacidades de OpenSees están orientadas al modelamiento, análisis y fiabilidad estructural. Sin embargo en este proyecto solamente nos limitaremos a los dos primeros.

2.1.1.1. Modelamiento

Pueden crearse cualquier tipo de modelos estructurales y geotécnicos en dos o tres dimensiones, en donde cada componente puede ser descrito en varios niveles como: elementos, secciones o fibras. Un elemento puede ser definido de distintas formas, pueden ser elementos armados “element truss” y “corotational element truss”; como elementos elásticos “elasticBeamColumn”; elementos no lineales “nonlinearBeamColumn”, “beamWithHinges” y “dispBeamcolumn”; elementos de longitud cero “zeroLength”, “zeroLengthND” y “zeroLengthSection”; elementos cuadrilátero “element quad”, entre otros más. Una sección puede ser definida como una

sección elástica “elastic section”, sección uniaxial “uniaxial Section”, sección de fibras “Fiber Section”, sección elástica de placa de membrana “Elastic Membrane Plate Section”, sección de placa de fibras “Plate Fiber Section”, o una sección bidireccional “Bidirectional Section”.

Algunos comandos predefinidos pueden ser usados para modelar el tipo de material del que están formados los elementos de la estructura, por ejemplo, los materiales uniaxiales que incluye OpenSees son: elásticos; elásto – plásticos; materiales de concreto por ejemplo, concrete01 el cual crea un material de concreto sin resistencia a la tensión; materiales para modelar el acero de refuerzo como el steel01 el cual crea un modelo de acero bilineal; y materiales histeréticos. Los materiales ND incluyen materiales isotrópicos elásticos, materiales de esfuerzo planos, y materiales de fibras para placas.

Las cargas externas pueden aplicarse a los nudos o estar uniformemente distribuidas a lo largo de la longitud del elemento usando cierto número de comando predefinidos. La excitación sísmica puede también puede simularse usando el comando “Series”. Se puede usar procedimientos para crear o convertir un registro de aceleración a formato de OpenSees. A estos registros se les puede aplicar varios factores de carga, además se pueden asignar más de un registro sísmico en más de un grado de libertad, y cada uno de ellos en los diferentes apoyos de la estructura.

Los resultados se pueden grabar en un archivo especificado por el usuario. El desplazamiento, velocidad, aceleración, fuerzas, esfuerzos, deformación, entre otros pueden ser monitoreados conforme avanza el análisis. Los datos contenidos en estos archivos pueden exportarse a Excel por ejemplo para procesarse y obtener curvas con información gráfica de lo que ocurrión durante la ejecución del análisis.

2.1.1.2. Análisis

Los análisis que se pueden realizar en OpenSees van desde un simple análisis lineal estático a un análisis no lineal transitorio.

El núcleo de OpenSees es la ejecución del análisis, el cual está enmarcado dentro de los parámetros siguientes: manejo de las restricciones, número de grados de libertad del modelo, tipo de análisis a ejecutar, integrador de cargas, algoritmos de solución, y el sistema de ecuaciones para la solución.

El manejo de las restricciones determina como manejar las restricciones impuestas a los grados de libertad en los nudos. Por ejemplo la definición de un “constraint Plain” solo sirve cuando las restricciones en los nudos no vincular sus desplazamientos a otros nudos, cuando se trata de modelos de edificaciones donde se generan diafragmas de piso se usa “constraint Transformation” para condensar estáticamente la matriz de rigidez,

otros tipos que se pueden usar son “Penalty method”, “Lagrange Multipliers”, “Transformation Method”.

El indicar el número de grados de libertad determina un mapeo entre el número de los grados de libertad y las ecuaciones que se deberán obtener para resolver el problema. Cuando se usa “numberer Plain” se está especificando el algoritmo que se usará para numerar los grados de libertad en el modelo, esta opción asigna de forma arbitraria los grados de libertad a los nudos, mientras que la opción “numberer RCM” asigna los grados de libertad a los nudos usando el algoritmo “Reverse Cuthill – McKee”.

Dependiendo del tipo de análisis se establece mediante el comando “analysis” el tiempo que durará el mismo. Además del tipo, un análisis estático se resuelve sin masa y matrices de amortiguamiento, para un análisis transitorio si se considera esto y se resuelve a pasos constantes de tiempo.

El integrador determina el número de pasos del análisis y depende de si es estático o transitorio. En un estático el integrador puede ser de tipo “load control” (carga controlada), “displacement control” (desplazamiento controlado), “mínimum unbanded displacement norm”, o “arc-length control”. Para un análisis transitorio el integrador puede ser el método de Newmark o el método de Hibert-Hughes-Taylor.

El algoritmo de solución indica el método a usar para alcanzar la convergencia cuando se avanza paso a paso en la solución de un problema no lineal. Existen varias opciones: Lineal, Newton, Newton con rigidez inicial, Newton modificado, Krylov-Newton, entre otros.

El sistema de ecuaciones especifica el algoritmo utilizado para resolver el sistema de ecuaciones lineales en el análisis. BandGeneral o ProfileSPD funcionan bien en la mayoría de casos. Pero hay también otras opciones: Sparse General, UmfPack, Sparse SPD.

En la siguiente figura se muestra una tabla resumen los componentes usados en OpenSees para la ejecución de un análisis.

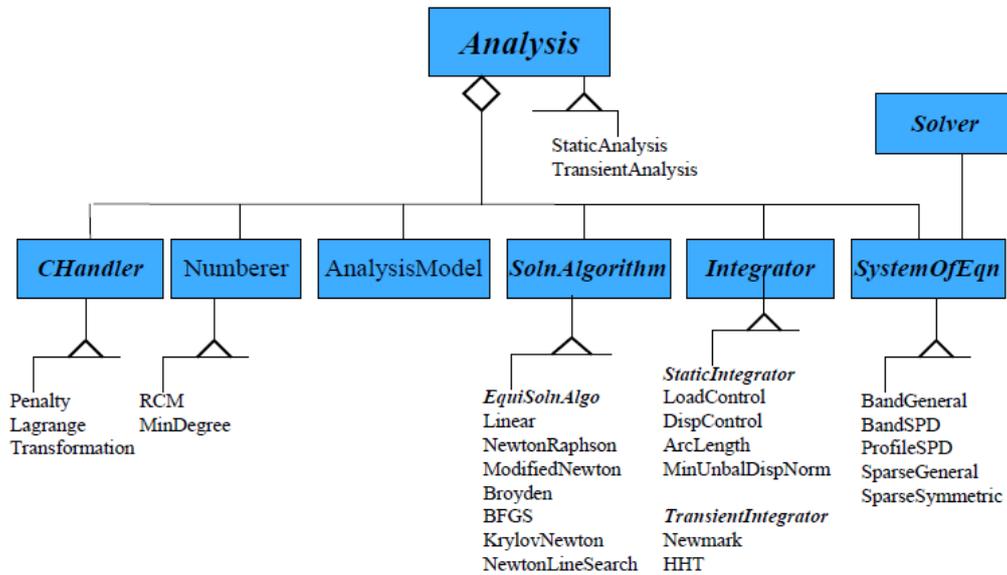


FIGURA 1 – Componentes para el análisis en OpenSees

2.2.INTRODUCCIÓN AL LENGUAJE tcl

El lenguaje de programación Tcl “Tool Command Language” es un lenguaje de programación dinámico muy poderoso y fácil de aprender, adecuado para un gran rango de usos incluyendo aplicaciones web y de escritorio, redes, administración pruebas y más. De código abierto, de fácil implementación y altamente extensible. Es por esto que fue escogido por los creadores de OpenSees como base para la implementación de comandos para la generación de modelos y análisis estructural. Es decir, los comandos de OpenSees se ejecutan desde un interpretador Tcl junto a comandos propios del lenguaje Tcl. El interpretador Tcl de OpenSees es la ventana que se abre cuando se ejecuta opensees.exe. Esta ventana se muestra en la siguiente figura:

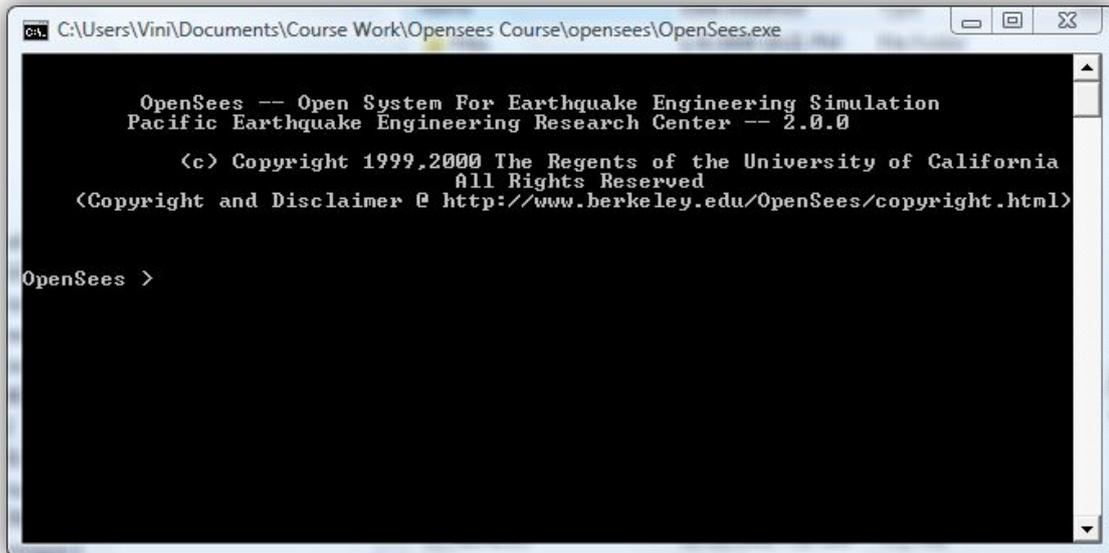


FIGURA 2 – Interpretador Tcl de OpenSees

Tcl es un lenguaje de programación tipo “script”, es decir, cuando escribimos un comando en el interpretador Tcl y presionamos ENTER, el comando se ejecuta en ese momento sin requerir compilación. En este sentido Tcl es similar a los lenguajes de programación incluidos en las calculadoras programables o a Matlab. Tcl soporta las siguientes funciones básicas:

- Uso de variables y sustitución de variables
- Evaluación de variables matemáticas
- Estructuras de control lógico y bucles
- Procedimientos
- Manipulación de archivos

Además de otras con las que se pueden crear aplicaciones variadas. Información sobre Tcl se puede encontrar en www.tcl.tk y otros sitios web que contienen tutoriales y documentación (<http://tmml.sourceforge.net/doc/tcl/index.html>). Tcl es gratuito.

2.2.1. Formato de los comandos Tcl

El código escrito en Tcl está formado de una serie de comandos separados por líneas o el símbolo “;”. La sintaxis básica de un comando Tcl es:

comando \$arg1 \$arg2

Donde:

comando es un comando tcl (o de OpenSees)
\$arg1 y \$arg2 son los argumentos requeridos por el comando.

Tcl admite que los argumentos sean comandos anidados:

comando [comando \$arg1 \$arg2] [comando \$arg1 \$arg2]

Donde:

[] delimitan los comandos anidados. El intérprete Tcl primero evalúa los comandos anidados y luego evalúa los comandos exteriores con los resultados de los comandos anidados.

A continuación se presentan comandos e instrucciones Tcl básicos a saber por un usuario de OpenSees. Así mismo, en el transcurso de este proyecto se irán presentando más comandos que a su tiempo se explicarán.

2.2.2. Definición de variables y evaluación de expresiones matemáticas

Para definir variables se utiliza el comando “set”. Por ejemplo:

set a 8

Da a la variable **a** el valor **8**. Si se desea agregar un comentario se usa el símbolo “#”, el comentario puede colocarse en una línea diferente a la línea en que se ejecuta un comando, o en la misma línea, al hacerlo en la misma línea se debe colocar el símbolo “;” para separar el comentario del comando, por ejemplo:

**# este comando asigna el valor de 8 a la variable a
set a 8**

ó también:

set a 8; # este comando asigna el valor de 8 a la variable a

Si luego se escribe **expr \$a** ENTER, el intérprete devuelve **8**. Cuando delante de una variable se escribe el signo \$, Tcl utiliza el valor contenido en la variable. El comando **expr** permite la evaluación de expresiones matemáticas. La sintaxis general es:

expr expresión_matemática

Así mismo si se escribe:

set b 4

luego presionamos ENTER y a continuación escribimos el comando:

expr \$a+\$b

al presionar nuevamente **ENTER**, retorna el valor de **12** en la ventana del interpretador.

2.2.3. Procedimientos

En Tcl es posible encapsular código dentro de un procedimiento. La sintaxis general de un procedimiento es:

```
proc $nombre { $arg1 $arg2....} {  
  código que usa $arg1 $arg2..... y que se ejecuta cuando se llama al  
procedimiento  
  return [ $resultado]  
}
```

Donde **proc** es el comando que inicializa el procedimiento y **return** es el comando que devuelve el resultado del procedimiento. Por ejemplo, el siguiente procedimiento calcula el área de una sección circular:

```
proc areacircular { dia } {  
  return [ expr 3.1416*$dia**2/4]  
}
```

La exponenciación se logra con el operador ****** cuando el exponente es entero. Alternativamente se puede usar el comando **pow(\$numero,\$exponente)**. Una vez definido el procedimiento, el área de una sección circular puede ser calculada en cualquier parte del código llamando al procedimiento. Por ejemplo el área de una barra de acero de 14mm se calcula con:

```
areacircular 14
```

Obteniendo el resultado 153.9384. La definición de procedimientos es muy útil cuando existen cálculos o estructuras lógicas que se repiten constantemente en el código.

2.2.3.1. Condicional

El comando **if** se utiliza en Tcl para ejecutar código en forma condicional. La sintaxis es:

```
If {prueba lógica} {  
  Código que se ejecuta si prueba es verdadera  
} else {  
  Código que se ejecuta si prueba es falsa
```

```
}
```

Donde la prueba lógica consiste en comparar dos variables usando operadores lógicos como: < > <= >=, etc. Por ejemplo, el código a continuación calcula la cuantía de acero en una sección circular de columna y reporta en pantalla un mensaje si la cuantía excede 4%. En este ejemplo se utiliza el procedimiento “**areacircular**”, definido anteriormente.

```
set Dcolumna 1000 ;# Diámetro de la sección en mm
set Dvarilla 20 ;# Diámetro del refuerzo principal en mm
set nvarillas 30 ;# Número de varillas en la sección
set cuantia [expr $nvarillas*[areacircular $Dvarilla]/ [areacircular
$Dcolumna] ]
if {$cuantia>0.04} {
puts “La cuantía es: $cuantia > 0.04”
} else {
puts “La cuantía es: $cuantia < 0.04”
}
```

En este código se han introducido comentarios utilizando el signo # y el signo; como separador de líneas. También se ha usado la instrucción **puts** que imprime texto en pantalla.

2.2.3.2. Bucle for

Estas funciones permiten que secciones de un código se repitan ejecutándose para distintos valores de una variable. La sintaxis de la función for es:

```
for {define valor inicial de la variable} {define valor final de la variable}
{define el valor del incremento} {
código que se ejecuta para cada valor de la variable
}
```

Por ejemplo, el código que se presenta a continuación incrementa el valor de la variable \$i de 1 a 10 y lo presenta en pantalla.

```
for {set i 1} {$i < 10} {incr i 1} {
puts “i es $i”
}
```

2.3.PRE-PROCESAMIENTO, ANÁLISIS Y POST-PROCESAMIENTO DE DATOS

Para preparar los archivos de entrada se puede utilizar cualquier editor de texto, sin embargo, se recomienda utilizar:

- notepad++ (<http://notepad-plus.sourceforge.net/>),
- TcLeditor (<http://opensees.berkeley.edu/OpenSees/user/tools.php>), o
- el notepad de Windows ya que estos son editores de texto plano no introducen caracteres ocultos ni formato.

Notepad++ y TcLeditor son editores que reconocen los comandos y sintaxis TCL, el segundo es un editor especialmente desarrollado para OpenSees. Una vez que se ha preparado el archivo de entrada, este se guarda con extensión “.tcl” de preferencia en la misma carpeta donde se encuentra el ejecutable opensees.exe. Para ejecutar el código tcl contenido en el archivo de entrada de datos se utiliza la instrucción que se muestra a continuación desde la ventana del intérprete TCL de OpenSees:

```
source $nombreadchivodedatos.tcl
```

Si el nombre del archivo de entrada es ejemplo0.tcl, el usuario debe digitar:

```
source ejemplo0.tcl
```

y luego pulsar ENTER. Cuando el archivo de datos está en una carpeta distinta a la de opensees.exe, entonces el “\$nombreadchivodedatos.tcl” debe incluir la ruta correspondiente y el nombre del archivo, por ejemplo:

```
source C:/tcl/ejemplo0.tcl
```

Nótese que en tcl el símbolo “\” con el que normalmente se define la ruta de los archivos se substituye con “/”. Como resultado de la simulación realizada, y en correspondencia con el archivo de datos preparado, se generaran uno o más archivos de texto conteniendo los valores de respuesta obtenidos del análisis. Los archivos generados son de texto plano y pueden abrirse con un editor de texto o con programas como Matlab o Excel para su procesamiento.

2.4.EJEMPLO. PERIÓDO DE VIBRACIÓN FUNDAMENTAL

A continuación se presenta un ejemplo para el cálculo del periodo de vibración fundamental de una estructura en el que se indica los pasos básicos para la simulación de un modelo en OpenSees.

Paso 1: Idealización del modelo.

El modelo a simular es una columna de 3m de altura, se modelará como un elemento elástico, estará empotrada en su base y se le asignará masa en dirección X en la cabeza de la columna, tal como se muestra en la figura:

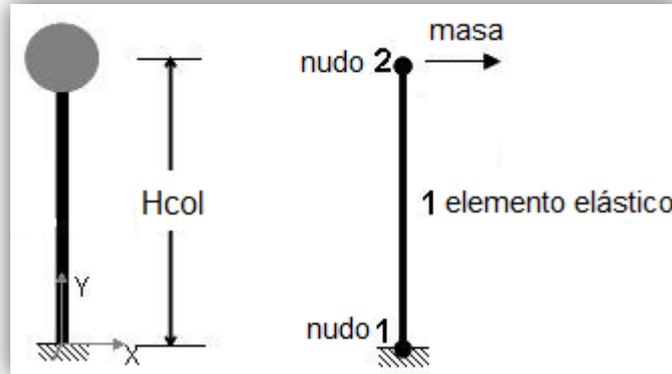


FIGURA 3 – Columna a modelar

Paso 2: Generación del modelo.

```
#Ejemplo0.tcl
#Periodo de vibracion fundamental de una estructura con un solo elemento elastico #
#Unidades: kN m s #
#####
wipe ;# Este comando borra todos los objetos existentes en el interpretador Tcl
model basic -ndm 2 -ndf 3 ;# 2 dimensiones; 3 Grados de Libertad (GDL) por nudo
# Se define la geometría
# Coordenadas de los nudos
# n x y
node 1 0 0
node 2 0 3

# Se empotra el nudo 1
# n 1 2 3
fix 1 1 1 1
# Se asigna una masa de 10 Ton (kN/g) al nudo 2 en la dirección X
# n 1 2 3
mass 2 10 1e-6 1e-6

# Definicion de Elementos
# Se crea un elemento elástico entre los nudos 1 y 2
# Al elemento elástico se le asigna un area de 0.16m2, un módulo de elasticidad # de
20000000 kPa
# y una inercia de 0.0021 m4
geomTransf Linear 1
#
# nele ni nj Area E I ngt
element elasticBeamColumn 1 1 2 0.16 20000000 0.0021 1
```

```
# A continuación usamos el comando eigen para obtener el primer valor propio #de la
estructura
# es decir la frecuencia al cuadrado del primer modo de vibración.
puts "El periodo fundamental de vibración es: [expr 2*3.1416/pow([eigen 1],0.5)]
segundos"
```

Paso 3: Ejecución del análisis.

Para la ejecución del análisis se abre el intérprete de OpenSees, se digita **source ejemplo0.tcl**, y se presiona **Enter**, luego de ello en la pantalla de OpenSees aparece:

El periodo fundamental de vibración es: 0.2908556321568438 segundos

En los ejemplos que se presentarán posteriormente se incluirá un cuarto paso “**Análisis de resultados**”. Este paso no se incluye en este ejemplo por cuanto no se genera ningún archivo que contenga respuesta al análisis debido a que el valor de respuesta es uno solo y no hay necesidad de grabarlo en un documento de texto. Para ejemplos en que los valores de respuesta obtenidos del análisis son mayores, lo que se hace es colocar el comando “**recorder**”. En los ejemplos que se muestran más adelante se podrá notar el uso de este comando.

Los cuatro pasos mencionados anteriormente son un poco generales, ya que el proceso en sí, siendo un poco más detallados, consta de:

- **Construcción del modelo.**- en esta etapa se define:
 - La dimensión del modelo (número de grados de libertad).
 - Coordenadas de los nudos.
 - Restricciones de los nudos.
 - Masas nodales.
 - Elementos y su conectividad.
 - Archivos de salida.

- **Definición y aplicación de cargas.**- se define:
 - Las cargas en los nudos o elementos.
 - Parámetros del análisis estático (tolerancias e incrementos de carga).
 - Análisis.
 - Congelamiento del análisis y reinicio del tiempo a cero (cuando se va ejecutar otro análisis).

- **Definición y aplicación de las cargas laterales.**- se define:
 - Los patrones de carga (las cargas en los nudos para el análisis estático, debido a movimiento sísmico).

- Parámetros del análisis lateral (tolerancias y desplazamientos en el incremento del tiempo).
 - Análisis lateral estático.- se define:
 - Los incrementos de desplazamiento y el desplazamiento esperado.
 - Análisis lateral dinámico.- se define:
 - El registro de aceleración del sismo y todos los parámetros asociados a él.
 - La duración del análisis y el incremento del tiempo.
 - Se define el amortiguamiento.
- Análisis

Para el modelamiento de estructuras en 3D a más de los puntos mencionados, se deben tomar en cuenta los siguientes:

- Se necesita considerar coordenadas adicionales para la definición de los nudos (x, y, z).
- Se debe considerar grados de libertad adicionales (6 grados de libertad) para definir:
 - Restricciones en los nudos.
 - Masas nodales.
 - Cargas nodales.
- La transformación de coordenadas de los ejes locales de los elementos a los ejes globales del sistema.
- Cargas en los elementos en la dirección de los ejes locales.
- Argumentos adicionales que son requeridos por los elementos.
- Consideración de diafragmas de piso rígido para la construcción del modelo

CAPÍTULO III

ANÁLISIS ELÁSTICO DE ESTRUCTURAS

3. ANÁLISIS ELÁSTICO DE ESTRUCTURAS EN 3D

A continuación se presenta un ejemplo sobre análisis elástico de estructuras, en el que a su vez se indica un poco más a fondo lo relacionado a la preparación de los archivos de entrada en Tcl para OpenSees. En este ejemplo se explica:

- La estructura general de los archivos de entrada.
- La sintaxis de varios comandos necesarios para la generación de modelos, ejecución del análisis y generación de archivos con resultados
- La estructura de los archivos de resultados y su procesamiento.

El ejemplo que se presenta sirve de base para la presentación de ejemplos más complejos en los siguientes apartados, puesto que los pasos a seguir son los mismos para cualquier modelo que se plantee.

3.1.EJEMPLO. ANÁLISIS ELÁSTICO 3D DE HISTORIA EN EL TIEMPO.

En este ejemplo se construirá un modelo 3D de la estructura para su análisis bajo la acción de las dos componentes horizontales del terremoto de Northridge. Con este ejemplo se introducen nuevos conceptos como:

- Análisis 3D,
- Pisos rígidos,
- Extremos rígidos en vigas y columnas,

- Elementos elásticos 3D,
- Análisis bajo dos componentes sísmicas.

A continuación se muestra el procedimiento para la generación del código TCL, análisis con OpenSees y post-procesamiento de resultados.

Antes que nada necesitamos obtener los registros sísmicos con los que vamos a trabajar, para ello usamos la base de datos de terremotos que posee PEER <http://peer.berkeley.edu/smcat/>. Los registros sísmicos se presentan en los archivos CNP106.AT2 y CNP196.AT2 disponibles en la página presentada. Estos archivos pueden abrirse con cualquier editor de texto incluso si su extensión no es “.txt”.

Paso 1: Idealización de la estructura

Antes de desarrollar el código es necesario idealizar la estructura y tener una idea clara de cómo vamos a modelarla. En este proceso se define la localización de los nudos, propiedades de materiales y secciones, conectividad de los elementos, tipos de elementos, cargas, opciones de análisis, el tipo de resultados que queremos registrar, etc. La calidad de la simulación depende de las opciones seleccionadas aquí. OpenSees al igual que otros programas de análisis estructural puede resolver un mismo problema de distintas maneras. El criterio del Ingeniero analista y su conocimiento sobre la dinámica de estructuras son esenciales para lograr que el comportamiento del modelo propuesto se asemeje al de la estructura real.

El pórtico 3D que se analiza en este ejemplo tiene una geometría como la mostrada en la figura 2. El pórtico tiene 5 m x 6 m en su base y dos niveles de 3.5 m. El pórtico está hecho con hormigón de resistencia a la compresión $f^c = 28$ MPa. Las columnas son cuadradas de 0.40m de lado. Las vigas tiene 0.25 m de base y 0.40 m de altura. Los entrepisos son losas nervadas de hormigón armado que pesan 6 kN/m².

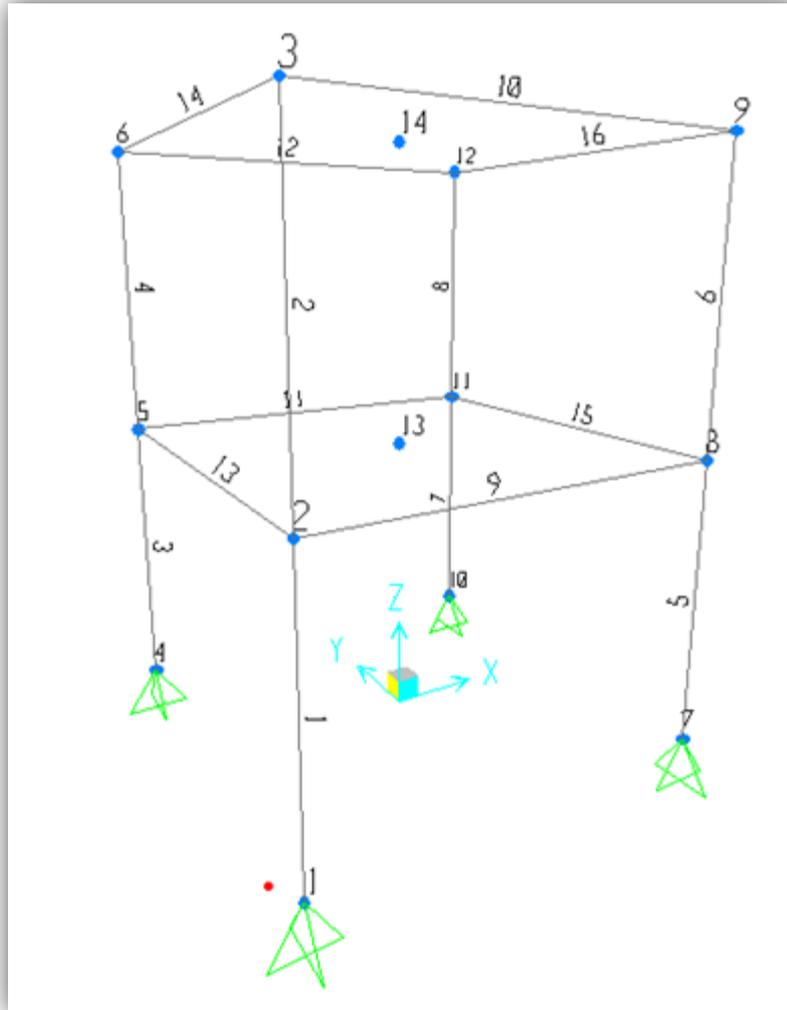


FIGURA 4 – Geometría del pórtico 3D

Para este ejemplo se ha decidido modelar la estructura de la siguiente manera:

- El modelo tiene 14 nudos y 16 barras. Los nudos 13 y 14 están en el centro de masa del piso 1 y 2 respectivamente
- Se modela las vigas y columnas como elementos elásticos. El modulo elástico del hormigón se toma como base para definir las propiedades mecánicas de las secciones. Para considerar el agrietamiento de las vigas y columnas se aplicará un coeficiente de reducción de inercia de 0.5 a las vigas y de 0.7 a las columnas.
- Se modelan los nudos como rígidos. Las fuerzas internas en vigas deberán obtenerse en la cara de las columnas y las fuerzas internas en las columnas deberán obtenerse en la cara de las vigas.
- Se simularan pisos rígidos mediante la definición de diafragmas en cada piso en donde el nudo en el centro de rigidez actúa como uno maestro y los demás nudos en el piso actúan como nudos esclavos.
- Se aplicaran sobre las vigas y columnas cargas uniformemente distribuidas para representar adecuadamente el peso propio de la estructura.

- No se considerara el aporte los entrepisos a la rigidez lateral de la estructura.
- La masa de la estructura se distribuirá apropiadamente en sus nudos
- Se considerara efectos P-Delta en el análisis de las columnas.

Paso 2: Generación del modelo 3D

A continuación se presenta el código TCL generado para este ejemplo con comentarios que explican paso a paso cada una de las partes del mismo.

```
#Ejemplo3.tcl #
#Análisis 3D Elástico de Historia en el tiempo de un pórtico de dos pisos
#####
#####
#Unidades: kN m s #
#####
#####
set pi [expr acos(-1.0)]
wipe ;# Este comando borra todos los objetos existentes
```

Se crea un modelo de 3 dimensiones con 6 grados de libertad por nudo

```
model basic -ndm 3 -ndf 6 ;# 3 dimensiones; 6 GDL por nudo
# Se define la geometría -----
# Coordenadas de los nudos
node 1 -2.5 -3 0
node 2 -2.5 -3 3.5
node 3 -2.5 -3 7
node 4 -2.5 3 0
node 5 -2.5 3 3.5
node 6 -2.5 3 7
node 7 2.5 -3 0
node 8 2.5 -3 3.5
node 9 2.5 -3 7
node 10 2.5 3 0
node 11 2.5 3 3.5
node 12 2.5 3 7
# Nudos maestros
node 13 2.5 2.5 3.5
node 14 2.5 2.5 7
```

El comando rigidDiaphragm condiciona el desplazamientos en el plano (grados de libertad 1 y 2 en este ejemplo) y la rotación perpendicular al plano de nudos esclavos (gdl 6) al desplazamiento en el plano y la rotación perpendicular al plano de un nudo

maestro de manera que todos los nudos se desplacen como si fueran parte de diafragma rígido. Los otros grados de libertad (3 4 5) no se acoplan.

```
# Creo los pisos rígidos con el comando rigidDiaphragm
# En el primer piso, el nudo maestro es el nudo 13 y los nudos esclavos son 2 5 8 11
rigidDiaphragm 3 13 2 5 8 11
# En el segundo piso, el nudo maestro es el nudo 14 y los nudos esclavos son 3 6 9 12
rigidDiaphragm 3 14 3 6 9 12
# Condiciones de borde - Las columnas están empotradas en su base
fix 1 1 1 1 1 1 1 ;# Se bloquean los 6 grados de libertad de cada nudo
fix 4 1 1 1 1 1 1
fix 7 1 1 1 1 1 1
fix 10 1 1 1 1 1 1
```

Es necesario bloquear los grados de libertad 3 4 5 de los nudos maestros ya que no existe rigidez asociada a estos grados de libertad en los diafragmas.

```
fix 13 0 0 1 1 1 0
fix 14 0 0 1 1 1 0
```

Considerando el peso propio de los elementos y el peso del entrepiso se estima una carga uniforme sobre las vigas de 10.58 kN/m y una carga uniforme sobre las columnas de 3.84 kN/m. Estas cargas actúan en la dirección $-Z$ para representar los efectos gravitacionales. A partir de estas cargas y de las longitudes de vigas y columnas se estima una masa concentrada en los nudos de 6.87 t.

```
# Se Define masas a partir de peso propio de la estructura
mass 2 6.87 6.87 6.87 1e-6 1e-6 1e-6
mass 3 6.87 6.87 6.87 1e-6 1e-6 1e-6
mass 5 6.87 6.87 6.87 1e-6 1e-6 1e-6
mass 6 6.87 6.87 6.87 1e-6 1e-6 1e-6
mass 8 6.87 6.87 6.87 1e-6 1e-6 1e-6
mass 9 6.87 6.87 6.87 1e-6 1e-6 1e-6
mass 11 6.87 6.87 6.87 1e-6 1e-6 1e-6
mass 12 6.87 6.87 6.87 1e-6 1e-6 1e-6
```

A partir del valor de $f'c$ y del valor de la relación de poisson para el hormigón se calculan los valores del modulo elástico E y modulo de corte G. También se calculan las propiedades mecánicas: Inercia en el eje local z , Iz, inercia en el eje local y, Iy, área, A y constante de torsión, J, para las secciones de viga y columna.

```
# Definición de Elementos -----
# Propiedades elásticas del hormigón
```

```

set fc 28 ;# f'c MPa
set poisson 0.25 ;# Relación de poisson para el concreto
set E [expr 4700*sqrt($fc)*1000]
set G [expr $E/(2*(1+$poisson))]
# Propiedades mecánicas de las secciones
set bviga 0.25 ;# base
set hviga 0.40 ;# altura
set RIviga 0.5 ;# reducción de inercia
set bcol 0.40 ;# base
set RIcol 0.7 ;# reducción de inercia
set Aviga [expr $bviga*$hviga]
set Iyviga [expr $RIviga*$bviga*$hviga**3/12]
set Izviga [expr $RIviga*$hviga*$bviga**3/12]
set bj 0.20 ;# Relación entre lado largo y lado corto para calculo de J
set Jviga [expr $bj*$hviga**$bviga**3]
set bcol 0.40 ;# las columnas son cuadradas
set Acol [expr $bcol**2]
set Iycol [expr $RIcol*$bcol**4/12]
set Izcol $Iycol
set bj 0.14 ;# Relación entre lado largo y lado corto para calculo de J
set Jcol [expr $bj*$bcol**4]

```

Para poder orientar correctamente los ejes locales de las vigas y columnas definiendo extremos rígidos se crean 4 transformaciones de coordenadas. La primera es para las vigas en el plano XZ, la segunda para las vigas en el plano YZ, la tercera para las columnas del primer piso y la cuarta para las columnas del 2 piso.

Para crear extremos rígidos se necesita calcular dX_i , dY_i y dZ_i en el extremo inicial del elemento y dX_j , dY_j y dZ_j en el extremo final. Estos parámetros son distancias medidas en el sistema global de coordenadas, ver figura 3. En el caso de las vigas que están en el plano XZ, $dX_i = 0.2$ m (1/2 del lado de la columna) y $dX_j = -0.2$ m. Los valores de dY_i , dZ_i , dY_j , dZ_j , son 0 porque no existe corrimiento del eje del elemento en esas direcciones.

En el caso de las vigas que están sobre el plano YZ el corrimiento del eje se da solo en la dirección Y y no en la dirección X como fuera el caso para las vigas sobre el plano XZ. Por esta razón es necesario definir una transformación de coordenadas distinta para las vigas en el plano YZ.

Las columnas también necesitan dos transformaciones de coordenadas. En las columnas del primer piso solo se crea un extremo rígido en el extremo j, mientras que en las columnas del segundo piso se crean entremos rígidos en los dos extremos. Además se asigna una transformación P-delta para las columnas.

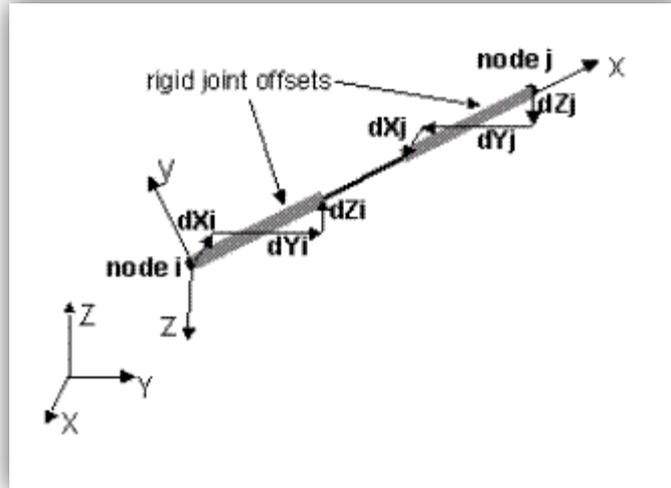


FIGURA 5 – Definición de los extremos rígidos

```

# Se define una transformación de coordenadas lineal para las vigas
# Se usara el vector (0 0 1) para orientar las vigas, usando este vector el eje local Z de
las vigas se orienta hacia arriba (Z)
# se crean nudos rígidos de 0.20 m en los extremos de las vigas
# para vigas en el plano X Z
geomTransf Linear 1 0 0 1 -jntOffset [expr $bcol/2] 0 0 [expr -$bcol/2] 0 0
# para vigas en el plano Y Z
geomTransf Linear 2 0 0 1 -jntOffset 0 [expr $bcol/2] 0 0 [expr -$bcol/2] 0
# Se define una transformación P-Delta para las columnas
# el vector (1 0 0) orienta el eje local Z de las columnas en la dirección global X
# Se crea un extremo rígido de 0.20 m en la cabeza de las columnas del primer piso
geomTransf Linear 3 1 0 0 -jntOffset 0 0 0 0 0 [expr $hвига/2]
# Se crea un extremo rígido de 0.20 m en los dos extremos de las columnas del segundo
piso
geomTransf Linear 4 1 0 0 -jntOffset 0 0 0.2 0 0 [expr -$hвига/2]
# sintaxis del elemento elástico viga columna
#element elasticBeamColumn $eleTag $iNode $jNode $A $E $G $Jcol $Iycol $Izcol $transfTag
# Columnas
element elasticBeamColumn 1 1 2 $Acol $E $G $Jcol $Iycol $Izcol 3
element elasticBeamColumn 2 2 3 $Acol $E $G $Jcol $Iycol $Izcol 4
element elasticBeamColumn 3 4 5 $Acol $E $G $Jcol $Iycol $Izcol 3
element elasticBeamColumn 4 5 6 $Acol $E $G $Jcol $Iycol $Izcol 4
element elasticBeamColumn 5 7 8 $Acol $E $G $Jcol $Iycol $Izcol 3
element elasticBeamColumn 6 8 9 $Acol $E $G $Jcol $Iycol $Izcol 4
element elasticBeamColumn 7 10 11 $Acol $E $G $Jcol $Iycol $Izcol 3
element elasticBeamColumn 8 11 12 $Acol $E $G $Jcol $Iycol $Izcol 4
# Vigas
element elasticBeamColumn 9 2 8 $Aviga $E $G $Jviga $Iyviga $Izviga 1
element elasticBeamColumn 10 3 9 $Aviga $E $G $Jviga $Iyviga $Izviga 1
element elasticBeamColumn 11 5 11 $Aviga $E $G $Jviga $Iyviga $Izviga 1
element elasticBeamColumn 12 6 12 $Aviga $E $G $Jviga $Iyviga $Izviga 1

```

```
element elasticBeamColumn 13 2 5 $Aviga $E $G $Jviga $Iyviga $Izviga 2
element elasticBeamColumn 14 3 6 $Aviga $E $G $Jviga $Iyviga $Izviga 2
element elasticBeamColumn 15 8 11 $Aviga $E $G $Jviga $Iyviga $Izviga 2
element elasticBeamColumn 16 9 12 $Aviga $E $G $Jviga $Iyviga $Izviga 2
```

Se utilizan recorders para registrar desplazamientos y derivas en los nudos maestros. También se registran reacciones en la base.

```
# Registro de resultados -----
# Estos recorders registran la historia en el tiempo de desplazamientos derivas y
reacciones
recorder Node -file Desplazamiento.out -time -node 13 14 -dof 1 2 disp
recorder Drift -file DerivaX.out -time -iNode 1 13 -jNode 13 14 -dof 1 -perpDim 3
recorder Drift -file DerivaY.out -time -iNode 1 13 -jNode 13 14 -dof 2 -perpDim 3
recorder Node -file Reacciones.out -time -node 1 4 7 10 -dof 1 2 3 4 5 6 reaction
# Estos recorders registran la envolvente de desplazamientos y derivas
recorder EnvelopeNode -file DesplazamientoE.out -time -node 13 14 -dof 1 2 disp
recorder EnvelopeDrift -file DerivaXE.out -time -iNode 1 13 -jNode 13 14 -dof 1 -
perpDim 3
recorder EnvelopeDrift -file DerivaYE.out -time -iNode 1 13 -jNode 13 14 -dof 2 -
perpDim 3
```

Se podrían crear otros recorders para registrar fuerzas y otros valores de respuesta en la estructura, sin embargo, en este ejemplo solamente usaremos los presentados en el código.

Se aplica la carga distribuida a las vigas y columnas, para ello se utiliza el comando “eleLoad” para cargar los elementos. Como se ve abajo, la opción “-ele” permite aplicar una misma carga a varios elementos. El tipo de carga “-beamUniform” requiere la definición de la carga distribuida que actúa en la dirección del eje “y” del elemento, luego en el eje “z” del elemento y finalmente en el eje “x” del elemento. Vea la figura 4. En el caso de las vigas, la carga se aplica en el eje “z”, en forma consistente con la orientación de ejes locales dada con el comando “geomTransf” y con signo negativo para que la carga actúe como carga gravitacional. En el caso de las columnas la carga se aplica en la dirección del eje axial “x” del elemento y también con signo negativo (recordemos que las columnas se crearon con el nudo inicial abajo y el final arriba) para logra el efecto de un carga gravitacional.

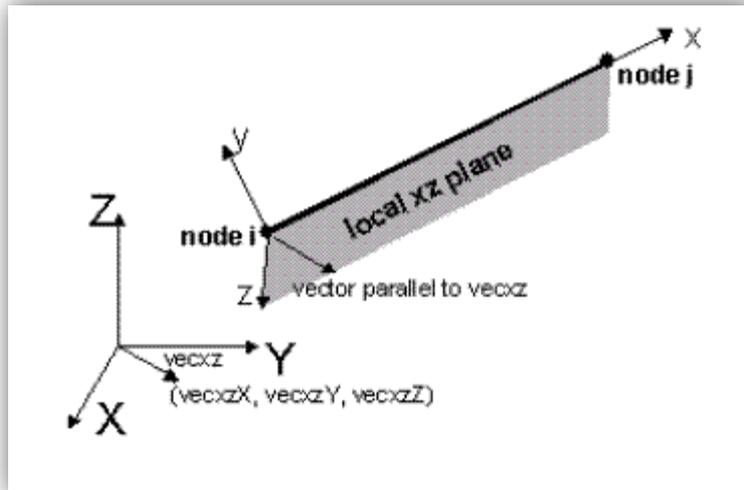


FIGURA 6 – Definición del sistema local de coordenadas

```

# Aplicacion de cargas -----
pattern Plain 1 Linear {
eleLoad -ele 9 10 11 12 13 14 15 16 -type -beamUniform 0 -10.58 0
eleLoad -ele 1 2 3 4 5 6 7 8 -type -beamUniform 0 0 -3.84
}
# Se ejecuta analisis estatico
constraints Transformation
numberer Plain
system BandGeneral
test NormDispIncr 1.0e-8 6
algorithm Linear
integrator LoadControl 1
analysis Static
analyze 1
# Se congela la aplicacion de la carga en todos los patrones definidos hasta ahora
loadConst -time 0.0

```

La duración del análisis y el paso del análisis se obtienen de la información sobre el terremoto que se va a simular. En este ejemplo se van a aplicar las dos componentes horizontales del sismo de Northridge. Los registros de aceleración se presentan los archivos CNP106.AT2 y CNP196.AT2. Estos archivos pueden abrirse con cualquier editor de texto.

```

# Define el patrón con el que se aplica el registro de aceleraciones
# Duración del análisis
set duration 25
set stepEq 0.01 ;# paso del sismo
set stepA [expr $stepEq/2] ;# paso del análisis

```

Se cargan las dos componentes sísmicas del sismo de Northridge. Para esto editamos los archivos originales que se descargaron de la base de datos de sismos con la finalidad de eliminar las primeras 4 líneas que contienen información sobre el sismo. El factor 9.81 se aplica para convertir los valores de aceleración en los registros sísmicos a m/s², ya que los valores que contienen estos archivos están en unidades de “g” (gravedad). Una vez modificados los archivos que contienen los valores de aceleración los guardamos con los siguientes nombres: accX.txt y accY.txt

```
set accX "Series -dt $stepEq -filePath accX.txt -factor 9.81"
set accY "Series -dt $stepEq -filePath accY.txt -factor 9.81"
```

Se definen dos patrones de carga para aplicar una componente en la dirección global X y otra en la dirección global Y

```
pattern UniformExcitation 2 1 -accel $accX
pattern UniformExcitation 3 2 -accel $accY
# Para asignar amortiguamiento Raleigh, tenemos que obtener los periodos de los dos
modos de vibración
set lambda1 0.05 ;# amortiguamiento asignado al primer modo de vibración
set lambda2 0.05 ;# amortiguamiento asignado al segundo modo de vibración
set eigenvalues [eigen 2] ;# Obtenemos los valores propios de los primeros 2 modos de
vibración
set omega1 [expr sqrt([lindex $eigenvalues 0])] ;# Las frecuencias de vibración son la
raíz cuadrada de los valores propios
set omega2 [expr sqrt([lindex $eigenvalues 1])] ;# Los valores propios están en una
lista, el comando lindex extrae elementos de la lista
set T1 [expr 2.0*$pi/$omega1] ;# Periodo del primer modo de vibración
puts "T1: $T1 segundos"
set T2 [expr 2.0*$pi/$omega2] ;# Periodo del segundo modo de vibración
puts "T2: $T2 segundos"
set alpha [expr (2*$omega1*$omega2*($omega1*$lambda2-
$omega2*$lambda1))/($omega1**2-$omega2**2)]
set beta [expr 2*($omega1*$lambda1-$omega2*$lambda2)/($omega1**2-
$omega2**2)]
```

A continuación se define el integrador y el tipo de análisis para correr los sismos.

```
# Se ejecuta análisis dinámico paso a paso en el tiempo
integrator Newmark 0.5 0.25 $alpha $beta 0 0
analysis Transient
analyze [format "%.0f" [expr $duration/$stepA]] $stepA
#####
#FIN
#####
```

Paso 3: Análisis

Para ejecutar la simulación, se inicia `opensees.exe` y se llama al código TCL con “`source ejemplo3.tcl`”. Los archivos `accX.txt` y `accY.txt` con los registros sísmicos deberán estar en la misma carpeta que `opensees.exe` y `ejemplo3.tcl`.

Como resultado del análisis se crearan los archivos generados por los recorders, los cuales se guardan en el directorio desde el cual se esta trabajando.

Paso 4: Análisis de resultados

El archivo `Desplazamiento.out` puede procesarse con Excel. De izquierda a derecha las columnas en el archivo contienen: tiempo, Desp. X nudo 13, Desp. Y nudo 13, Desp. X nudo 14, Desp. Y nudo 14. Este orden viene dado por la forma en la que se creó el recorder. Si se grafican los desplazamientos en función del tiempo se obtiene:

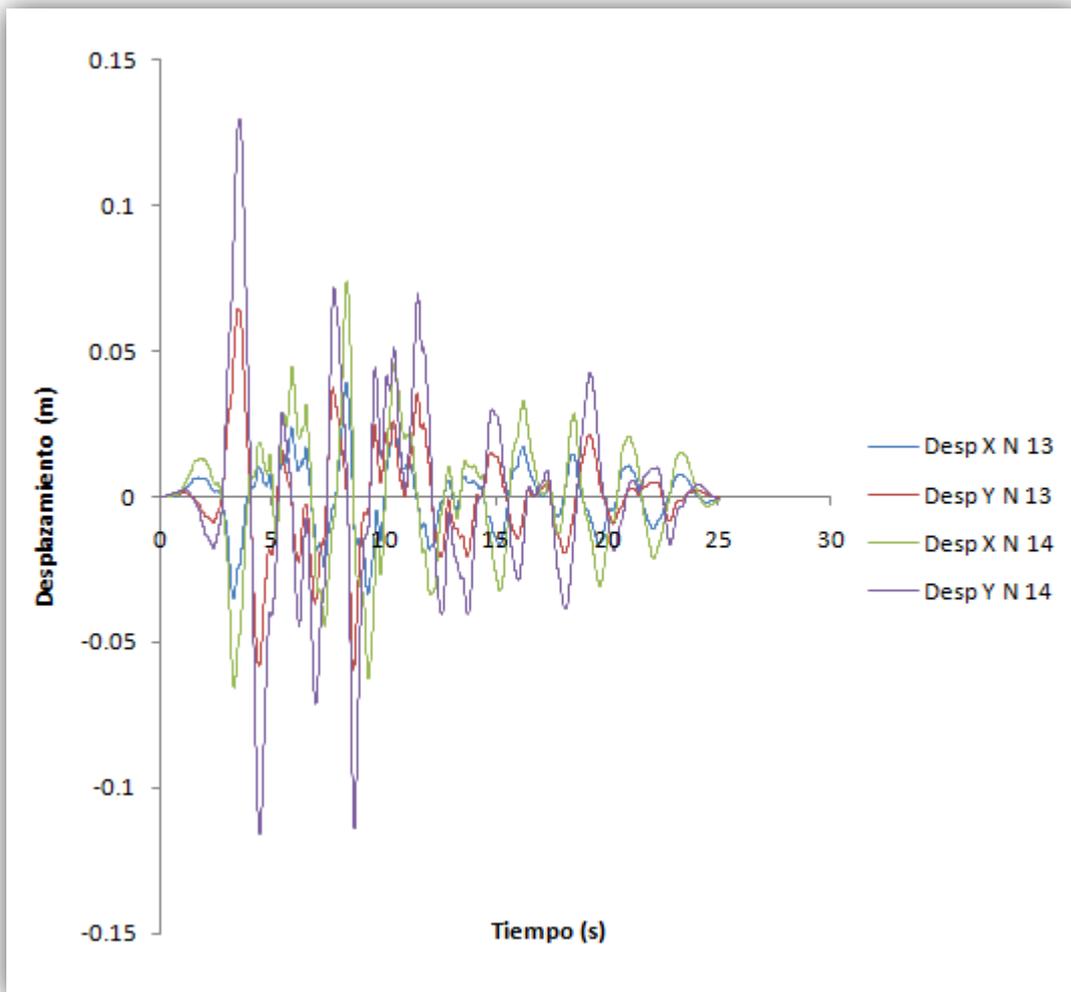


FIGURA 7 – Análisis de resultados (Desplazamiento vs Tiempo)

De esta gráfica o del archivo DesplazamientoE.out se puede extraer los valores máximos de respuesta. Los desplazamientos máximos ocurren en el instante $t= 3.56s$, en el segundo piso (nudo 14), en la dirección Y, con magnitud 0.129494 m.

Los archivos DerivaX.out y DerivaY.out también pueden ser procesados con Excel. De izquierda a derecha las columnas en el archivo contienen: tiempo, Deriva piso1, Deriva piso 2. Este orden viene dado por la forma en la que se creó el recorder. Como ejemplo, si se grafican las derivas en la dirección Y para el segundo piso se obtiene la siguiente gráfica. A partir de esta grafica o desde el archivo DerivaYE.out se puede determinar el valor de la deriva máxima. Es importante notar que los valores de deriva también pueden calcularse a partir de los valores de desplazamiento.

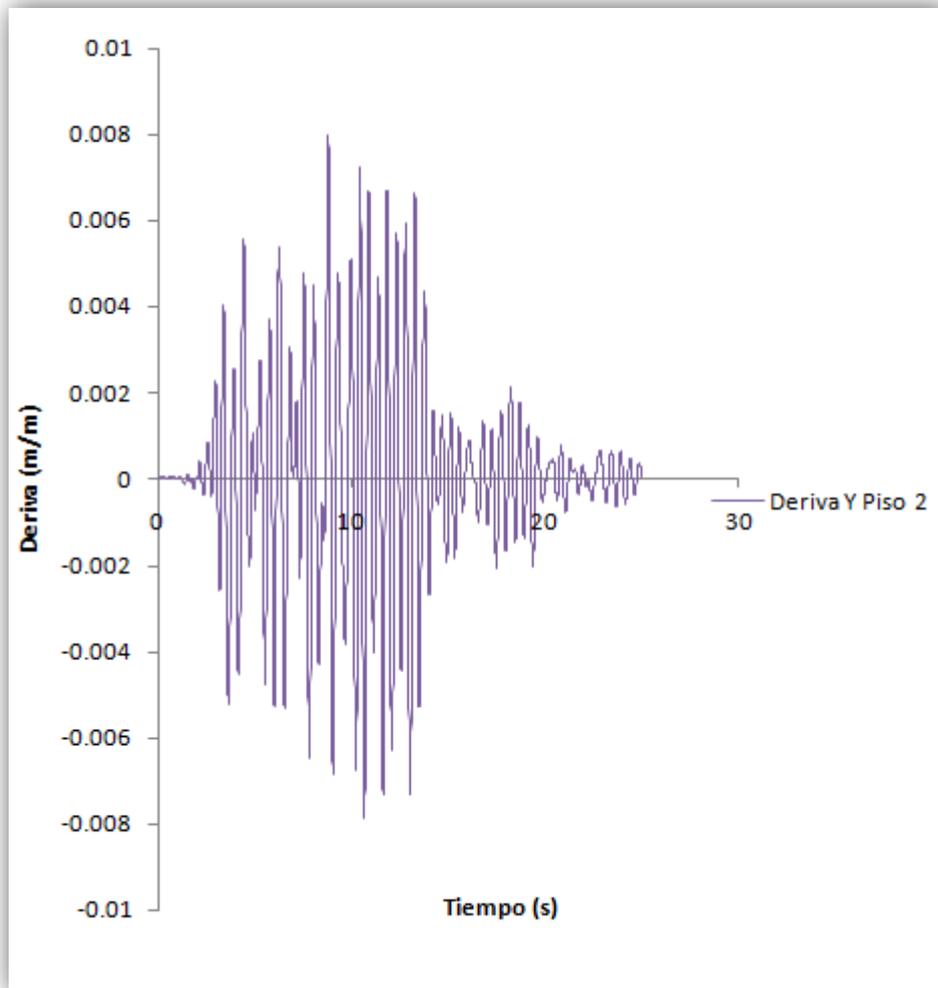


FIGURA 8 – Análisis de resultados (Deriva vs Tiempo)

CAPÍTULO IV

SECCIONES EN OPENSEES

4. SECCIONES

4.1. SECCIONES CON FIBRAS

La generación de secciones es el paso previo a la generación de modelos de elementos vigas o columnas para el análisis no-lineal de una estructura. Para la creación de estas secciones de hormigón armado se hará uso de los materiales previamente definidos en OpenSees, como son:

- Elastic
- Elastic-Perfectly Plastic Gap Material
- Steel01
- Steel02
- Concrete01
- Concrete02
- Viscous, entre otros.

El comportamiento no-lineal de secciones de hormigón armado puede ser investigado con un tipo especial de análisis llamado Momento-Curvatura. El resultado de este análisis es la relación entre el momento aplicado y la curvatura generada en la sección.

La teoría de flexión en vigas establece que la curvatura es la segunda derivada del desplazamiento. De ahí que la relación momento-curvatura es muy importante ya que permite el cálculo de desplazamientos durante el análisis no-lineal de estructuras. Otra información muy importante producida por este tipo de análisis es la relación entre

curvatura y la deformación unitaria en las fibras de hormigón y acero en la sección, así como la distancia al eje neutro. Esta información sirve para definir estados límites de daño tales como: agrietamiento del concreto, fluencia del acero, trituración del concreto, ruptura del acero y otros.

Para simular correctamente la relación momento-curvatura de una sección de hormigón armado se debe definir correctamente la geometría de la sección y la relación esfuerzo-deformación de los materiales constituyentes. Cuando la sección contenga una cantidad considerable de refuerzo transversal, la relación esfuerzo-deformación del hormigón confinado es diferente a la del hormigón de recubrimiento (no-confinado) tal como se muestra en la siguiente figura.

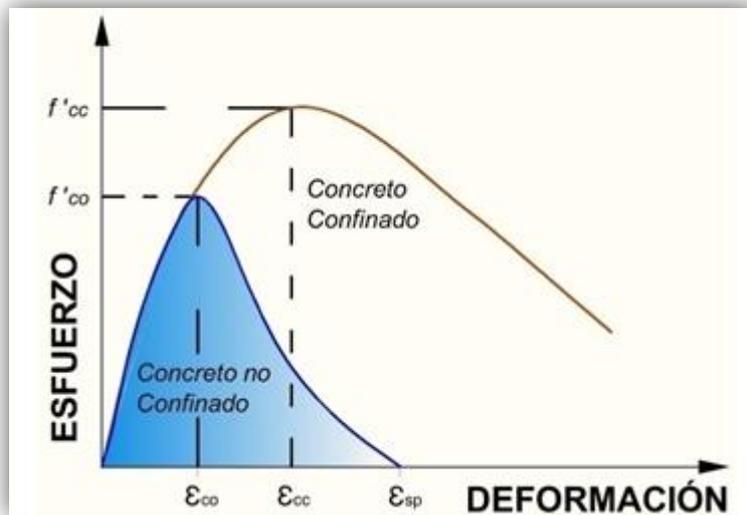


FIGURA 9 – Esfuerzo Deformación del concreto

Existen varios programas de computación especialmente diseñados para la obtención del diagrama momento – curvatura. El programa RC-Analysis genera una curva Momento-Curvatura junto a una curva de resistencia al corte para los diferentes tipos de secciones de concreto. Otro programa es Response 2000 que sirve básicamente para la obtención de diagramas momento – curvatura para secciones de hormigón armado y hormigón pre-esforzado.

4.2.GENERACIÓN DE SECCIONES EN OPENSEES

En OpenSees las secciones de vigas o columnas pueden ser de tres tipos: Elástica, Uniaxial y de Fibras.

4.2.1. Sección Elástica

Se crean con el comando “section Elastic” con base en las propiedades elásticas de la sección. La sintaxis es la siguiente:

section Elastic \$secTag \$E \$A \$Iz <\$Iy \$G \$J>

\$secTag	Etiqueta única para la sección
\$E	Módulo Elástico
\$A	Área de la sección
\$Iz	Inercia con respecto al eje local z
\$Iy	Inercia con respecto al eje local y (opcional, usado en análisis 3D)
\$G	Módulo de corte (opcional, usado en análisis 3D)
\$J	Inercia torsional de la sección (opcional, usado en análisis 3D)

Esta sección se utiliza para dar comportamiento elástico a elementos no-lineales viga columna. Su uso es limitado ya que se pueden crear elementos elásticos de manera más sencilla y directa usando el comando “Elastic Beam Column Element”.

4.2.2. Sección Uniaxial

Se crea con el comando “section Uniaxial” usando la respuesta fuerza-desplazamiento de materiales uniaxiales previamente definidos. La sintaxis es la siguiente:

section Uniaxial \$secTag \$matTag \$string

\$secTag	Etiqueta única para la sección
\$matTag	Etiqueta de un material uniaxial previamente definido
\$string	Identificador del tipo de respuesta. Aplican las siguientes:
P	Fuerza Axial – deformación.
Mz	Momento – curvatura respecto a eje local z.
Vy	Fuerza cortante – deformación respecto a eje local y.
My	Momento – curvatura respecto a eje local y.
Vz	Fuerza cortante – deformación respecto a eje local z.
T	Torque – deformación.

Por ejemplo, el comando:

section Uniaxial 1 10 Mz

Crea la sección uniaxial 1 usando el material 10 para definir la respuesta momento – curvatura en el eje local z. El material 10 podría ser un material cualquier tipo de material disponible en OpenSees, incluso podría ser uno creado en base a los resultados de un análisis momento curvatura de la sección real realizada previamente.

4.2.3. Sección de Fibras

Se crea con el comando “section Fiber”. Este tipo de sección se forma con fibras de distintos materiales uniaxiales previamente definidos. La sintaxis es la siguiente:

```
section Fiber $secTag {  
    fiber <fiber arguments>  
    patch <patch arguments>  
    layer <layer arguments>  
}
```

Una sección de fibras puede tener cualquier forma. Los parámetros geométricos se definen con respecto a los ejes locales y, z. Un ejemplo de sección de fibras se muestra en la siguiente figura.

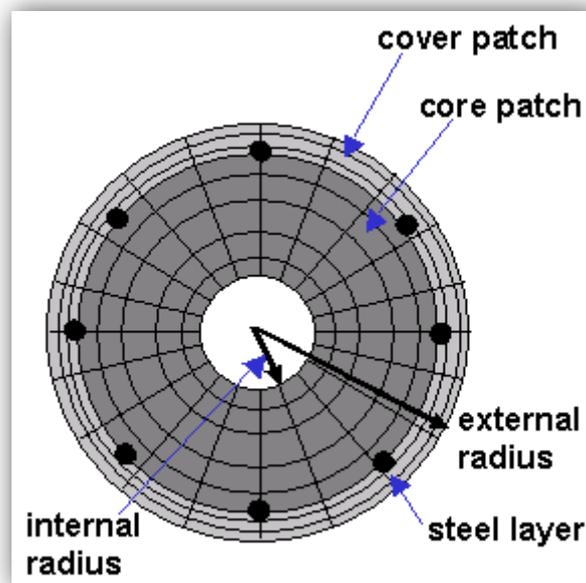


FIGURA 10 – Sección circular con fibras

El permitir la creación de secciones con fibras es sin duda una de las mejores características de OpenSees. Algunas de las ventajas de utilizar este tipo de secciones son:

- La incorporación automática la interacción entre carga axial y momento. Es decir la sección cambia instantáneamente su resistencia a flexión de acuerdo a la carga axial que se le aplica.
- El modelo captura de forma explícita los efectos de flexión biaxial.
- Puede tener cualquier forma y estar formada por fibras de varios materiales
- Es posible monitorear la respuesta esfuerzo-deformación de cada fibra en la sección.

También tiene algunas limitaciones:

- Considera que las secciones planas permanecen planas
- La magnitud de las deformaciones unitarias varia con el tipo de elemento y número de puntos de integración por lo que se debe usar con precaución
- Su uso es computacionalmente costoso

Para discretizar una sección en un conjunto de fibras se usa varios comandos. Subregiones de fibras con formas simples (cuadriláteros o círculos) se generan con el comando “patch”. Esto permite, por ejemplo, la creación de las fibras de hormigón confinado y de concreto no confinado en una sección. El comando “layer” permite crear capas de fibras con el acero de refuerzo. Adicionalmente, las fibras pueden ser agregadas una a una con el comando “fiber”. La sintaxis de “fiber” es:

fiber \$yLoc \$zLoc \$A \$matTag

\$yLoc	Coordenada local y de la fibra en la sección
\$zLoc	Coordenada local z de la fibra en la sección
\$A	Área de la fibra
\$matTag	Etiqueta del material uniaxial con el modelo esfuerzo deformación para la fibra.

NOTA: La flexión en 2D es en el eje local z

Como ejemplo:

fiber 0 0 1 1

Crea una fibra de area 1 en las coordenadas 0,0 de la sección usando el material 1

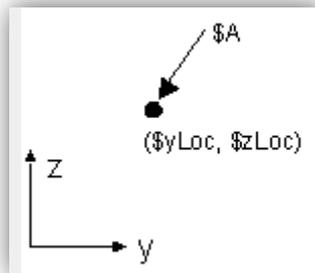


FIGURA 11 – Comando fibra

Para crear parches con forma de cuadrilátero, el comando “patch quad” tiene la siguiente sintaxis:

**patch quad \$matTag \$numSubdivIJ \$numSubdivJK \$yI \$zI \$yJ \$zJ
\$yK \$zK \$yL \$zL**

\$matTag	Es la etiqueta del material que forma las fibras
\$numSubdivIJ	Numero de fibras en las dirección IJ

\$numSubdivJK	Numero de fibras en la dirección JK
\$yI \$zI	Coordenadas y & z del vértice I (coordenadas locales)
\$yJ \$zJ	Coordenadas y & z del vértice J (coordenadas locales)
\$yK \$zK	Coordenadas y & z del vértice K (coordenadas locales)
\$yL \$zL	Coordenadas y & z del vértice L (coordenadas locales)

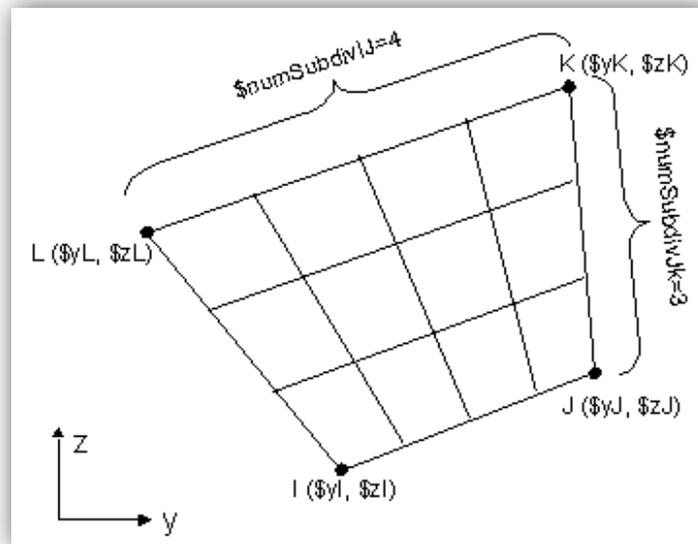


FIGURA 12 – Características del comando “patch quad”

Para crear parches con forma circular, el comando “patch circ” tiene la sintaxis:

patch circ \$matTag \$numSubdivCirc \$numSubdivRad \$yCenter \$zCenter \$intRad \$extRad <\$startAng \$endAng>

\$matTag	Etiqueta del material de las fibras
\$numSubdivCirc	Numero de fibras en la dirección circunferencial
\$numSubdivRad	Numero de fibras en la dirección radial
\$yCenter	Coordenada y del centro del parche
\$zCenter	Coordenada z del centro del parche
\$intRad	Radio interior
\$extRad	Radio exterior
\$startAng	Angulo inicial
\$endAng	Angulo final

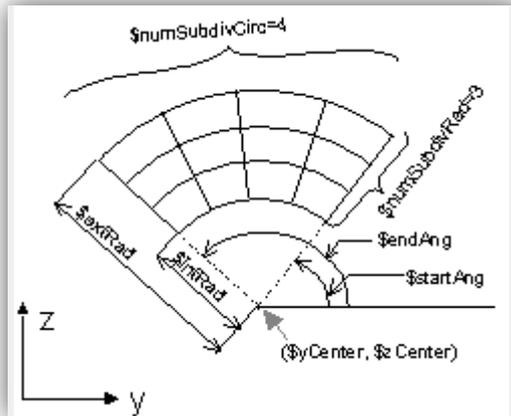


FIGURA 13 – Características del comando “patch circ”

Para crear capas lineales de refuerzo se utiliza:

layer straight \$matTag \$numBars \$areaBar \$yStart \$zStart \$yEnd \$zEnd

- \$matTag** Es la etiqueta del material en las fibras
- \$numBars** Número de barras (fibras) en la capa
- \$areaBar** Área de cada barra
- \$yStart** Coordenada y al inicio de la capa
- \$zStart** Coordenada z al inicio de la capa
- \$yEnd** Coordenada y al final de la capa
- \$zEnd** Coordenada z al final de la capa

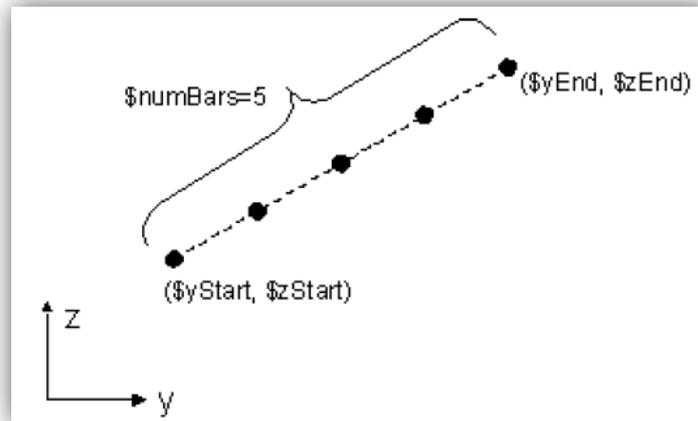


FIGURA 14 – Características del comando “layer straight”

Para crear capas circulares de refuerzo se utiliza:

**layer circ \$matTag \$numBar \$areaBar \$yCenter \$zCenter \$radius
<\$startAng \$endAng>**

\$matTag	Etiqueta del material de las fibras
\$numBar	numero de barras (fibras) en la capa
\$areaBar	Área de cada barra
\$yCenter	Coordenada y del centro de la capa
\$zCenter	Coordenada z del centro de la capa
\$radius	Radio
\$startAng	Angulo inicial
\$endAng	Angulo final

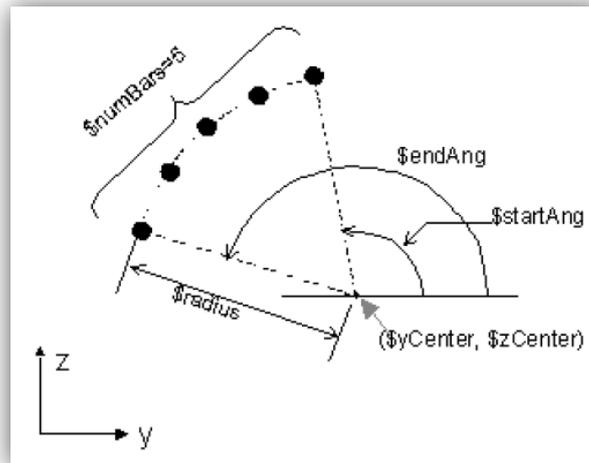


FIGURA 15 – Características del comando “layer circ”

NOTA: Se debe tomar en cuenta que el comando crea una fibra al comienzo y al final de la capa. Por lo tanto, si el ángulo inicial es 0 y el final 360 se duplicara la fibra del inicio.

4.3.AGREGADOR DE SECCIONES

Todo los elementos viga o columna de una estructura tienen que soportar simultáneamente flexión, corte y posiblemente fuerzas axiales y de torsión. Las secciones de fibras resisten flexo-compresión solamente, por lo que la respuesta a corte y torsión debe asignarse por separado, creando para ello una sección compuesta. Lamentablemente hasta el momento la respuesta a cortante esta desacoplada de la respuesta a flexo-compresión.

El comando “**section Aggregator**” permite agregar a una sección previamente definida nuevos componentes de respuesta. Por ejemplo, a una sección uniaxial a la que se le ha asignado respuesta momento-curvatura, se le puede añadir la respuesta a cortante, como se muestra a continuación:

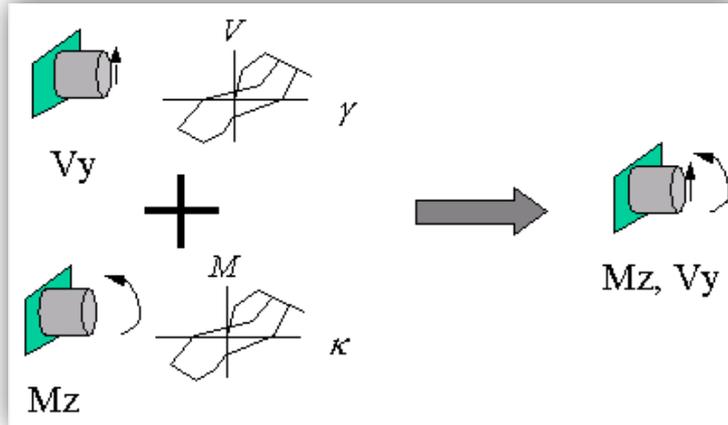


FIGURA 16 – Agregador de secciones 1

También es posible agregar a la respuesta flexo-compresión de una sección de fibras, el comportamiento a cortante o a torsión contenido en otros materiales uniaxiales.

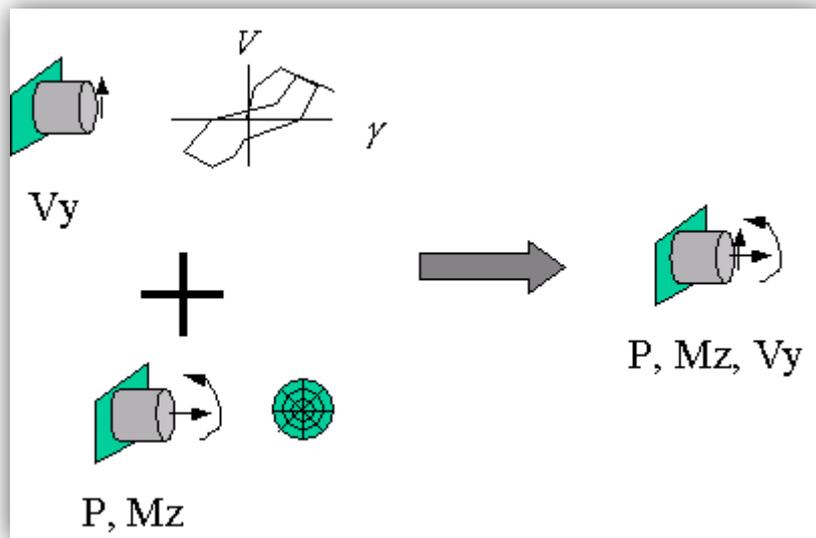


FIGURA 17 – Agregador de secciones 2

La sintaxis es la siguiente:

```
section Aggregator $secTag $matTag1 $string1 $matTag2 $string2
..... <-section $sectionTag>
```

\$secTag	Etiqueta única de la sección
\$matTag1, \$matTag2 ...	Materiales uniaxiales previamente definidos
\$string1, \$string2	Identificador de la respuesta asignada a cada material. Son validos los siguientes:
P	Fuerza Axial-Deformación
Mz	Momento-Curvatura respecto a eje local z

Vy	Fuerza cortante-deformación respecto a eje local y
My	Momento-Curvatura respecto a eje local y
Vz	Fuerza cortante-deformación respecto a eje local z
T	Torque – deformación
<-section \$sectionTag>	Sección previamente definida a la que se le añade nuevos componentes de respuesta

A continuación se presenta un ejemplo donde se muestra el diagrama momento-curvatura para secciones rectangulares y circulares de hormigón armado, obtenidos mediante OpenSees y mediante el programa RC-Analysis.

4.3.1. Ejemplo. Análisis Momento-Curvatura.

El código en que se presenta en este ejemplo permite realizar análisis momento curvatura en OpenSees de secciones hormigón armado de forma circular y rectangular. Esto se logra mediante la creación de un modelo de 2 dimensiones con 3 GDL por nudo y con las siguientes características:

- Se crean secciones de fibras para las secciones circulares y rectangulares definidas por el usuario.
- Las propiedades del concreto confinado son calculadas en tiempo de ejecución utilizando el modelo de Mander.
- La sección de fibras se asigna a un elemento de longitud 0 que conecta a dos nudos ubicados en las mismas coordenadas.
- Durante la ejecución del análisis primero se analiza el modelo bajo la acción de la carga axial definida por el usuario y luego bajo la acción de un momento que se incrementa controlando el giro del nudo libre.

```
#####
# Código para Analisis Momento Curvatura de una seccion de fibras
# El analisis se realiza bajo la aplicacion de una historia de curvaturas
# Unidades: kN m s
#####

wipe

set Tductility 10 ; # ductilidad meta a la que se llevará la seccion
set npoints 100 ; # número de puntos de analisis
set secT 2 ; # 1 seccion ciclar, 2 seccion cuadrada
```

El código presentado nos brinda la posibilidad de calcular el diagrama momento – curvatura para cualquiera de las dos secciones antes mencionadas con solo cambiar el

valor de la variable “secT”, si damos el valor de 1 se calcula para la sección circular, y si damos el valor de 2 se calcula para la sección cuadrada.

```
set pi [expr acos(-1.0)]

wipe
# Se crea un modelo de dos dimensiones y 3 grados de libertad
model BasicBuilder -ndm 2 -ndf 3
```

Este código a su vez hace uso de otros archivos Tcl: “circularsection.tcl” y “quadsection.tcl” en donde se detallan las características de las secciones que se va a analizar, las propiedades del hormigón, los materiales que forman las fibras tanto para la sección circular como para la rectangular respectivamente, entre otros.

```
switch $secT {
  1 { source circularsection.tcl }
  2 { source quadsection.tcl }
}
#Nudos
node 1 0.0 0.0
node 2 0.0 0.0

# Se restringen todos los grados de libertad en el nudo 1 y
# en el nudo 2 se libera el axial y el momento flector
fix 1 1 1 1
fix 2 0 1 0

#Se define el elemento 0 longitud
# tag ndI ndJ secTag
element zeroLengthSection 1 1 2 1

# Resultados
recorder Node -file Momentocurvatura.txt -time -node 2 -dof 3 disp

# Primero aplicamos la carga axial
pattern Plain 1 "Constant" {
  load 2 -$P 0.0 0.0
}

# Parametros de analisis
integrator LoadControl 0.0
system SparseGeneral -piv;
test NormUnbalance 1.0e-9 10
numberer Plain
constraints Plain
algorithm Newton
```

```
analysis Static
```

```
# Un solo analisis para la carga axial constante  
analyze 1
```

Una vez ejecutado el análisis de carga axial que se aplico en un solo paso y logró comprimir la sección se aplica un momento unitario en el grado de libertad 3 en el nudo 2, este momento se aplica de forma lineal (patttern Plain 2 “Linear”) de manera que pueda ser incrementado durante el análisis. Por último se aplicará un integrador que controla desplazamientos para que controle el desplazamiento del grado de libertad 3 en el nudo 2 para que incremente el desplazamiento en una cantidad igual al valor que se calcula en la variable “curv_inc”. El análisis que se ejecutará será un análisis cíclico, es decir se moverá hacia la derecha ya que “curv_inc” es positiva al inicio, luego se moverá con un “- curv_inc” negativo el doble de puntos que al inicio y por último se volverá a mover con un incremento positivo el doble de puntos igualmente.

```
# Definimos un momento de referencia  
pattern Plain 2 "Linear" {  
load 2 0.0 0.0 1  
}  
  
# Se utiliza un integrador controlado por desplazamiento  
set curv_inc [expr $phyy*$Tductility/$npoints]  
  
# Se aplica curvatura en ambos sentidos para lograr un ciclo completo de respuesta  
integrator DisplacementControl 2 3 $curv_inc  
analyze $npoints  
integrator DisplacementControl 2 3 -$curv_inc  
analyze [expr 2*$npoints]  
integrator DisplacementControl 2 3 $curv_inc  
analyze [expr 2*$npoints]
```

En caso que se desee ejecutar el análisis de manera monótona como lo hacen la mayoría de los programas, se debería eliminar las 4 últimas líneas del código

A continuación se muestra el código de cada una de estas secciones:

Sección Circular:

```
# Datos de seccion para analisis momento curvatura  
# Debe entenderse que la informacion proporcionada sobre los estribos sirve  
unicamente para calcular
```

las propiedades mecánicas de concreto confinado

Se definen las variables que presentan las características de la sección circular, teniendo cuidado de las unidades con que se va a trabajar para evitar inconsistencias en los resultados.

```
set D 0.609 ; # diametro de la seccion (m)
set db 0.019 ;# diametro de las barras longitudinales (m)
set nbl 28 ;# # número de barras longitudinales
set dbt 0.0064 ;# diametro de la espiral (m)
set s 0.025 ;# espaciamiento de la espiral (m)
set cover 0.029 ; # recubrimiento hasta la espiral (m)
set fc 34500 ; # resistencia no confinada del concreto (kPa)
set fy 441300 ; # resistencia de fluencia del acero longitudinal (kPa)
set fyu 602000 ; # resistencia última del acero longitudinal (kPa)
set fyh 606800 ;# resistencia última del acero en la espiral (kPa)
set Es 200000000 ; # modulo de elasticidad del acero
set P 800 ;# carga axial que actua sobre la sección (positivo es compresión) (kN)

# Parametros de discretizacion de la seccion en fibras
set numSubdivCirc 16 ; # número de divisiones circunferenciales
set numSubdivRadcover 4 ; # número de divisiones radiales en el recubrimiento
set numSubdivRadcore 16 ; # número de divisiones radiales en el nucleo

# Opciones para el modelamiento del acero de refuerzo
set RSM 1 ;# 1 para Steel01, 2 para steel02

# Se calcula las propiedades del hormigon confinado de acuerdo al modelo de mander
set eco 0.002
set epss 0.004
set esm 0.1
set Ec [expr 4700.0*sqrt($fc/1000.0)*1000.0]
set Dc [expr $D-2*$cover]
set sp [expr $s-$dbt]
set rcc [expr $db**2*$nbl/$Dc**2]
set ke [expr (1-$sp/2/$Dc)/(1-$rcc)]
set rs [expr $pi*$dbt**2/$Dc/$s]
set fpl [expr 1/2.0*$ke*$rs*$fy]

set fpcc [expr (-1.254 + 2.254 * (1 + 7.94 * $fpl / $fc) ** 0.5 - 2 * $fpl / $fc) * $fc]
set ecc [expr $eco * (1 + 5 * ($fpcc / $fc - 1))]
set Esec [expr $fpcc / $ecc]
set r [ expr $Ec / ($Ec - $Esec)]
set ecu [ expr 1.5 * (0.004 + 1.4 * $rs * $fyh * $esm / $fpcc) ]
set x [expr $ecu / $ecc ]
```

```

set fcu [expr $fpcc * $x * $r / ($r - 1 + $x ** $r)]

puts "La resistencia de hormigon confinado es fpcc: $fpcc"
puts "La defomación cuando se alcanza fpcc es ecc: $ecc"
puts "La resistencia ultima del concreto confinado es fcu: $fcu"
puts "La defomación cuando se alcanza fcu es ecu: $ecu"

```

Se crean los materiales uniaxiales tanto para el concreto confinado como para el concreto sin confinamiento, así como los materiales para las barras de acero longitudinal y transversal.

```

uniaxialMaterial Concrete01 1 -$fc -$eco 0.0 -$epss ; # Material para fibras de
concreto sin confinamiento
uniaxialMaterial Concrete01 2 -$fpcc -$ecc -$fcu -$ecu ; # Material para fibras de
concreto confinado

# Ahora modelamos el acero de refuerzo longitudinal

set b [expr ($fyu-$fy)/(0.1-0.002)/$Es] ; # Modulo post fluencia
switch $RSM {
1 {
uniaxialMaterial Steel01 4 $fy $Es $b
}
2
{
set R0 15
set cR1 0.925
set cR2 0.15
uniaxialMaterial Steel02 4 $fy $Es $b $R0 $cR1 $cR2
}
}

```

Una vez establecidos los materiales se procede a crear la sección de fibras, con el comando “section fiber”, para ello se crean parches para las fibras de hormigón confinado y no confinado, así como la capa de acero longitudinal con el comando “layer”. Esta sección de fibras será capaz de simular el comportamiento axial y a flexión del material. El comportamiento a corte del material tendrá que signarse de forma separada en caso de tratarse de una columna dentro de un modelo estructural de varios elementos, como lo veremos en ejemplos posteriores.

```

# Ahora creamos la seccion de fibras para modelar respuesta a flexion y axial

section Fiber 1 {

```

```
set intRad [expr $Dc/2] ; # Radio interior
set extRad [expr $D/2] ; # Radio exterior
patch circ 1 $numSubdivCirc $numSubdivRadcover 0 0 $intRad $extRad 0 360 ; #
parche de fibras de concreto no confinado
```

“patch circ 1” indica que se creará un parche de fibras con el material 1, así mismo “patch circ 2” con el material 2.

```
set intRad 0
set extRad [expr $Dc/2]
patch circ 2 $numSubdivCirc $numSubdivRadcore 0 0 $intRad $extRad 0 360 ; #
parche de fibras de concreto confinado

set asb [expr $pi*$db**2/4]
layer circ 4 $nbl $asb 0 0 $extRad 0 [expr 360-360/$nbl] ; # Fibras de acero
longitudinal

}

set phyy [expr 2.2*$fy/$Es/$D] ; # Curvatura de fluencia estimada para la seccion que
se utiliza como referencia para calcular la maxima curvatura que se aplicara a la seccion
durante el analisis
```

La sección rectangular tiene características similares que la circular, salvo que por el mismo hecho de ser rectangular la sección es un poco más compleja.

Sección Rectangular:

```
# Datos de seccion para analisis momento curvatura
# Debe entenderse que la informacion proporcionada sobre los estribos sirve
unicamente para calcular
# las propiedades mecánicas de concreto confinado

set b 0.40 ; # base de la sección (eje Y)
set h 0.60 ; # altura de la seccion (eje Z)
set db 0.014 ;# diametro de las barras longitudinales (m)
set nbltb 8 ;# # número de barras arriba y abajo (arriba + abajo)
set nbll 4 ;# # número de barras laterales (todas, izq+der.)
```

La razón por la que se da información sobre los estribos es importante porque a través de la cuantía del refuerzo transversal se puede estimar las propiedades del concreto

confinado, más no para crear fibras que modelen los estribos, ya que una sección de fibras solo modela el comportamiento axial y a flexión de una sección.

```
set dbt 0.008 ;# Diametro de las barras en estribos
set nvsx 2 ;# numero de ramas de estribos resistiendo corte en la direccion X
set nvsy 2 ;# numero de ramas de estribos resistiendo corte en la direccion Y
set s 0.1 ;# espaciamiento de los estribos espiral (m)
set cover 0.04 ;# recubrimiento hasta los estribos (m)
set fc 21000 ;# resistencia no confinada del concreto (kPa)
set fy 420000 ;# resistencia de fluencia del acero longitudinal (kPa)
set fyu 602000 ;# resistencia última del acero longitudinal (kPa)
set fyh 602000 ;# resistencia última del acero en la espiral (kPa)
set Es 200000000 ;# modulo de elasticidad del acero
set P 800 ;# carga axial que actua sobre la sección (positivo es compresión) (kN)

# Parametros de discretizacion de la seccion en fibras
set numSubdivCoreZ 16 ;# número de fibras de concreto confinado en la direccion
local Z
set numSubdivCoreY 16 ;# número de fibras de concreto confinado en la direccion
local Y
set numSubdivCover 2 ;# número de fibras en el recubrimiento

# Opciones para el modelamiento del acero de refuerzo
set RSM 1 ;# 1 para Steel01, 2 para steel02

# Se calcula las propiedades del hormigon confinado de acuerdo al modelo de mander
set eco 0.002
set epss 0.004
set esm 0.1
set Ec [expr 4700.0*sqrt($fc/1000.0)*1000.0]
set sp [expr $s-$dbt]
set rs [expr ($nvsx*$dbt**2*0.875*($b-2*$cover)+$nvsy*$dbt**2*0.875*($h-2*$cover))/((($b-2*$cover)*($h-2*$cover)*$s)]
set areaBar [expr $db**2*0.875]
set areacon [expr ($b-2*$cover)*($h-2*$cover)]
set rcc [expr $areaBar*($nbltb+$nbl1)/$areacon]
set wi 0.2
set ke [expr ((1 - $wi**2 / (6 * $areacon)) * (1 - $s / (2 * $b)) * (1 - $s / (2 * $h))) / (1 - $rcc)]
set fpl [expr 1/2.0*$ke*$rs*$fyh]

set fpcc [expr (-1.254 + 2.254 * (1 + 7.94 * $fpl / $fc) ** 0.5 - 2 * $fpl / $fc) * $fc]
set ecc [expr $eco * (1 + 5 * ($fpcc / $fc - 1))]
set Esec [expr $fpcc / $ecc]
set r [expr $Ec / ($Ec - $Esec)]
set ecu [ expr 1.5 * (0.004 + 1.4 * $rs * $fyh * $esm / $fpcc) ]
```

```

set x [expr $ecu / $ecc ]
set fcu [expr $fpcc * $x * $r / ($r - 1 + $x ** $r)]

puts "La resistencia de hormigon confinado es fpcc: $fpcc"
puts "La defomación cuando se alcanza fpcc es ecc: $ecc"
puts "La resistencia ultima del concreto confinado es fcu: $fcu"
puts "La defomación cuando se alcanza fcu es ecu: $ecu"

uniaxialMaterial Concrete01 1 -$fc -$eco 0.0 -$epss ; # Material para fibras de
concreto sin confinamiento
uniaxialMaterial Concrete01 2 -$fpcc -$ecc -$fcu -$ecu ; # Material para fibras de
concreto confinado

# Ahora modelamos el acero de refuerzo longitudinal

set shr [expr ($fy-$fy)/(0.1-0.002)/$Es] ; # Modulo post fluencia
switch $RSM {
1 {
uniaxialMaterial Steel01 4 $fy $Es $shr
}
2
{
set R0 15
set cR1 0.925
set cR2 0.15
uniaxialMaterial Steel02 4 $fy $Es $shr $R0 $cR1 $cR2
}
}

```

Para la formar la sección se crean 5 rectángulos, uno contendrá las fibras del núcleo confinado y 4 parches rectangulares más pequeños contendrán las fibras del hormigón no confinado en el perímetro de la sección.

```

# Ahora creamos la seccion de fibras para modelar respuesta a flexion y axial

section Fiber 1 {

set y0 [expr $b/2]
set y1 [expr $y0-$cover]
set z0 [expr $h/2]
set z1 [expr $z0-$cover]

set numSubdivCoreZ 16 ; # número de fibras de concreto confinado en la direccion
local Z

```

```

set numSubdivCoreY 16 ; # número de fibras de concreto confinado en la direccion
local Y
set numSubdivCover 2 ; # número de fibras en el recubrimiento

# parches para recubrimiento lateral
patch quad 1 $numSubdivCover [expr $numSubdivCoreZ+2*$numSubdivCover] -$y0 -
$z0 -$y1 -$z0 -$y1 $z0 -$y0 $z0
patch quad 1 $numSubdivCover [expr $numSubdivCoreZ+2*$numSubdivCover] $y1 -
$z0 $y0 -$z0 $y0 $z0 $y1 $z0

# parches para recubrimiento inferior y superior
patch quad 1 $numSubdivCoreY $numSubdivCover -$y1 -$z0 $y1 -$z0 $y1 -$z1 -$y1 -
$z1
patch quad 1 $numSubdivCoreY $numSubdivCover -$y1 $z1 $y1 $z1 $y1 $z0 -$y1 $z0

# parche de fibras para el hormigon confinado
patch quad 2 $numSubdivCoreY $numSubdivCoreZ -$y1 -$z1 $y1 -$z1 $y1 $z1 -$y1
$z1

```

Se usa el comando “layer straight” para crear capas lineales de acero, para ello se definen el número de barras que están en las capas, el área de las varillas que se dará a cada fibra y 2 puntos que definen las coordenadas inicial y final de la capa lineal.

```

# Acero superior e inferior
layer straight 4 [expr $nbltb/2] $areaBar -$y1 -$z1 $y1 -$z1
layer straight 4 [expr $nbltb/2] $areaBar -$y1 $z1 $y1 $z1
# Acero lateral
set sep [expr 2*$z1/($nbl/2+1)]
layer straight 4 [expr $nbl/2] $areaBar -$y1 [expr -$z1+$sep] -$y1 [expr $z1-$sep]
layer straight 4 [expr $nbl/2] $areaBar $y1 [expr -$z1+$sep] $y1 [expr $z1-$sep]
}

set phy [expr 2.1*$fy/$Es/$h] ; # Curvatura de fluencia estimada para la seccion que se
utiliza como referencia para calcular

# la maxima curvatura que se aplicara a la seccion durante el analisis

```

Luego de ejecutar el análisis y al procesar los datos se obtuvo las siguientes gráficas para las secciones cuadrada y circular respectivamente:

Sección rectangular:

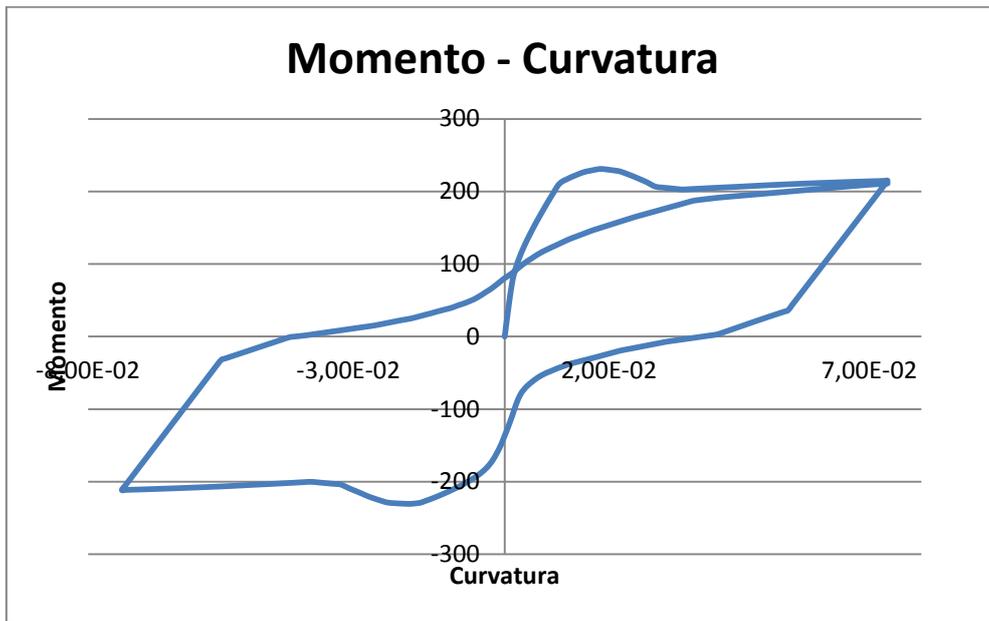


FIGURA 18 – Gráfica momento – curvatura sección rectangular con Opensees

Para la sección circular:

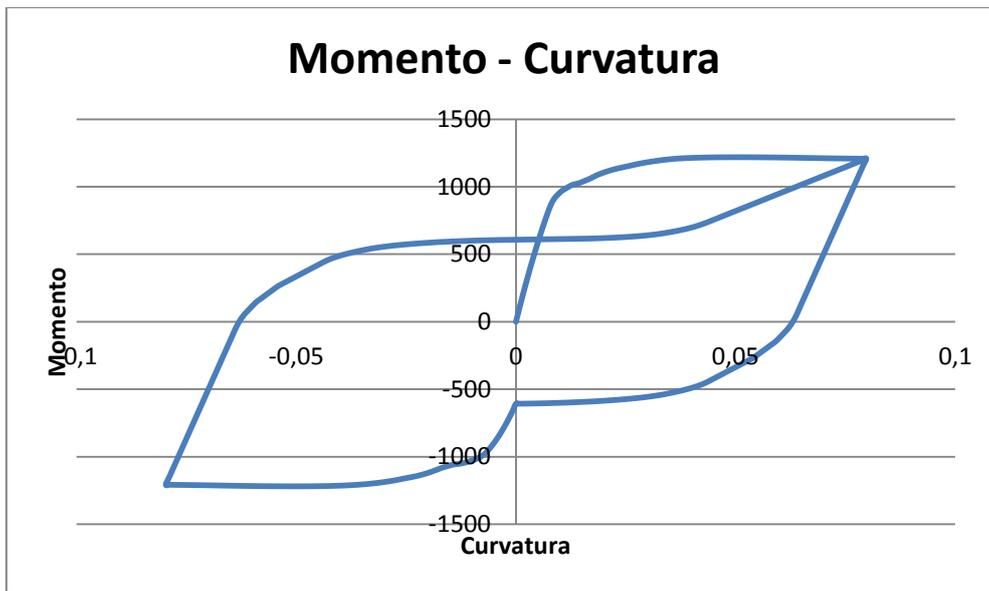


FIGURA 19 – Gráfica Momento – Curvatura sección circular con Opensees

Así mismo mediante el programa RC-Analysis:



LABORATORIO VIRTUAL DE INGENIERÍA SÍSMICA

ANÁLISIS: COLUMNA RECTANGULAR

MOMENTO-CURVATURA Y CAPACIDAD CORTANTE DE SECCIONES DE CONCRETO REFORZADO

PROPIEDADES DE LA SECCIÓN

400 BASE DE LA SECCIÓN B (mm)

600 ALTURA DE LA SECCIÓN H (mm)

40 REQUERIMIENTO AL CENTRO DE LAS VARILLAS SUPERIORES E INFERIORES r' (mm)

40 REQUERIMIENTO AL CENTRO DE LAS VARILLAS LATERALES r (mm)

14 DIÁMETRO DE LAS VARILLAS DE ACERO SUPERIOR E INFERIOR (mm)

8 # NÚMERO DE VARILLAS SUPERIORES E INFERIORES

0 DIÁMETRO DE LAS VARILLAS LATERALES (mm)

0 # VARILLAS DE ACERO LATERAL

2 # DE RAMAS DE ESTRIBO QUE RESISTEN CORTE EN LA DIRECCIÓN "X"

2 # DE RAMAS DE ESTRIBO QUE RESISTEN CORTE EN LA DIRECCIÓN "Y"

8 DIÁMETRO DE LOS ESTRIBOS

100 ESPACIAMIENTO DE LOS ESTRIBOS (mm)

2000 LONGITUD DE CORTE (mm)

800 CARGA AXIAL (kN)

0 ÁNGULO BIAXIAL (θ)

PROPIEDADES DE LOS MATERIALES

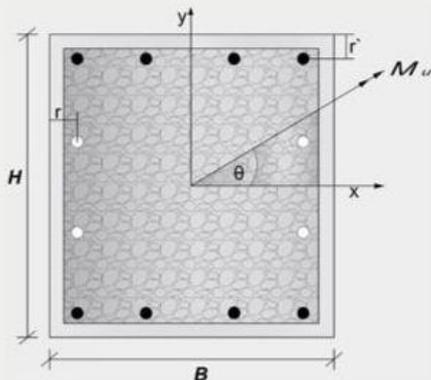
21 RESISTENCIA A LA COMPRESIÓN DEL CONCRETO NO CONFINADO f'_c (Mpa)

420 ESFUERZO DE FLUENCIA DEL ACERO LONGITUDINAL f_y (Mpa)

420 ESFUERZO DE FLUENCIA DEL ACERO TRANSVERSAL f_{yh} (Mpa)

1.35 ÍNDICE DE ENDURECIMIENTO DEL ACERO [> 1]

uniax MODO DE DUCTILIDAD



The diagram shows a rectangular cross-section of a reinforced concrete column. The width is labeled 'B' and the height is 'H'. A coordinate system with x and y axes is centered on the section. Reinforcement is shown with top and bottom bars at distance r' from the center, and side bars at distance r . A diagonal line represents the neutral axis at an angle θ to the horizontal. A moment M_u is applied at this angle. The concrete is represented by a textured pattern, and the steel bars are solid circles.

Figura 20 – Entorno de RC-Analysis para sección rectangular

Diagrama obtenido en RC-Analysis:

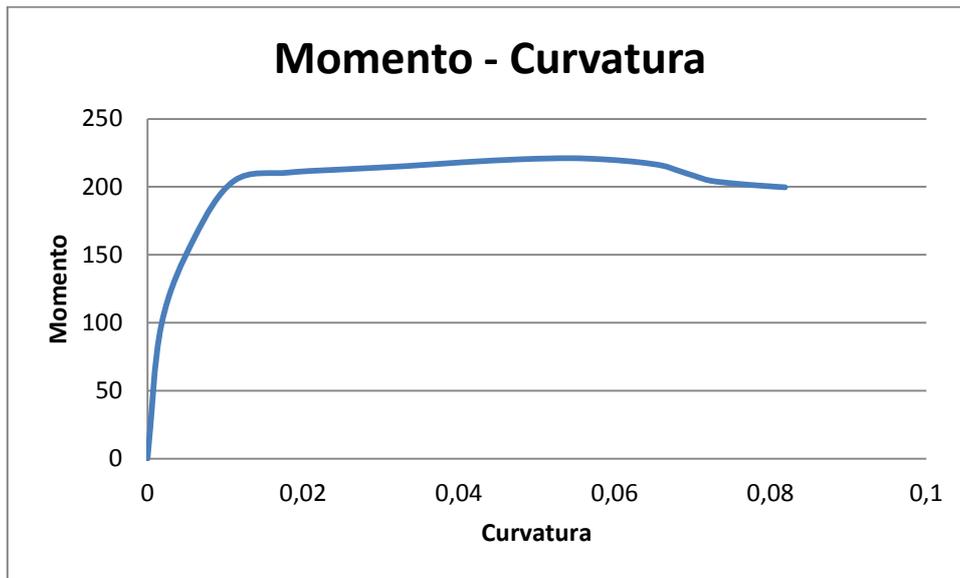


FIGURA 21 – Gráfica momento – curvatura sección rectangular con RC-Analysis

Para la sección circular:

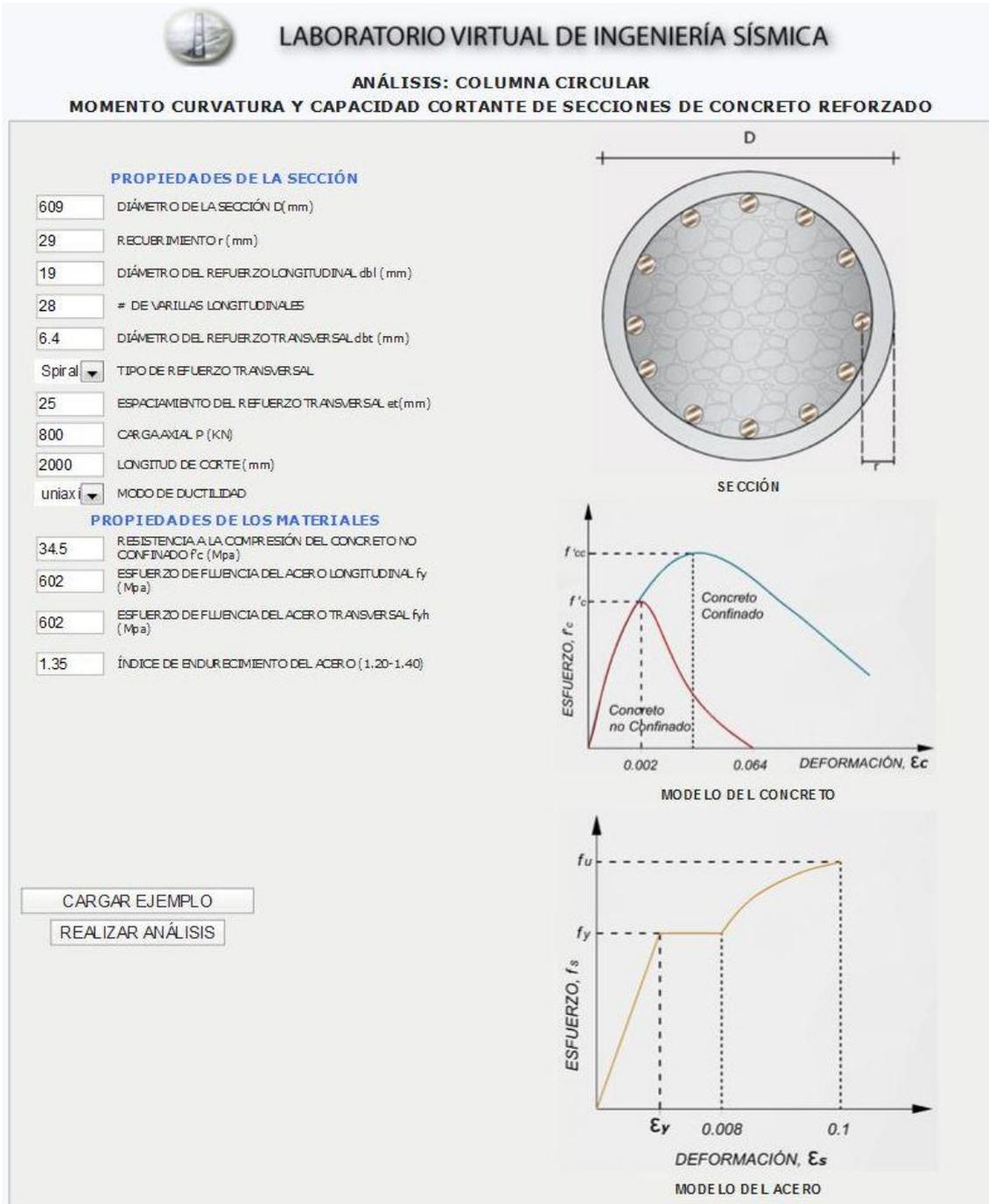


FIGURA 22 - Entorno de RC-Analysis para sección circular

Diagrama obtenido en RC-Analysis:

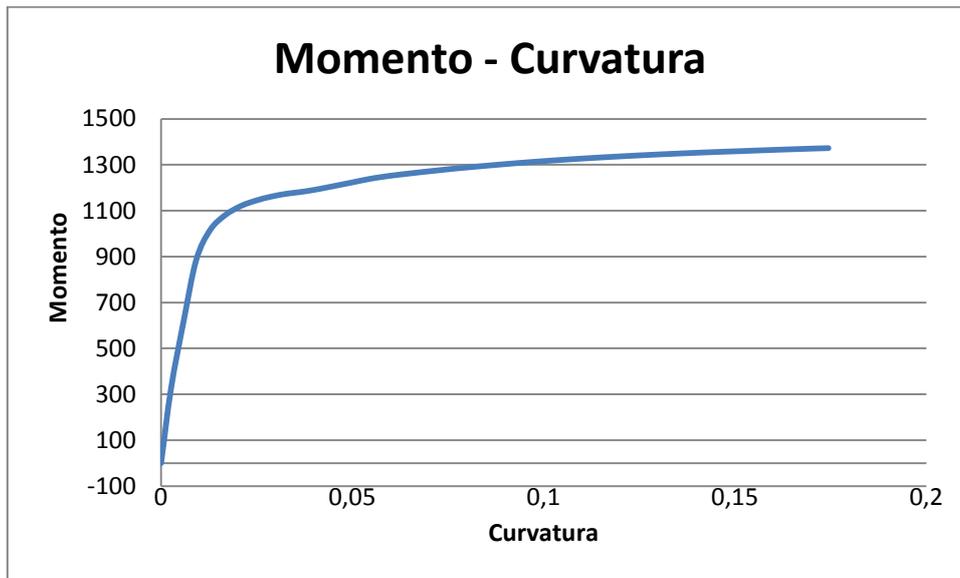


FIGURA 23 – Gráfica Momento – Curvatura sección circular obtenida con RC-Analysis

A continuación se muestran las dos curvas sobrepuestas para lograr una mejor comparación. Para ello se considera solamente una parte de la gráfica obtenida con OpenSees:

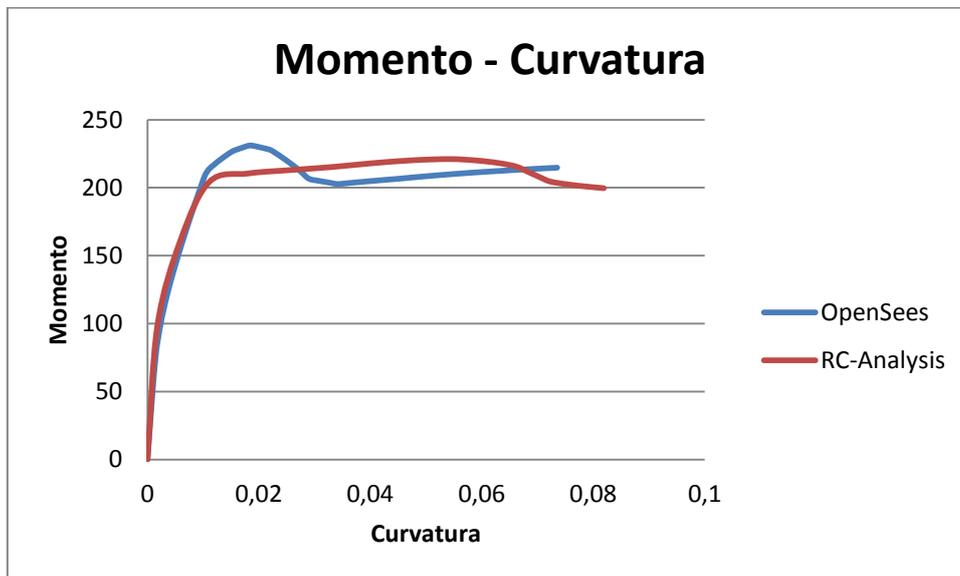


Figura 24 - Comparación Momento-Curvatura sección rectangular

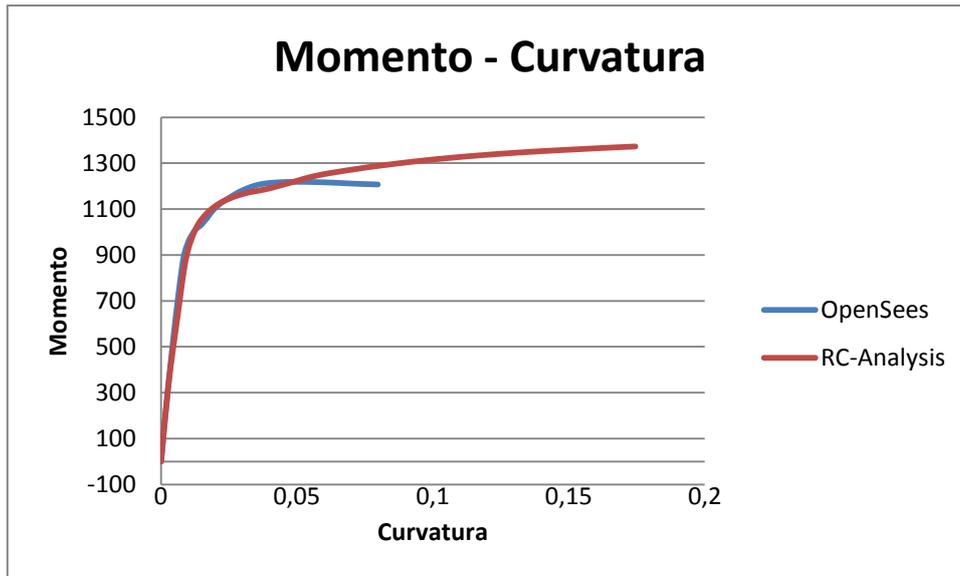


Figura 25 - Comparación Momento-Curvatura sección circular

Con esto se puede comprobar a simple vista que RC-Analysis no realiza un diagrama momento-curvatura para un ciclo completo de carga como lo hace OpenSees, esto hace que el análisis sea más completo y real y le brinda una ventaja mayor que cualquier otro programa.

En cuanto al análisis de las gráficas obtenidas por los programas tanto para la sección rectangular como para la circular, se ve que hay una diferencia en los resultados presentados de los dos programas, esto se debe a que el programa RC-Analysis analiza las secciones considerando sobre-resistencia de materiales para obtener la curva momento-curvatura con acoplamiento de la resistencia a cortante. Esta diferencia es más notoria en la sección rectangular por el mismo hecho de tratarse de una sección no simétrica como en el caso de la sección circular.

CAPÍTULO V

SECCIONES Y ELEMENTOS DE LONGITUD “0”

5. SECCIONES Y ELEMENTOS DE LONGITUD “cero”

5.1.INTRODUCCIÓN

Los elementos y secciones de longitud 0 que están disponibles en OpenSees pueden ser usados para identificar con precisión los efectos de penetración de esfuerzos o las rotaciones en los extremos fijos como los mostrados en la figura 26, desplazamientos y rotación en las uniones del concreto y los miembros de acero como en la base de las columnas y en las uniones viga – columna. Estos elementos y secciones son generalmente usados para análisis de secciones para calcular el momento correspondiente para una determinada curvatura como se vio en el ejemplo anterior.

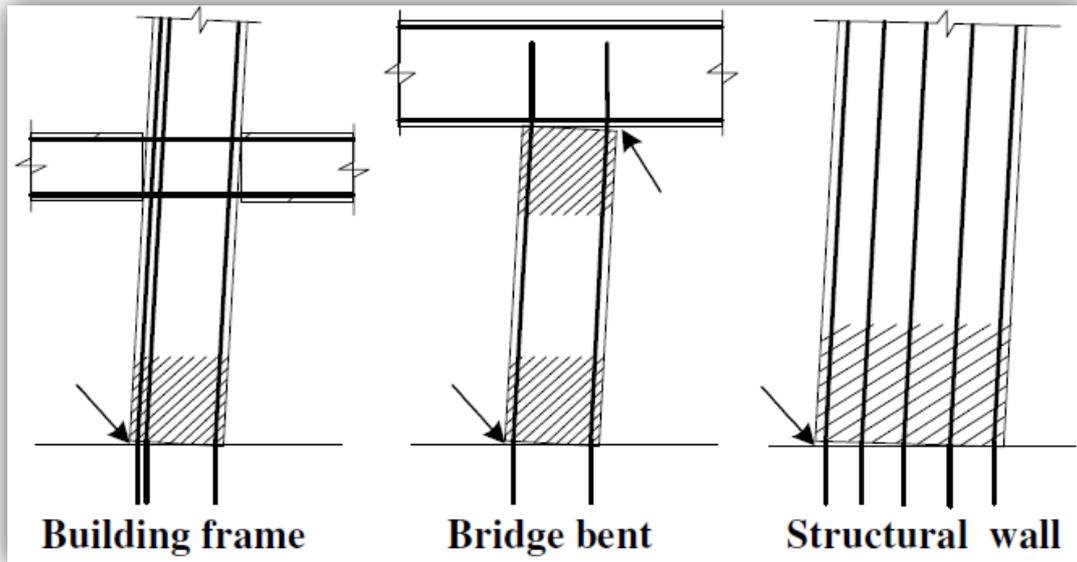


FIGURA 26 – Regiones inelásticas previstas en columnas y pared.

Por ejemplo, para un modelo en que se fija la rotación al final del elemento, el elemento de longitud cero debería colocarse en la intersección entre el miembro flexional y su unión por la base con el otro elemento, tal como se muestra en la figura 27.

También se debe duplicar el nudo en la unión entre la fibra base del elemento viga o columna y el otro elemento adyacente de concreto, como se ve en la figura 27. El grado de libertad de traslación de este nuevo nudo (es decir el nudo j en la figura 27) debería ser fijado al otro nudo (al nudo i de la figura 27) para evitar deslizamientos del elemento viga – columna bajo cargas laterales debido a que los elementos y secciones de longitud cero no poseen la característica de resistencia al cortante.

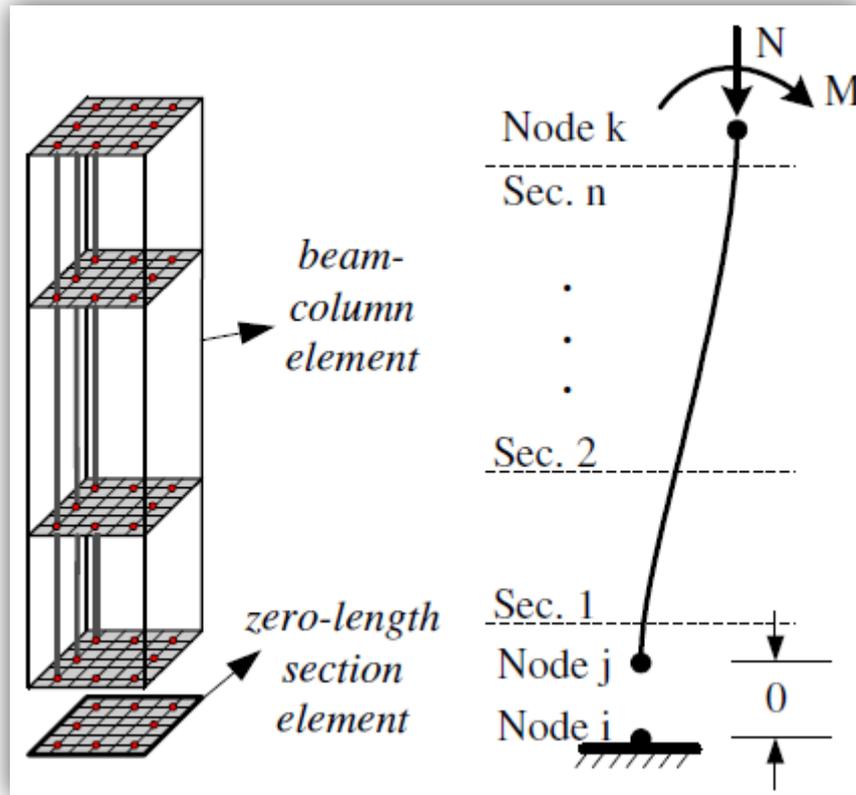


FIGURA 27 – Ubicación de los elementos y secciones de longitud cero.

Se asume que los elementos y secciones de longitud cero tienen una unidad de longitud de tal manera que las deformaciones de los elementos, por ejemplo la elongación y rotación son iguales a las deformaciones en la sección (deformación axial y curvatura). Los elementos y secciones de longitud cero en los modelos de materiales para las fibras de acero representan el deslizamiento de la barra en lugar de la deformación que da lugar a la tensión en las barras. El material uniaxial “**Bond_SP01**” fue desarrollado para añadir elementos de longitud cero en fibras de acero.

5.2.ELEMENTOS DE LONGITUD “0”

Los elementos de longitud cero se usan para conectar o unir dos puntos con las mismas coordenadas. OpenSees posee tres comandos para crear este tipo de elementos, los cuales se detallan a continuación:

5.2.1. Comando: ZERO-LENGTH ELEMENT

Este comando se usa para construir elementos entre dos puntos de iguales coordenadas los cuales están conectados por múltiples materiales uniaxiales (UniaxialMaterial). Representa la relación fuerza – deformación del elemento.

Su sintáxis es la siguiente:

element zeroLength \$eleTag \$iNode \$jNode -mat \$matTag1 \$matTag2 ... -dir \$dir1 \$dir2 ... <-orient \$x1 \$x2 \$x3 \$yp1 \$yp2 \$yp3>

Donde:

\$eleTag	etiqueta única para el material
\$iNode \$jNode	nudos de inicio y fin del elemento
\$matTag1	
\$matTag2 ...	etiquetas de los materiales uniaxiales previamente definidos
\$dir1 \$dir2 ...	dirección del material: 1, 2, 3 traslación a lo largo de los ejes X, Y, Z respectivamente 4, 5, 6 rotación alrededor de los ejes X, Y, Z respectivamente

Se puede especificar la orientación de los vectores para el elemento, sin embargo esto es opcional:

\$x1 \$x2 \$x3	componentes del vector en el sistema global de coordenadas para definir el eje X (vector X)
\$yp1 \$yp2 \$yp3	componentes del vector en el sistema global de coordenadas para definir el vector Yp el cual está en sobre el plano XY del elemento.

El eje local Z es definido por el producto entre los vectores X y Yp.

Si los parámetros opcionales (orientación de vectores) no se especifican, los ejes locales del elemento coinciden con los ejes globales. Las recorders válidos para este tipo de elemento son: fuerza, deformación, rigidez y las respuestas de los materiales, hay que tener en cuenta que el parámetro **\$matNum** corresponde a la etiqueta del material del cual se quiere obtener las respuestas.

5.2.2. Comando: ZERO-LENGTH ND ELEMENT

Se usa para construir elementos entre dos puntos de iguales coordenadas los cuales están conectados por un solo material (NDMaterial). Representa la relación fuerza – deformación del elemento.

Su sintáxis es la siguiente:

element zeroLengthND \$eleTag \$iNode \$jNode \$matTag <\$uniTag> <-orient \$x1 \$x2 \$x3 \$yp1 \$yp2 \$yp3>

Donde:

\$eleTag	etiqueta única para el material
-----------------	---------------------------------

\$iNode \$jNode nudos de inicio y fin del elemento
\$matTag etiquetas del material NDMaterial previamente definido
\$uniTag etiqueta del material uniaxial previamente definido, el mismo que puede ser usado para representar comportamiento de desacople ortogonal para el plano de la respuesta del material NDMaterial.

Si el material NDMaterial es de orden 2, la respuesta del elemento se encuentra sobre el plano local XY y el material uniaxial puede ser usado para representar el comportamiento de desacople ortogonal para este plano, por ejemplo a lo largo del eje local Z.

Si el material NDMaterial es de orden 3, la respuesta estará a lo largo de cada eje local del elemento.

La orientación de los vectores para el elemento pueden o no especificarse:

\$x1 \$x2 \$x3 componentes del vector en el sistema global de coordenadas para definir el eje X (vector X)
\$yp1 \$yp2 \$yp3 componentes del vector en el sistema global de coordenadas para definir el vector Yp el cual está en sobre el plano XY del elemento.

El eje local Z es definido por el producto entre los vectores X y Yp.

Si la orientación de los vectores no se especifica, los ejes locales del elemento coinciden con la orientación de los ejes globales.

Este tipo de elemento únicamente representa la respuesta de traslación entre los nudos. Las recorders válidos para este tipo de elemento son: fuerza, deformación, rigidez y las respuestas del material.

5.2.3. Comando: ZERO-LENGTH SECTION ELEMENT

Se usa para construir elementos entre dos puntos de iguales coordenadas los cuales están conectados por una sola sección (SectionForceDeformation). Representa la relación fuerza – deformación del elemento.

Su sintáxis es la siguiente:

```
element zeroLengthSection $eleTag $iNode $jNode $secTag <-orient $x1 $x2
    $x3 $yp1 $yp2 $yp3>
```

Donde:

\$eleTag	etiqueta única para el material
\$iNode \$jNode	nudos de inicio y fin del elemento
\$secTag	etiqueta de la sección previamente definida.

La orientación de los vectores para el elemento pueden o no especificarse:

\$x1 \$x2 \$x3 componentes del vector en el sistema global de coordenadas para definir el eje X (vector X)

\$yp1 \$yp2 \$yp3 componentes del vector en el sistema global de coordenadas para definir el vector Yp el cual está en sobre el plano XY del elemento.

El eje local Z es definido por el producto entre los vectores X y Yp.

Si la orientación de los vectores no se especifica, los ejes locales del elemento coinciden con la orientación de los ejes globales.

La sección representa la respuesta fuerza – deformación para dicha sección donde la la fuerza axial P actúa a lo largo del eje local X del elemento, y la respuesta para el cortante Vy a lo largo del eje local Y.

Las respuestas válidas para este tipo de elemento cuando se crean los recorders son: fuerza, deformación, rigidez y las respuestas en la sección. Hay que tener en cuenta que este elemento solamente tiene 1 sección, por lo tanto no hay necesidad de identificarla cuando se crea el recorder. Por ejemplo:

```
recorder Element -file Elemento1.out -time -ele 1 section fiber 0.10 0.10 stressStrain
```

5.3.EJEMPLO. CREACIÓN DE ELEMENTOS DE LONGITUD CERO

Con el siguiente ejemplo se pretende mostrar la aplicación de los elementos de longitud cero, estos elementos así como los que se verán en el siguiente capítulo sirven para simular un comportamiento inelástico (próximo capítulo), en este modelo nos remiteremos solamente a realizar un análisis elástico.

Para el siguiente modelo se ha tomado en consideración la geometría de la siguiente figura:

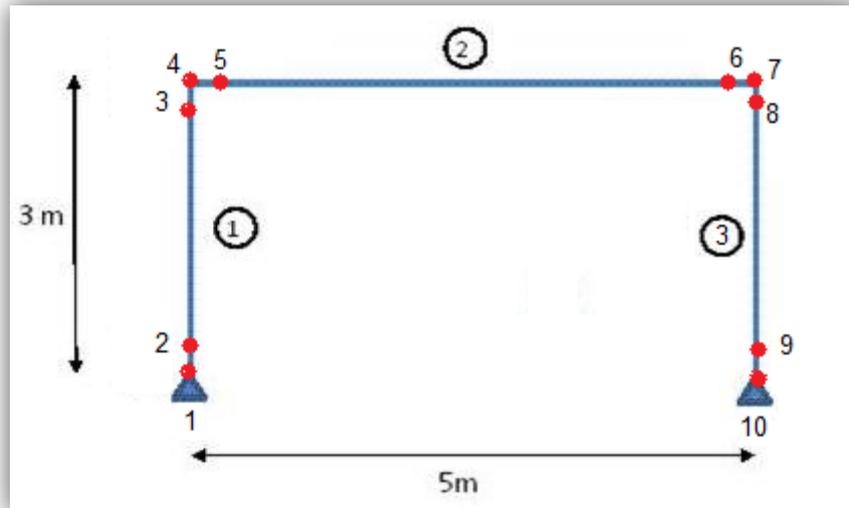


FIGURA 28 – Geometría del modelo

En el gráfico se muestran exageradas las dimensiones de los elementos de longitud cero puesto que ellos no tienen longitud. Esto se lo hizo con el propósito de que la explicación del ejemplo sea más sencilla.

El pórtico será analizado bajo el efecto de la carga gravitacional sobre la viga. En este ejemplo, los elementos elásticos de las columnas y viga tienen un módulo de elasticidad $E = 20000 \text{ MPa}$ un área $A = 0.16 \text{ m}^2$ y una inercia $I = 0.0021 \text{ m}^4$. Sobre el pórtico actúa una carga de 10 kN/m sobre la viga. El objetivo del análisis es determinar los momentos flectores y cortantes en los extremos de vigas y desplazamientos laterales en los nudos 4 y 7. Para formar los elementos de longitud cero se creará un material uniaxial elástico cuyo módulo de elasticidad $E=25000 \text{ MPa}$.

A continuación se presenta el código TCL desarrollado incluyendo notas que explican el uso de los comandos.

Aunque no es obligatorio es recomendable empezar el código explicando el modelo, tipo de análisis, unidades y otra información necesaria para describir el código.

```
#####
#####
#Ejemplo6.tcl                                #
#Análisis 2D de un portico simple con elementos de longitud cero      #
#en los extremos de las columnas y vigas elasticas                    #
#Unidades: kN m s                                                      #
#####
#####
```

Siendo un pórtico plano, son necesarias dos dimensiones y tres grados de libertad por nudo. Una vez definida las dimensiones del modelo se crean los nudos. Como vamos formar elementos de longitud cero es necesario duplicar el nudo en la base y cabeza de las columnas así como en los extremos de las vigas, esto se lo hace de acuerdo a la definición que tiene el comando **element ZeroLength**, a continuación se crean las restricciones, en las cuales se fijan la base de las columnas. Luego como los elementos de longitud cero son “internos” si se podría decir a los elementos principales, se los fija a ellos, para esto usamos el comando **equalDOF**, el cual restringe los grados de libertad de un nudo “esclavo” a otro nudo denominado “maestro”, es decir los grados de libertad que son restringidos en el nudo maestro son igualmente restringidos en el nudo esclavo, por ejemplo: *equalDOF 4 3 1 2* le dice a OpenSees que el nudo maestro es el nudo 4, el nudo esclavo el 3 y que se están restringiendo los grados de libertad 1 y 2.

```
set pi [expr acos(-1.0)]
wipe ;# Este comando borra todos los objetos existentes
model basic -ndm 2 -ndf 3 ;# 2 dimensiones; 3 GDL por nudo

# Coordenadas de los nudos
node 1 0 0
node 2 0 0
node 3 0 3
node 4 0 3
node 5 0 3
node 6 5 3
node 7 5 3
node 8 5 3
node 9 5 0
node 10 5 0

# Condiciones de borde
fix 1 1 1 0
fix 10 1 1 0

# equalDOF $rNodeTag $cNodeTag $dof1 $dof2 ...
equalDOF 1 2 1 2
equalDOF 4 3 1 2
equalDOF 4 5 1 2
equalDOF 7 6 1 2
equalDOF 7 8 1 2
equalDOF 10 9 1 2
```

Se definen las masas de los nudos 4 y 7 en cada uno de los tres grados de libertad del nudo. Ya que en este ejemplo aplicará la aceleración del suelo en la dirección x, solo es necesaria la masa en el primer grado de libertad de cada nudo. En los otros grados de libertad libres puede o no asignarse un valor correcto de masa. Se recomienda no

asignar 0 de masa a ningún grado de libertad libre ya que esto puede ocasionar problemas en la solución. Por ejemplo, si no conozco la masa rotacional de un nudo, le asigno $1e-6$. La masa en los grados de libertad 1 y 2, se calcula de la siguiente manera: Como la carga sobre la viga es de 10kN/m y la longitud de la viga es 5m , entonces la carga en cada nudo es $(10 \times 5)/2 = 25\text{kN}$ como queremos definir la masa en el nudo y no el peso, entonces dividimos para la gravedad: $25/9.81 = 2.54 \approx 2.5$.

Luego creamos el material elástico que formará los elementos de longitud cero, definimos una transformación de coordenadas para los elementos, creamos los elementos elásticos viga – columna y a continuación los elementos de longitud cero tal como tal como se indicó en la Sección 5.2.1 de este capítulo.

```

# Se Define masas por cuanto se va a ejecutar un analisis dinamico
mass 1 0 0 1e-6
mass 10 0 0 1e-6
mass 4 2.5 2.5 1e-6
mass 7 2.5 2.5 1e-6

# Definicion de un material elastico
set E 25000 ;# MPa Modulo de elasticidad
#uniaxialMaterial Elastic $etiqueta $E
uniaxialMaterial Elastic 1 $E

# Definicion de Elementos -----
# Se define una transformacion de coordenadas lineal que se usara con todos los
elementos
geomTransf Linear 1
# Elementos elasticos
element elasticBeamColumn 1 2 3 0.16 20000000 0.0021 1
element elasticBeamColumn 2 5 6 0.16 20000000 0.0021 1
element elasticBeamColumn 3 8 9 0.16 20000000 0.0021 1

# Elementos de longitud cero
#element zeroLength $eleID $iNode $jNode -mat $matTag -dir $dof
element zeroLength 4 1 2 -mat 1 -dir 6
element zeroLength 5 3 4 -mat 1 -dir 6
element zeroLength 6 4 5 -mat 1 -dir 6
element zeroLength 7 6 7 -mat 1 -dir 6
element zeroLength 8 7 8 -mat 1 -dir 6
element zeroLength 9 9 10 -mat 1 -dir 6

```

Se crean los recorders para registrar los fuerzas globales de los elementos y para registrar la historia de desplazamientos de los nudos 4 y 7. A continuación se define el

patrón de cargas para el análisis gravitacional, se definen los parámetros para este análisis y se lo ejecuta.

```
# Registro de resultados -----
# Estos recorders registran la historia en el tiempo de desplazamientos y fuerzas
recorderNode -file Desplazamiento.txt -time -node 4 7 -dof 1 disp recorder Element -
file Viga.txt -time -ele 6 2 7 localForce
recorderElement -file Columnas.txt -time -ele 4 1 5 9 3 8 globalForce
recorder Element -file Esf_Def_Ele6.txt -time -ele 6 material 1 stressStrain
# Estos recorders registran la envolvente de desplazamientos y fuerzas
recorderEnvelopeNode -file DesplazamientoE.txt -time -node 4 7 -dof 1 disp
recorderEnvelopeDrift -file DerivaE.txt -time -iNode 1 -jNode 4 -dof 1 -perpDirn 2
recorderEnvelopeElement -file VigaE.txt -time -ele 6 2 7 localForce
recorderEnvelopeElement -file ColumnasE.txt -time -ele 4 1 5 9 3 8 globalForce

# Aplicacion de cargas -----
pattern Plain 1 Linear {
  eleLoad -ele 2 -type -beamUniform -10
}

# Se ejecuta analisis estatico

constraints Plain
numberer Plain
system BandGeneral
test NormDispIncr 1.0e-8 6
algorithm Linear
integrator LoadControl 1
analysis Static
analyze 1

# Se congela la aplicacion de la carga en todos los patrones definidos hasta ahora
loadConst -time 0.0
```

El análisis transitorio se realizó de igual forma que se hizo para el ejemplo 3 (pórtico 3D), por ello no se da explicación de este procedimiento. Para verificar si el modelo está correcto se obtienen los períodos de los dos primeros modos de vibración, los cuales serán presentados en el intérprete de OpenSees durante la ejecución del análisis.

```
# Define el patron con el que se aplica el registro de aceleraciones
# Se analizara la estructura bajo un sismo que actua en X con una duracion de 38.14
segundos
# cada punto del registro es aceleración (g). El intervalo de tiempo que transcurre entre
cada punto es 0.01s
```

```

set serie1 "Series -dt 0.005 -filePath S1H1.txt -factor 9.81" ;# asigna a la variable serie1
los argumentos del registro de aceleracion
pattern UniformExcitation 2 1 -accel $serie1

# Para asignar amortiguamiento Raleigh, tenemos que obtener los periodos de los dos
modos de vibracion
set lambda1 0.01 ;# amortiguamiento asignado al primer modo de vibración
set lambda2 0.03 ;# amortiguamiento asignado al segundo modo de vibración
set eigenvalues [eigen 2] ;# Obtenemos los valores propios de los primeros 2 modos de
vibracion
set omega1 [expr sqrt([lindex $eigenvalues 0])] ;# Las frecuencias de vibracion son la
raiz cuadrada de los valores propios
set omega2 [expr sqrt([lindex $eigenvalues 1])] ;# Los valores propios estan en una
lista, el comando lindex extrae elementos de la lista
set T1 [expr 2.0*$pi/$omega1] ;# Periodo del primer modo de vibración
puts "T1: $T1 segundos"
set T2 [expr 2.0*$pi/$omega2] ;# Periodo del segundo modo de vibracion
puts "T2: $T2 segundos"
set      alpha      [expr      (2*$omega1*$omega2*($omega1*$lambda2-
$omega2*$lambda1))/($omega1**2-$omega2**2)]
set      beta      [expr      2*($omega1*$lambda1-$omega2*$lambda2)/($omega1**2-
$omega2**2)]

# Se ejecuta analisis dinamico paso a paso en el tiempo
constraints Plain
numberer Plain
system BandGeneral
test NormDispIncr 1.0e-8 6
algorithm Linear
integrator Newmark 0.5 0.25 $alpha $beta 0 0
analysis Transient
# Se fijara el paso del analisis 0.005
# El numero de pasos en el analisis es 7629
analyze 7629 0.005

#####
#FIN
#####

```

Una vez finalizado el proceso de creación del código abrimos el intérprete y ejecutamos el análisis como se muestra en la figura:

```
C:\Users\Rommel\Documents\TESIS\Ejemplos del curso OpenSees\Ejemplo6\OpenSees 2.1.0.exe

OpenSees -- Open System For Earthquake Engineering Simulation
Pacific Earthquake Engineering Research Center -- 2.1.0

(c) Copyright 1999,2000 The Regents of the University of California
All Rights Reserved
(Copyright and Disclaimer @ http://www.berkeley.edu/OpenSees/copyright.html)

OpenSees >source Ejemplo6.tcl
T1: 0.3316141605566339 segundos
T2: 0.009619123350466982 segundos
0
OpenSees >_
```

FIGURA 29 – Ejecución del ejemplo6.tcl

De donde se obtienen los siguientes resultados:

T1: 0.3316

T2: 0.0096

Lo que quiere decir que el modelo se ejecutó correctamente. A continuación se muestran algunas gráficas obtenidos luego de procesar los datos con ayuda de Excel. Con los archivos creados podemos obtener:

- La historia de desplazamientos del nudo 4 y 7 con el archivo Desplazamiento.txt

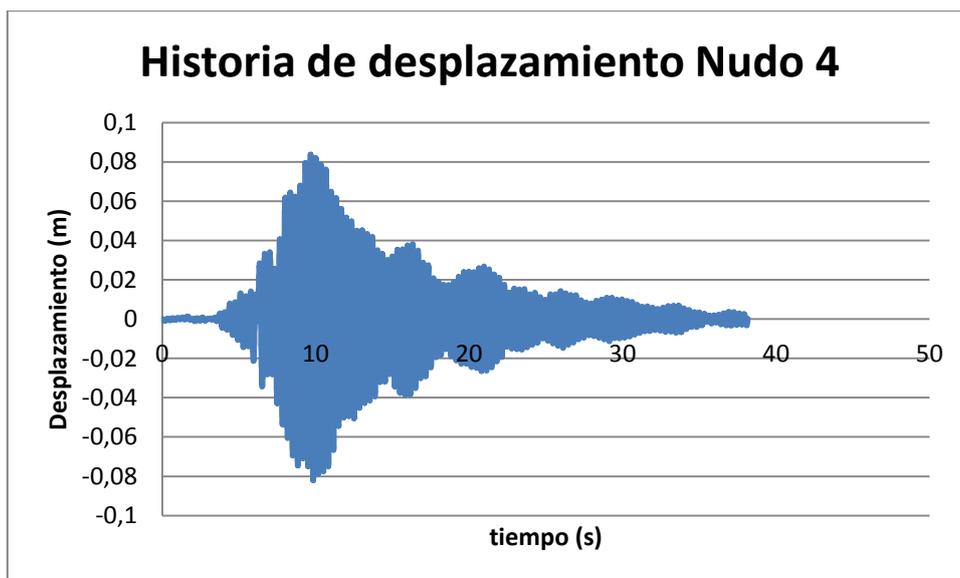


FIGURA 30 – Historia de desplazamiento nudo 4 en dirección X

- Los valores de desplazamiento máximo de los nudos 4 y 7 con el archivo DesplazamientoE.txt
 9.83 -0.0819475 9.83 -0.0819515
 9.665 0.0836598 9.665 0.0836559
 9.665 0.0836598 9.665 0.0836559

Donde los valores máximos de desplazamiento hacia la izquierda se muestran en la primera fila, los máximos hacia la derecha en la segunda columna y los máximos absolutos en la tercera fila.

- Fuerzas locales en la viga con el archivo Viga.txt

La primera fila del archivo contiene las fuerzas al final de la aplicación de las cargas gravitacionales y el resto de filas corresponde a la historia de fuerzas durante el sismo. Luego de la primera columna de tiempo, las seis siguientes registran las fuerzas en la viga: axial en extremo i, cortante i, momento i , axial j , cortante j, momento j. Como ejemplo, se presenta la historia de momentos en el extremo inicial (izquierdo) de la viga:

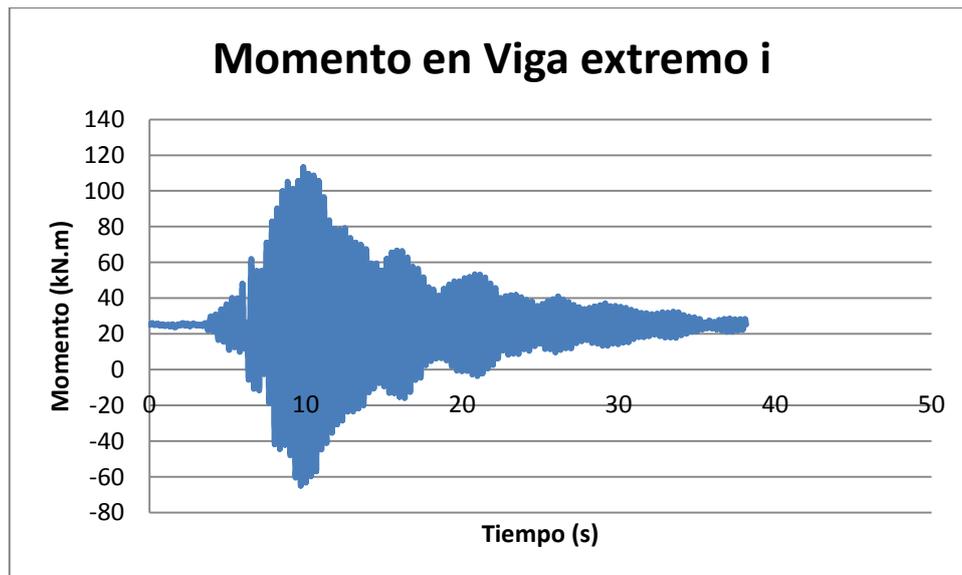


FIGURA 31 – Momento en Viga extremo i

- Fuerzas globales en las columnas con el archivo Columnas.txt.

Los resultados de las columnas se presentan de la misma forma que para la viga, a continuación se presenta el cortante de la columna 3 en el extremo inferior (extremo i):

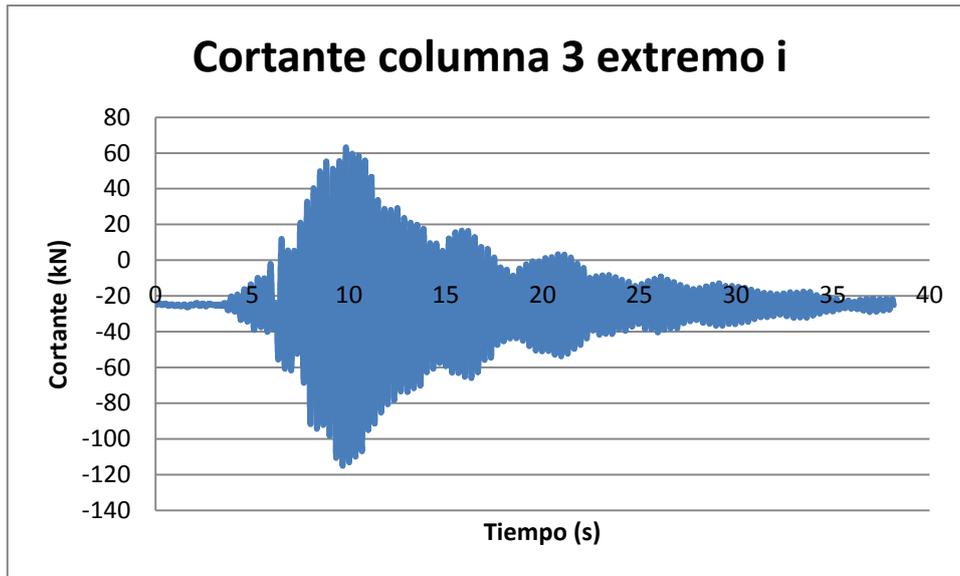


FIGURA 32 – Cortante en columna 3 extremo i

- Histéresis del material de los elementos de longitud cero.

Como ejemplo se muestra la histéresis del material elástico del elemento 6. Como es de esperarse el comportamiento es lineal, no hay deformaciones permanentes ni disipación de energía (el área dentro de la curva es cero).

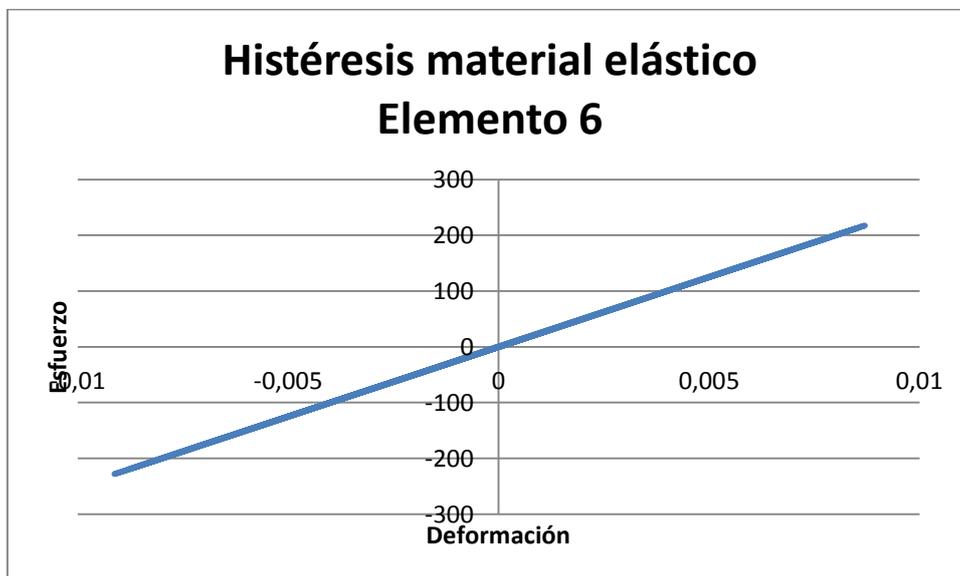


FIGURA 33 – Histéresis del material del elemento 6

Así concluye la explicación de este ejemplo.

El otro tipo de elemento “**element zeroLengthND**” puede así mismo usarse para crear estructuras como la que se acaba de ver en este ejemplo. En este ejemplo pudo también haberse hecho uso del elemento **element zeroLengthSection** en lugar del que se usó tal

como se vio en el ejemplo en que se realizó el análisis momento curvatura de una sección.

En el siguiente capítulo se realizarán análisis no lineales de estructuras con elementos no lineales, en donde esas mismas estructuras pueden ser modeladas de igual forma que se hizo en este ejemplo considerando pequeños cambios para simular un comportamiento inelástico.

CAPÍTULO VI

ANÁLISIS NO-LINEAL DE ESTRUCTURAS APORTICADAS

6. PÓRTICOS EN EDIFICIOS Y PUENTES

6.1.INTRODUCCIÓN

La mayoría de edificios y puentes son diseñados para exceder el límite elástico bajo el ataque del terremoto de diseño, aunque esto signifique daño permanente en las secciones de vigas y columnas.

La estructura debe ser capaz de soportar y transferir a la cimentación las cargas que recibe. El diseño estructural debe partir por tanto de la identificación de las cargas, y a partir de ahí escoger aquella tipología estructural que sea capaz de transferir adecuadamente las cargas. En general la respuesta nunca será única, ni en lo referente al tipo de estructura, ni en lo referente a los materiales a emplear ni en su construcción.

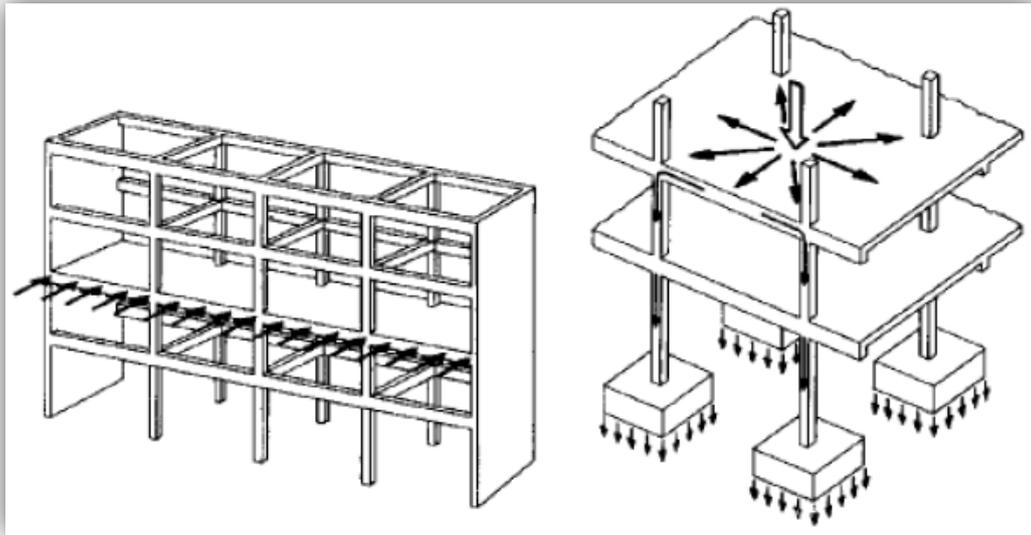


FIGURA 34 – Solicitaciones horizontales y verticales en una estructura

Hoy en día se tienen programas de cálculo poderosos, los análisis no lineales están casi al alcance de todos, pero la única restricción es el precio. El cálculo no lineal se realiza para el cálculo del desempeño de cualquier estructura y así verificar factores como ductilidad, proceso de formación de rótulas, etc. De esta manera se obtiene la respuesta ante una sollicitación que puede ser un sismo u otra sollicitación a la que nuestra estructura tenga que soportar durante su vida útil.

Para lograr un análisis no-lineal, se pueden emplear los siguientes métodos:

- Métodos clásicos de análisis no lineal por el material a nivel global: Métodos basados en la definición de rótulas plásticas.
- Métodos clásicos para el análisis no lineal por la geometría: Método de las funciones de estabilidad.
- Método de los elementos finitos en análisis no lineal.
- Otros

En nuestro caso y como se ha mencionado desde un inicio OpenSees es una plataforma basada en el método de los elementos finitos, por ello nos referiremos a este.

El método de los elementos finitos consiste en la descomposición del continuo de la estructura en un número discreto de partes o elementos, conectados entre si mediante un número igualmente discreto de puntos, denominados nodos. Los movimientos de los nodos constituyen las incógnitas fundamentales del problema. Dentro de cada elemento, los movimientos de puntos cualquiera mediante una interpolación de los movimientos nodales. Las restantes variables o resultados del problema estructural (deformaciones, tensiones, esfuerzos,...), son obtenidos como consecuencia del empleo de la

mencionada interpolación de los movimientos al establecer las condiciones de equilibrio y compatibilidad y las relaciones constitutivas de los materiales.

6.2.PÓRTICOS EN EDIFICOS

Basados en lo mencionado en el párrafo anterior para el siguiente ejemplo se pretende ejecutar un análisis Pushover estático controlado por desplazamientos. Un Pushover es un análisis no-lineal, en el cual la magnitud de la carga lateral aplicada a la estructura se va incrementando de acuerdo a un patrón predefinido de carga (puede ser triangular o uniformemente distribuida). Un análisis pushover es usado para averiguar cual será la fuerza real que soportará la estructura hasta alcanzar un desplazamiento meta. Éste análisis es muy útil para el diseño basado en desempeño.

6.2.1. Ejemplo. Análisis pushover 3D de un pórtico de 3 pisos con 2 vanos en ambas direcciones.

En este ejemplo se modifica el código del ejemplo 3 para construir el nuestro y correr un análisis pushover. En este ejemplo se introducen nuevos conceptos y se hace uso de nuevos comandos que se detallaran conforme avance el ejemplo.

Al igual que en los ejemplos anteriores es necesario idealizar la estructura. Así tenemos que la estructura se diseña con las siguientes características:

- El pórtico 3D que se analiza en este ejemplo se basa en la geometría de las figuras 22, 23 y 24 presentadas a continuación. El pórtico tiene en el sentido “X” dos vanos de 3m y 6m, y en sentido “Y” dos vanos de 4m y 5m. Tiene tres niveles de 3.5 m cada uno. El pórtico está hecho con hormigón de resistencia a la compresión $f'_c = 24$ MPa. Las columnas son cuadradas de 0.40m de lado. Las vigas tiene 0.30 m de base y 0.50 m de altura. Los entrepisos son losas nervadas de hormigón armado que pesan 6 kN/m².
- El modelo tiene 39 nudos (los nudos 37, 38 y 39 son los centros de masa del piso uno, dos y tres respectivamente) y 63 elementos, los 27 primeros corresponden a las columnas y los restantes a las vigas.
- Se crean secciones de fibras para las secciones de columnas y vigas.
- Se modela las vigas y columnas como elementos con articulaciones en los extremos, es decir se crearán rótulas plásticas. El módulo elástico del hormigón se toma como base para definir las propiedades mecánicas de las secciones. Para considerar el agrietamiento de las vigas y columnas se aplicará un coeficiente de reducción de inercia de 0.5 a las vigas y de 0.7 a las columnas.
- Se simularán pisos rígidos mediante la definición de diafragmas en cada piso en donde el nudo en el centro de rigidez actúa como nudo maestro y los demás nudos en el piso actúan como nudos esclavos.

- Se aplicaran sobre las vigas y columnas cargas uniformemente distribuidas para representar adecuadamente el peso propio de la estructura.
- No se considerará el aporte de los entrepisos a la rigidez lateral de la estructura.
- La masa de la estructura se distribuirá apropiadamente en sus nudos
- Las propiedades del concreto confinado serán calculadas en tiempo de ejecución utilizando el modelo de Mander.
- Para poder ejecutar el análisis pushover primero se requiere ejecutar un análisis gravitacional.

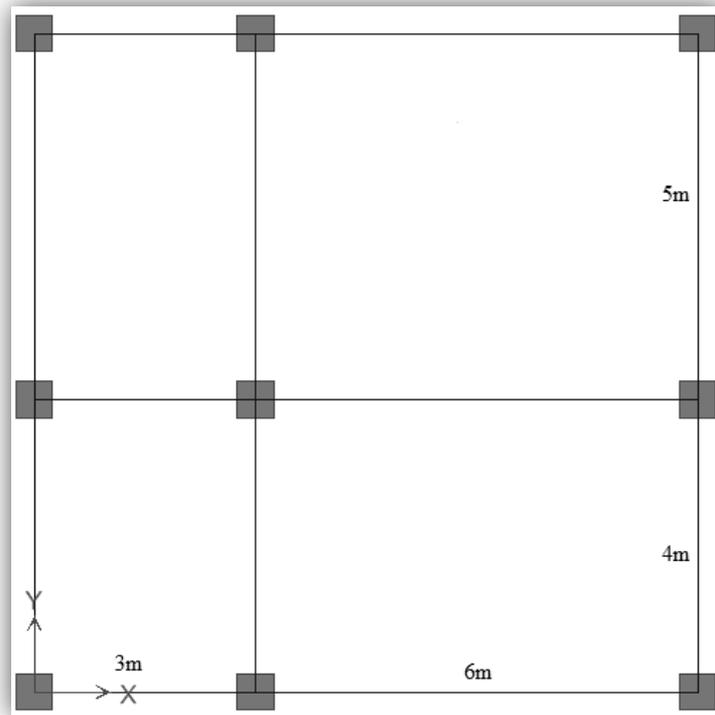


FIGURA 35 – Vista en planta del modelo

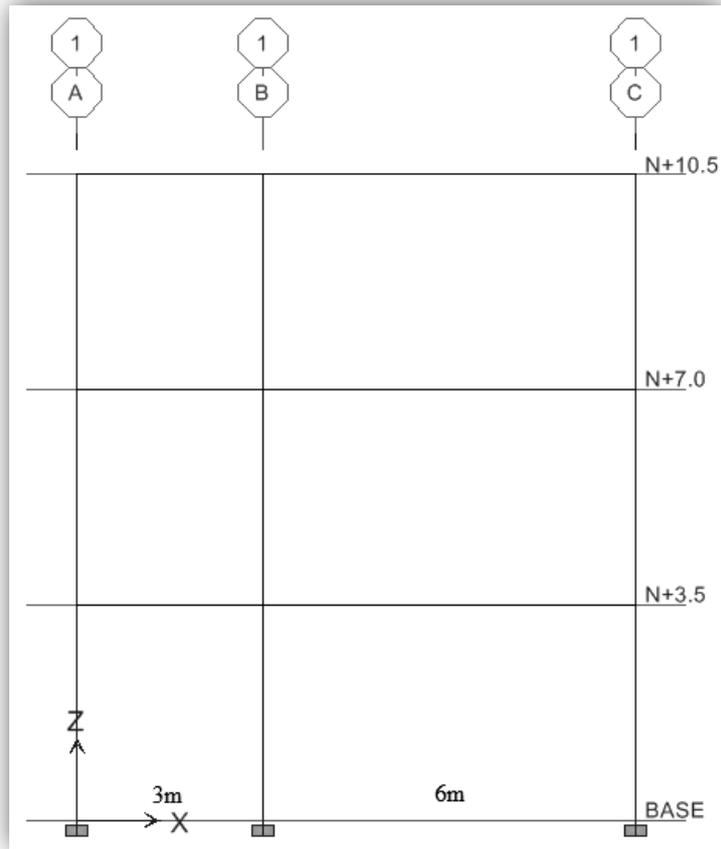


FIGURA 36 – Elevación plano XZ del modelo

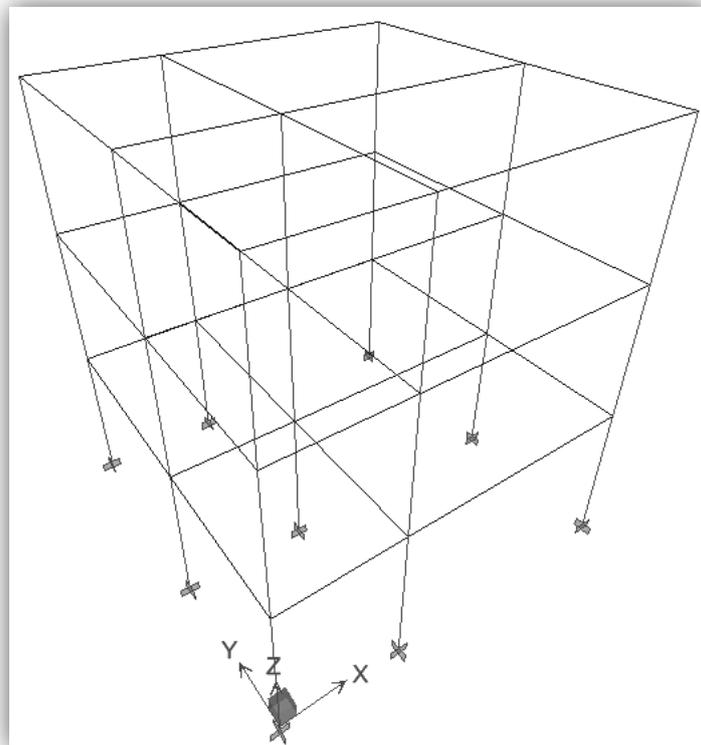


FIGURA 37 – Vista 3D del modelo

Como en todo modelo se empieza con la creación de los nudos del edificio. El orden que se ha seguido es numerar los nudos por planta desde el primer piso hasta el tercero. Así mismo se definen los centros de masa (nudo 37, 38 y 39).

```
#####
#Ejemplo7.tcl
#Análisis 3D PUSHOVER de un portico de 3 pisos
#####

# Creación del modelo
# -----
wipe ;# Este comando borra todos los objetos existentes
model BasicBuilder -ndm 3 -ndf 6 ;# 3 dimensiones; 6 GDL por nudo

# Se define la geometría -----
# Coordenadas de los nudos
node 1 0 0
node 2 3 0
node 3 9 0
node 4 0 4
node 5 3 4
node 6 9 4
node 7 0 9
node 8 3 9
node 9 9 9
node 10 0 3.5
node 11 3 3.5
node 12 9 3.5
node 13 0 4 3.5
node 14 3 4 3.5
node 15 9 4 3.5
node 16 0 9 3.5
node 17 3 9 3.5
node 18 9 9 3.5
node 19 0 7
node 20 3 7
node 21 9 7
node 22 0 4 7
node 23 3 4 7
node 24 9 4 7
node 25 0 9 7
node 26 3 9 7
node 27 9 9 7
node 28 0 10.5
node 29 3 10.5
node 30 9 10.5
```

```

node 31 0 4 10.5
node 32 3 4 10.5
node 33 9 4 10.5
node 34 0 9 10.5
node 35 3 9 10.5
node 36 9 9 10.5

# Nudos maestros
node 37 4.5 4.5 3.5
node 38 4.5 4.5 7
node 39 4.5 4.5 10.5

```

Se crean los pisos rígidos para las tres plantas. Se empotran las columnas del primer piso, para ello se bloquea todos los 6 gdl de la base de las columnas, además se bloquea los grados de libertad 3 4 5 de los nudos maestros ya que no existe rigidez asociada a estos grados de libertad en los diafragmas.

```

# Creo los pisos rigidos con el comando rigidDiaphragm
# En el primer piso, el nudo maestro es el nudo 37 y los nudos esclavos son 10,
11,12,... 18
rigidDiaphragm 3 37 10 11 12 13 14 15 16 17 18
# En el segundo piso, el nudo maestro es el nudo 38 y los nudos esclavos son 19, 20,
21,... 27
rigidDiaphragm 3 38 19 20 21 22 23 24 25 26 27
# En el tercer piso, el nudo maestro es el nudo 39 y los nudos esclavos son 28, 29, 30,...
36
rigidDiaphragm 3 39 28 29 30 31 32 33 34 35 36

# Condiciones de borde - Las columnas estan empotradas en su base
# -----
fix 1 1 1 1 1 1 1
fix 2 1 1 1 1 1 1
fix 3 1 1 1 1 1 1
fix 4 1 1 1 1 1 1
fix 5 1 1 1 1 1 1
fix 6 1 1 1 1 1 1
fix 7 1 1 1 1 1 1
fix 8 1 1 1 1 1 1
fix 9 1 1 1 1 1 1
fix 37 0 0 1 1 1 0 ; # Se liberan el desplazamiento en "x" y "y" y el momento Mz, en los
nudos maestros
fix 38 0 0 1 1 1 0
fix 39 0 0 1 1 1 0

```

Considerando el peso propio de los elementos y el peso del entrepiso se estima una carga uniforme sobre las vigas de 15 kN/m y una carga uniforme sobre las columnas de 3.84 kN/m. Estas cargas actúan en la dirección $-Z$ para representar los efectos gravitacionales. A partir de estas cargas y de las longitudes de vigas y columnas se estima una masa concentrada en los nudos de 8 t.

Se Define masas a partir de la carga sobre las vigas

```
mass 10 8 8 8 1e-6 1e-6 1e-6
mass 11 8 8 8 1e-6 1e-6 1e-6
mass 12 8 8 8 1e-6 1e-6 1e-6
mass 13 8 8 8 1e-6 1e-6 1e-6
mass 14 8 8 8 1e-6 1e-6 1e-6
mass 15 8 8 8 1e-6 1e-6 1e-6
mass 16 8 8 8 1e-6 1e-6 1e-6
mass 17 8 8 8 1e-6 1e-6
mass 18 8 8 8 1e-6 1e-6
mass 19 8 8 8 1e-6 1e-6 1e-6
mass 20 8 8 8 1e-6 1e-6 1e-6
mass 21 8 8 8 1e-6 1e-6 1e-6
mass 22 8 8 8 1e-6 1e-6 1e-6
mass 23 8 8 8 1e-6 1e-6 1e-6
mass 24 8 8 8 1e-6 1e-6 1e-6
mass 25 8 8 8 1e-6 1e-6 1e-6
mass 26 8 8 8 1e-6 1e-6 1e-6
mass 27 8 8 8 1e-6 1e-6 1e-6
mass 28 8 8 8 1e-6 1e-6 1e-6
mass 29 8 8 8 1e-6 1e-6 1e-6
mass 30 8 8 8 1e-6 1e-6 1e-6
mass 31 8 8 8 1e-6 1e-6 1e-6
mass 32 8 8 8 1e-6 1e-6 1e-6
mass 33 8 8 8 1e-6 1e-6 1e-6
mass 34 8 8 8 1e-6 1e-6 1e-6
mass 35 8 8 8 1e-6 1e-6 1e-6
mass 36 8 8 8 1e-6 1e-6 1e-6
```

```
set pi [expr acos(-1.0)]
```

A partir del valor de f'_c y del valor de la relación de poisson para el hormigón se calculan los valores del modulo elástico E y modulo de corte G. También se calculan las propiedades mecánicas: Inercia en el eje local z “Iz”, inercia en el eje local y “Iy”, área A y constante de torsión, J, para las secciones de vigas y columnas, así mismo se calculan las propiedades para el hormigón confinado de acuerdo al modelo de mander.

```

# DEFINICIÓN DE ELEMENTOS
#Propiedades del hormigón y acero:
set fc 24000 ; # resistencia a la compresión no confinada del concreto (kPa)
set fy 420000 ; # resistencia de fluencia del acero longitudinal (kPa)
set fyu 602000 ; # resistencia última del acero longitudinal (kPa)
set fyh 602000 ;# resistencia última del acero de estribos (kPa)
set Es 200000000 ; # modulo de elasticidad del acero
set shr 0.003 ; # razón de rigidez post fluencia (parámetro a usar para definir el material Steel01)
set poisson 0.25 ;# Relacion de poisson para el concreto
set E [expr 4700*sqrt($fc/1000)*1000]
set G [expr $E/(2*(1+$poisson))]
set loadviga 20 ; # carga distribuida en sentido gravitacional a ser aplicada sobre las vigas (kN/m)

# Propiedades mecanicas de las secciones:
# COLUMNAS
set bcol 0.40 ;# columnas cuadradas
set RIcol 0.70 ;# reduccion de inercia
set Acol [expr $bcol**2]
set Iycol [expr $RIcol*$bcol**4/12]
set Izcol $Iycol
set bjc 0.14 ;# Relacion entre lado largo y lado corto para calculo de Jcol(momento torsional de inercia)
set Jcol [expr $bjc*$bcol**4]
set dbc 0.025 ;# diametro de las barras longitudinales (m)
set nbltbc 8 ;# # número de barras arriba y abajo (arriba + abajo)
set nbllc 4 ;# # número de barras laterales (todas, izq+der.)
set dbtc 0.01 ;# Diametro de las barras en estribos
set nvsvc 2 ;# numero de ramas de estribos resistiendo corte en la direccion X
set nvsvy 2 ;# numero de ramas de estribos resistiendo corte en la direccion Y
set sc 0.2 ;# espaciamiento de los estribos(m)
set coverc 0.04 ; # recubrimiento hasta los estribos (m)

# Propiedades del hormigon confinado a usar en la sección de columnas de acuerdo al modelo de mander,
set ecocol 0.002 ; # Deformacion unitaria a maxima resistencia
set epssc 0.004 ; # Deformacion de trituracion del hormigon
set esmcol 0.1
set spcol [expr $sc-$dbtc]
set rscol [expr ($nvsvc*$dbtc**2*(pi/4)*($bcol-2*$coverc)+$nvsvy*$dbtc**2*(pi/4)*($bcol-2*$coverc))/((($bcol-2*$coverc)*($bcol-2*$coverc)*$sc)]
set areaBarcol [expr $dbc**2*(pi/4)]
set areaconcol [expr ($bcol-2*$coverc)*($bcol-2*$coverc)]
set rccol [expr $areaBarcol*($nbltbc+$nbllc)/$areaconcol]
set wic 0.2

```

```

set kec [expr ((1 - $wic**2 / (6 * $areaconcol)) * (1 - $sc / (2 * $bcol)) * (1 - $sc / (2 * $bcol))) / (1 - $rccol)]
set fplcol [expr 1/2.0*$kec*$rscol*$fyh]
set fpccol [expr (-1.254 + 2.254 * (1 + 7.94 * $fplcol / $fc) ** 0.5 - 2 * $fplcol / $fc) * $fc]
set eccol [expr $ecocol * (1 + 5 * ($fpccol / $fc - 1))]
set Esec [expr $fpccol / $eccol]
set rcol [ expr $E / ($E - $Esec)]
set ecucol [ expr 1.5 * (0.004 + 1.4 * $rscol * $fyh * $esmcoll / $fpccol) ]
set xcol [expr $ecucol / $eccol ]
set fcucol [expr $fpccol * $xcol * $rcol / ($rcol - 1 + $xcol ** $rcol)]

```

VIGAS

```

set bviga 0.3 ;# base (eje y)
set hviga 0.5 ;# altura (eje z)
set RIviga 0.5 ;# reduccion de inercia
set Aviga [expr $bviga*$hviga]
set Iyviga [expr $RIviga*$bviga*$hviga**3/12]
set Izviga [expr $RIviga*$bviga*$bviga**3/12]
set bj 0.20 ;# Relacion entre lado largo y lado corto para calculo de Jviga (momento torsional de inercia)
set Jviga [expr $bj*$hviga**$bviga**3]
set db 0.016 ;# diametro de las barras longitudinales (m)
set nbltb 8 ;# # número de barras arriba y abajo (arriba + abajo)
set nbll 4 ;# # número de barras laterales (todas, izq+der.)
set dbt 0.01 ;# Diametro de las barras en estribos
set nvxs 2 ;# numero de ramas de estribos resistiendo corte en la direccion X
set nvys 2 ;# numero de ramas de estribos resistiendo corte en la direccion Y
set sv 0.1 ;# espaciamiento de los estribos (m)
set cover 0.04 ; # recubrimiento hasta los estribos (m)

```

Propiedades del hormigon confinado a usar en la sección de vigas de acuerdo al modelo de mander,

```

set eco 0.002
set epss 0.004
set esm 0.1
set sp [expr $sv-$dbt]
set rs [expr ($nvxs*$dbt**2*(pi/4)*($bviga-2*$cover)+$nvys*$dbt**2*(pi/4)*($hviga-2*$cover))/((($bviga-2*$cover)*($hviga-2*$cover)*$sv)]
set areaBar [expr $db**2*(pi/4)]
set areacon [expr ($bviga-2*$cover)*($hviga-2*$cover)]
set rcc [expr $areaBar*( $nbltb+$nbll)/$areacon]
set wi 0.2
set ke [expr ((1 - $wi**2 / (6 * $areacon)) * (1 - $sv / (2 * $bviga)) * (1 - $sv / (2 * $bviga))) / (1 - $rcc)]
set fpl [expr 1/2.0*$ke*$rs*$fyh]
set fpcc [expr (-1.254 + 2.254 * (1 + 7.94 * $fpl / $fc) ** 0.5 - 2 * $fpl / $fc) * $fc]

```

```

set ecc [expr $eco * (1 + 5 * ($fpcc / $fc - 1))]
set Esec [expr $fpcc / $ecc]
set r [ expr $E / ($E - $Esec)]
set ecu [ expr 1.5 * (0.004 + 1.4 * $rs * $fyh * $esm / $fpcc) ]
set x [expr $ecu / $ecc ]
set fcu [expr $fpcc * $x * $r / ($r - 1 + $x ** $r)]

```

Se definen los materiales a usar tanto en las vigas como en las columnas, como lo que se quiere lograr es formar elementos lo más reales posibles, se crean cuatro tipos de materiales que se usarán para el hormigón confinado, no confinado, para el acero de refuerzo y un material elástico para añadir respuesta a cortante a la sección.

```

# DEDINICIÓN DE MATERIALES:
# -----
# Materiales para columnas
uniaxialMaterial Concrete01 1 -$fc -$ecocol 0 -$epsscol ; # Material para fibras de
concreto sin confinamiento
uniaxialMaterial Steel01 2 $fy $Es $shr ; # Material para acero de refuerzo longitudinal
uniaxialMaterial Concrete01 3 -$fpccol -$eccol -$fcucol -$ecucol ; # Material para
fibras de concreto confinado
uniaxialMaterial Elastic 4 $E

# Materiales para vigas
uniaxialMaterial Concrete01 5 -$fc -$eco 0 -$epss ; # Material para fibras de concreto
sin confinamiento
uniaxialMaterial Steel01 6 $fy $Es $shr ; # Material para acero de refuerzo longitudinal
uniaxialMaterial Concrete01 7 -$fpcc -$ecc -$fcu -$ecu ; # Material para fibras de
concreto confinado
uniaxialMaterial Elastic 8 $E

# Definición de secciones para modelar la respuesta a flexión y axial
# para luego asignarle respuesta a cortante
# -----

```

Una vez definidos los materiales se forman las secciones de fibras, para el hormigón no confinado y confinado se crean parches rectangulares y para el acero tanto lateral como el superior e inferior se crean fibras. Creada esta sección, se forma una segunda en la que se toma como base la primera y se le añade la respuesta a cortante del material elástico.

```

# COLUMNAS
section Fiber 1 {

```

```

set y0c [expr $bcol / 2]
set y1c [expr $y0c - $coverc]

# Parametros de discretizacion de la seccion en fibras
set numSubdivCoreZc 16 ; # número de fibras de concreto confinado en la direccion
local Z
set numSubdivCoreYc 16 ; # número de fibras de concreto confinado en la direccion
local Y
set numSubdivCoverc 2 ; # número de fibras en el recubrimiento

# parches para recubrimiento lateral
patch quad 1 $numSubdivCoverc [expr $numSubdivCoreZc+2*$numSubdivCoverc] -
$y0c -$y0c -$y1c -$y0c -$y1c $y0c -$y0c $y0c
patch quad 1 $numSubdivCoverc [expr $numSubdivCoreZc+2*$numSubdivCoverc]
$y1c -$y0c $y0c -$y0c $y0c $y0c $y1c $y0c
# parches para recubrimiento inferior y superior
patch quad 1 $numSubdivCoreYc $numSubdivCoverc -$y1c -$y0c $y1c -$y0c $y1c -
$y1c -$y1c -$y1c
patch quad 1 $numSubdivCoreYc $numSubdivCoverc -$y1c $y1c $y1c $y1c $y1c $y0c
-$y1c $y0c
# parche de fibras para el hormigon confinado
patch quad 3 $numSubdivCoreYc $numSubdivCoreZc -$y1c -$y1c $y1c -$y1c $y1c
$y1c -$y1c $y1c
# Acero superior e inferior
fiber 0 $y1c [expr ($nbltbc/2) * $areaBarcol] 2
fiber 0 -$y1c [expr ($nbltbc/2) * $areaBarcol] 2
# Acero lateral
fiber $y1c 0 [expr ($nbllc/2) * $areaBarcol] 2
fiber -$y1c 0 [expr ($nbllc/2) * $areaBarcol] 2
}
# Se crea otra sección donde se añade la respuesta a cortante del material elastico 4 a la
seccion creada 1
section Aggregator 2 4 Vy -section 1

# VIGAS
section Fiber 3 {
set y0 [expr $bviga/2]
set y1 [expr $y0-$cover]
set z0 [expr $hviga/2]
set z1 [expr $z0-$cover]

set numSubdivCoreZ 16 ; # número de fibras de concreto confinado en la direccion
local Z
set numSubdivCoreY 16 ; # número de fibras de concreto confinado en la direccion
local Y
set numSubdivCover 2 ; # número de fibras en el recubrimiento

# parches para recubrimiento lateral

```

```

patch quad 5 $numSubdivCover [expr $numSubdivCoreZ+2*$numSubdivCover] -$y0 -
$z0 -$y1 -$z0 -$y1 $z0 -$y0 $z0
patch quad 5 $numSubdivCover [expr $numSubdivCoreZ+2*$numSubdivCover] $y1 -
$z0 $y0 -$z0 $y0 $z0 $y1 $z0
# parches para recubrimiento inferior y superior
patch quad 5 $numSubdivCoreY $numSubdivCover -$y1 -$z0 $y1 -$z0 $y1 -$z1 -$y1 -
$z1
patch quad 5 $numSubdivCoreY $numSubdivCover -$y1 $z1 $y1 $z1 $y1 $z0 -$y1 $z0
# parche de fibras para el hormigon confinado
patch quad 7 $numSubdivCoreY $numSubdivCoreZ -$y1 -$z1 $y1 -$z1 $y1 $z1 -$y1
$z1
# Acero superior e inferior
fiber 0 $z1 [expr ($nbltb/2) * $areaBar] 6
fiber 0 -$z1 [expr ($nbltb/2) * $areaBar] 6
# Acero lateral
fiber $y1 0 [expr ($nbll/2) * $areaBar] 2
fiber -$y1 0 [expr ($nbll/2) * $areaBar] 2
}

# Se añade la respuesta a cortante del material elastico 8 a la seccion creada 3
section Aggregator 4 8 Vy -section 3

```

Para orientar correctamente los ejes locales de las vigas y columnas se crean 2 transformaciones de coordenadas. La primera es para las vigas en el plano XZ y para las vigas en el plano YZ, este vector es el mismo para ambos planos ya que el eje local z de las dos tiene el mismo sentido y dirección (hacia arriba). La segunda transformación de coordenadas es para las columnas para las cuales se orienta el eje local X en la dirección global Z. Para un mejor entendimiento se muestran unas figuras a continuación para un modelo en 2D:

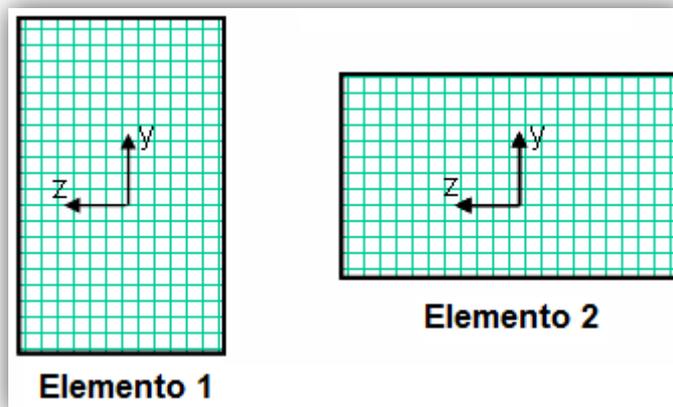


FIGURA 38 – Sección transversal del elemento

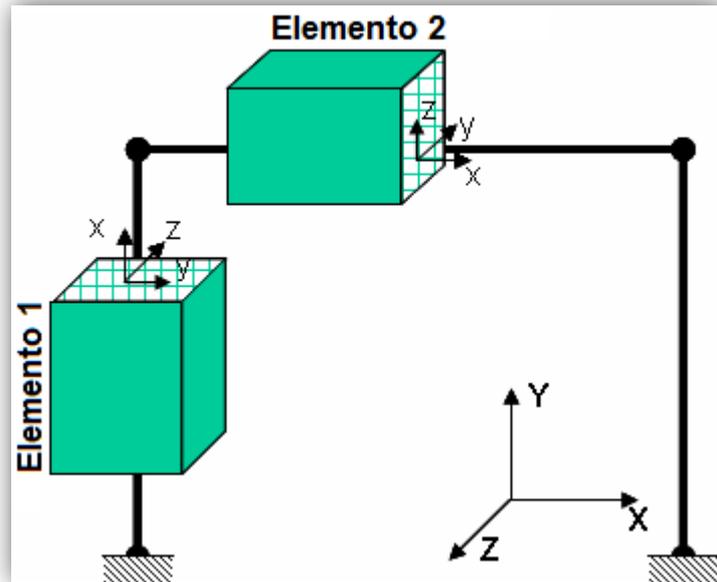


FIGURA 39 – Orientación de los elementos

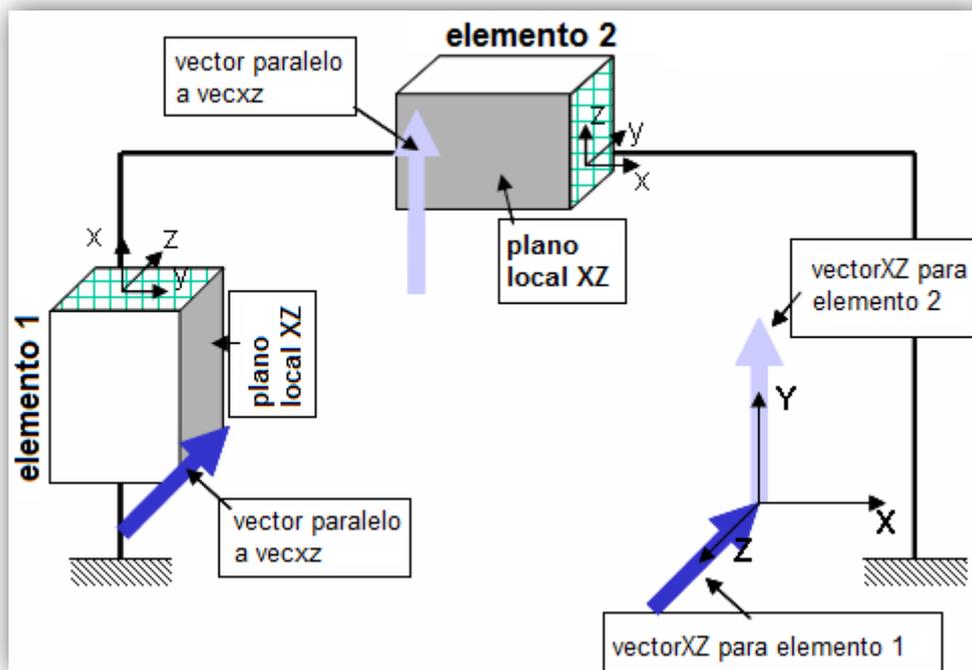


FIGURA 40 – Vector paralelo al plano XZ

Por lo tanto para el modelo en 2D de las gráficas sería:

- Para el elemento 1: vectorXZ = eje Z, coordenadas (0 , 0 , -1)
- Para el elemento 2: vectorXZ = eje Y, coordenadas (0 , 1 , 0)

De la misma forma que para el modelo en 2D se hace para el modelo en 3D, y se tiene:

Se define una transformacion de coordenadas lineal para las vigas

```

# Se usara el vector (0 0 1) para orientar el eje local Z de las vigas en la dirección global
Z
#para vigas en el plano XZ y en el plano YZ
geomTransf Linear 1 0 0 1
# Se define una transformacion de coordenadas lineal para las columnas
# el vector (1 0 0) orienta el eje local X de las columnas en la direccion global Z
geomTransf Linear 2 1 0 0

```

Como se mencionó al inicio de este capítulo lo que se pretende es ejecutar un análisis no lineal, por tanto los elementos que se creen deben cumplir con esta característica. Para ello usaremos el comando **beamWithHinges**, el cual crea elementos no lineales viga – columna. La característica de este comando es que permite considerar la plasticidad del elemento concentrada solamente en los extremos, mientras que la parte interior del elemento tiene un comportamiento elástico. Ésta particularidad es muy útil cuando se quiere elementos con rótulas plásticas.

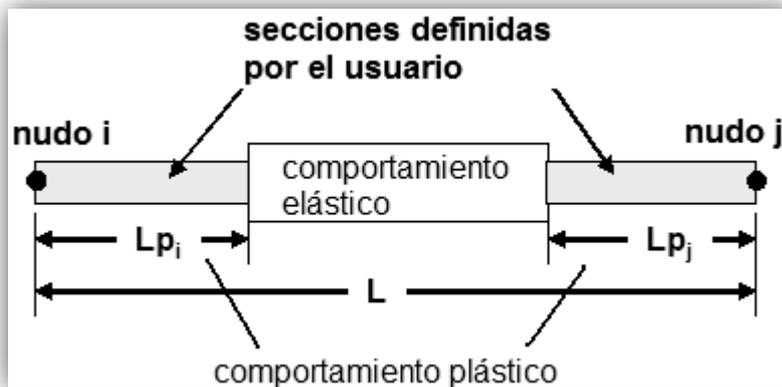


FIGURA 41 – elemento beamWithHinges

Este tipo de comando divide el elemento en tres partes: dos articulaciones en los extremos con comportamiento plástico, y la región media con comportamiento elástico. Las regiones extremas se definen mediante la asignación de secciones previamente definidas por el usuario, al igual que la longitud de que tendrán estas zonas plásticas.

Este comando se basa en un proceso iterativo o no por medio de la integración de Gauss-Radau. Los puntos de integración para este proceso están localizados en las regiones plásticas. Dos puntos de integración por articulación se usan para poder representar la distribución de la curvatura con precisión.

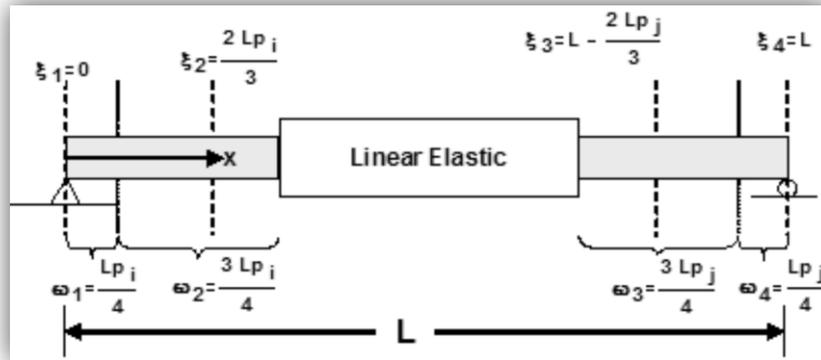


FIGURA 42 - Puntos de integración

Hay muchas ventajas de la formación de este tipo de elementos en comparación con las demás:

- El comportamiento no lineal se limita únicamente a los puntos de integración en los extremos de los elementos.
- El usuario sólo tiene que especificar la longitud de cada región plástica.
- Captura el mayor momento de flexión en los extremos.
- Representa la distribución lineal de la curvatura exactamente.

Los argumentos para la construcción del elemento depende de los grados de libertad que tenga el modelo, así:

Para un modelo en dos dimensiones:

```
element beamWithHinges $eleTag $iNode $jNode $secTagI $Lpi $secTagJ
$Lpj $E $A $Iz $transfTag <-mass $massDens> <-iter $maxIters $tol>
```

Para un modelo en tres dimensiones:

```
element beamWithHinges $eleTag $iNode $jNode $secTagI $Lpi $secTagJ
$Lpj $E $A $Iz $Iy $G $J $transfTag <-mass $massDens> <-iter $maxIters
$tol>
```

Donde:

- \$ EleTag** etiqueta única
- \$iNode \$jNode** nudo de inicio y fin del elemento
- \$secTagI** identificador de la sección previamente definida que forma la región plástica en el nudo i
- \$Lpi** longitud de las region plástica en el nudo i
- \$secTagJ** identificador de la sección previamente definida que forma la región plástica en el nudo j
- \$Lpj** longitud de las region plástica en el nudo j
- \$E** módulo de young
- \$A** área de la sección transversal del elemento
- \$Iz** momento de inercia de la sección alrededor del eje local z
- \$Iy** momento de inercia de la sección alrededor del eje local y

\$G	momento cortante
\$J	momento torsional de inercia de la sección transversal
\$transfTag	identificador de la transformación de coordenadas previamente definida
\$massDens	elemento masa densidad (por unidad de longitud) para formar la matriz de masa (opcional, por defecto 0).
\$maxIters	número máximo de iteraciones hasta lograr la convergencia del elemento (opcional, por defecto 1).
\$tol	tolerancia permitida para satisfacer la convergencia (opcional, por defecto 1).

Hay que tener en cuenta que el cambiar el parámetro **-iter** puede mejorar la convergencia global a costa de un mayor tiempo en la ejecución del análisis. También hay que recordar que las propiedades elásticas del elemento sólo se considerarán en la región interior (la elástica) del elemento.

Otro aspecto a considerar es la creación de los recorders de los elementos **ElementRecorder**, ya que si lo que se desea es obtener las respuestas de fuerza, deformación, rigidez, esfuerzo, el parámetro **\$secNum** del recorder se refiere al punto de integración del cual se quieren obtener los resultados, más no la etiqueta de la sección que forma el elemento. Este tipo de elementos tiene cuatro secciones elásticas y dos secciones plásticas, en total 6 secciones, es decir la 1 y la 6 corresponden a las secciones de inicio y fin respectivamente del elemento.

\$secNum = 1 : región plástica en el nudo i

\$secNum = 6 : región plástica en el nudo j

```
# Se define los elementos
# -----
#COLUMNAS
# 1er piso
set hingec [expr $bcol/2]
element beamWithHinges 1 1 10 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G $Jcol
2
element beamWithHinges 2 2 11 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G $Jcol
2
element beamWithHinges 3 3 12 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G $Jcol
2
element beamWithHinges 4 4 13 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G $Jcol
2
element beamWithHinges 5 5 14 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G $Jcol
2
element beamWithHinges 6 6 15 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G $Jcol
2
```

```

element beamWithHinges 7 7 16 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G $Jcol
2
element beamWithHinges 8 8 17 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G $Jcol
2
element beamWithHinges 9 9 18 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G $Jcol
2
# 2do piso
element beamWithHinges 10 10 19 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G
$Jcol 2
element beamWithHinges 11 11 20 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G
$Jcol 2
element beamWithHinges 12 12 21 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G
$Jcol 2
element beamWithHinges 13 13 22 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G
$Jcol 2
element beamWithHinges 14 14 23 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G
$Jcol 2
element beamWithHinges 15 15 24 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G
$Jcol 2
element beamWithHinges 16 16 25 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G
$Jcol 2
element beamWithHinges 17 17 26 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G
$Jcol 2
element beamWithHinges 18 18 27 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G
$Jcol 2
# 3er piso
element beamWithHinges 19 19 28 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G
$Jcol 2
element beamWithHinges 20 20 29 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G
$Jcol 2
element beamWithHinges 21 21 30 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G
$Jcol 2
element beamWithHinges 22 22 31 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G
$Jcol 2
element beamWithHinges 23 23 32 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G
$Jcol 2
element beamWithHinges 24 24 33 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G
$Jcol 2
element beamWithHinges 25 25 34 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G
$Jcol 2
element beamWithHinges 26 26 35 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G
$Jcol 2
element beamWithHinges 27 27 36 2 $hingec 2 $hingec $E $Acol $Izcol $Iycol $G
$Jcol 2

# VIGAS
# 1er piso
set hingeV [expr $bviga]

```

element beamWithHinges 28 10 11 4 \$hingeV 4 \$hingeV \$E \$Aviga \$Izviga \$Iyviga \$G \$Jviga 1
element beamWithHinges 29 11 12 4 \$hingeV 4 \$hingeV \$E \$Aviga \$Izviga \$Iyviga \$G \$Jviga 1
element beamWithHinges 30 13 14 4 \$hingeV 4 \$hingeV \$E \$Aviga \$Izviga \$Iyviga \$G \$Jviga 1
element beamWithHinges 31 14 15 4 \$hingeV 4 \$hingeV \$E \$Aviga \$Izviga \$Iyviga \$G \$Jviga 1
element beamWithHinges 32 16 17 4 \$hingeV 4 \$hingeV \$E \$Aviga \$Izviga \$Iyviga \$G \$Jviga 1
element beamWithHinges 33 17 18 4 \$hingeV 4 \$hingeV \$E \$Aviga \$Izviga \$Iyviga \$G \$Jviga 1
element beamWithHinges 34 10 13 4 \$hingeV 4 \$hingeV \$E \$Aviga \$Izviga \$Iyviga \$G \$Jviga 1
element beamWithHinges 35 13 16 4 \$hingeV 4 \$hingeV \$E \$Aviga \$Izviga \$Iyviga \$G \$Jviga 1
element beamWithHinges 36 11 14 4 \$hingeV 4 \$hingeV \$E \$Aviga \$Izviga \$Iyviga \$G \$Jviga 1
element beamWithHinges 37 14 17 4 \$hingeV 4 \$hingeV \$E \$Aviga \$Izviga \$Iyviga \$G \$Jviga 1
element beamWithHinges 38 12 15 4 \$hingeV 4 \$hingeV \$E \$Aviga \$Izviga \$Iyviga \$G \$Jviga 1
element beamWithHinges 39 15 18 4 \$hingeV 4 \$hingeV \$E \$Aviga \$Izviga \$Iyviga \$G \$Jviga 1
2do piso
element beamWithHinges 40 19 20 4 \$hingeV 4 \$hingeV \$E \$Aviga \$Izviga \$Iyviga \$G \$Jviga 1
element beamWithHinges 41 20 21 4 \$hingeV 4 \$hingeV \$E \$Aviga \$Izviga \$Iyviga \$G \$Jviga 1
element beamWithHinges 42 22 23 4 \$hingeV 4 \$hingeV \$E \$Aviga \$Izviga \$Iyviga \$G \$Jviga 1
element beamWithHinges 43 23 24 4 \$hingeV 4 \$hingeV \$E \$Aviga \$Izviga \$Iyviga \$G \$Jviga 1
element beamWithHinges 44 25 26 4 \$hingeV 4 \$hingeV \$E \$Aviga \$Izviga \$Iyviga \$G \$Jviga 1
element beamWithHinges 45 26 27 4 \$hingeV 4 \$hingeV \$E \$Aviga \$Izviga \$Iyviga \$G \$Jviga 1
element beamWithHinges 46 19 22 4 \$hingeV 4 \$hingeV \$E \$Aviga \$Izviga \$Iyviga \$G \$Jviga 1
element beamWithHinges 47 22 25 4 \$hingeV 4 \$hingeV \$E \$Aviga \$Izviga \$Iyviga \$G \$Jviga 1
element beamWithHinges 48 20 23 4 \$hingeV 4 \$hingeV \$E \$Aviga \$Izviga \$Iyviga \$G \$Jviga 1
element beamWithHinges 49 23 26 4 \$hingeV 4 \$hingeV \$E \$Aviga \$Izviga \$Iyviga \$G \$Jviga 1
element beamWithHinges 50 21 24 4 \$hingeV 4 \$hingeV \$E \$Aviga \$Izviga \$Iyviga \$G \$Jviga 1

```

element beamWithHinges 51 24 27 4 $hingeV 4 $hingeV $E $Aviga $Izviga $Iyviga $G
$Jviga 1
# 3er piso
element beamWithHinges 52 28 29 4 $hingeV 4 $hingeV $E $Aviga $Izviga $Iyviga $G
$Jviga 1
element beamWithHinges 53 29 30 4 $hingeV 4 $hingeV $E $Aviga $Izviga $Iyviga $G
$Jviga 1
element beamWithHinges 54 31 32 4 $hingeV 4 $hingeV $E $Aviga $Izviga $Iyviga $G
$Jviga 1
element beamWithHinges 55 32 33 4 $hingeV 4 $hingeV $E $Aviga $Izviga $Iyviga $G
$Jviga 1
element beamWithHinges 56 34 35 4 $hingeV 4 $hingeV $E $Aviga $Izviga $Iyviga $G
$Jviga 1
element beamWithHinges 57 35 36 4 $hingeV 4 $hingeV $E $Aviga $Izviga $Iyviga $G
$Jviga 1
element beamWithHinges 58 28 31 4 $hingeV 4 $hingeV $E $Aviga $Izviga $Iyviga $G
$Jviga 1
element beamWithHinges 59 31 34 4 $hingeV 4 $hingeV $E $Aviga $Izviga $Iyviga $G
$Jviga 1
element beamWithHinges 60 29 32 4 $hingeV 4 $hingeV $E $Aviga $Izviga $Iyviga $G
$Jviga 1
element beamWithHinges 61 32 35 4 $hingeV 4 $hingeV $E $Aviga $Izviga $Iyviga $G
$Jviga 1
element beamWithHinges 62 30 33 4 $hingeV 4 $hingeV $E $Aviga $Izviga $Iyviga $G
$Jviga 1
element beamWithHinges 63 33 36 4 $hingeV 4 $hingeV $E $Aviga $Izviga $Iyviga $G
$Jviga 1

```

Se cargan las vigas y columnas en forma consistente con la orientación de ejes locales dada con el comando `geomTransf`, para las vigas “-Z” y para las columnas “-X”, el signo negativo para simular las cargas gravitacionales, se ejecuta el análisis. Se crean los recorders, el primero para grabar las respuestas de desplazamiento en el grado de libertad 1 (eje X global) del nudo maestro del último piso. Las cuatro a continuación guardan la información de fuerza y deformación del elemento 10 en el inicio “sección 1” y en el final “sección 6” de los elementos, tal como se explico en la creación de los elementos **beamWithHinges**.

Los siguientes cuatro recorders guardan la información igualmente de las articulaciones pero esta vez la respuesta esfuerzo – deformación. Los recorders restantes almacenan la misma información antes mencionada con la única diferencia que corresponden a una viga (elemento 40) y no a una columna.

Podrían crearse más recorders para registrar fuerzas y otros tipos de respuesta en la estructura. Sin embargo lo que buscamos con este ejemplo es analizar el comportamiento de las rótulas plásticas.

El comando **constraints** le dice a OpenSees como manejar las restricciones entre grados de libertad dentro del modelo, en este caso hemos cambiado la opción **Plain** que veníamos usando en los modelos anteriores ya esta opción es válida únicamente cuando las restricciones en los grados de libertad de un nudo no implican una relación entre un grado de libertad y otro, por ejemplo en un modelo en 2D las únicas restricciones que hay son en los apoyos de la estructura, es decir en la base de las columnas y esas restricciones no implican relaciones entre un grado de libertad de un nudo con un grado de libertad de otro nudo. No es el caso cuando se modela un edificio donde se crean diafragmas de piso y se quiere que OpenSee vincule los grados de libertad de los nudos del piso para formar un diafragma y simular un piso rígido, como es este caso, para ello usaremos la opción **Transformation**, esta opción transforma la matriz de rigidez del sistema condensando las limitaciones de los grados de libertad. Este metodo reduce el tamaño del sistema de restricciones de múltiples puntos. Este es el método recomendado para un análisis transitorio.

El comando **system** le dice a OpenSees que algoritmo usar para la solución de las ecuaciones lineales, en este caso se algoritmo es **BandGeneral**. Cuando se hace un análisis no lineal hay que decirle a OpenSees de que manera chequear la convergencia del modelo esto se lo hace con el comando **test** donde se indica la tolerancia aceptable para la convergencia y el número de iteraciones máximas para llegar a ella. El comando **algorithm** indica como resolver y llegar a la convergencia cuando el análisis avanza paso a paso en el tiempo en el caso no lineal. El comando **integrator** nos dice como se aplicara la carga, en este caso como sse va a ejecutar un análisis debido a las cargas gravitacionales se hará en un solo paso. **Analysis** define si el análisis va ser estático o va ser en el tiempo, por último el comando **analyze** es el que hace que se ejecute el análisis.

```

#Ejecución del análisis
# Análisis gravitacional
# ++++++
# se define el patrón de carga
# -----
pattern Plain 1 Linear {
# COLUMNAS
eleLoad -ele 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 -
type -beamUniform 0 0 -[expr $bcol*$bcol*24]
# VIGAS
eleLoad -ele 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51
52 53 54 55 56 57 58 59 60 61 62 63 -type -beamUniform 0 -$loadviga 0
}

# Registro de resultados
# -----

```

```

recorder Node -file disp39.txt -time -node 39 -dof 1 disp
recorder Element -file ele10sec1Force.txt -time -ele 10 section 1 force
recorder Element -file ele10sec6Force.txt -time -ele 10 section 6 force
recorder Element -file ele10sec1def.txt -time -ele 10 section 1 deformation
recorder Element -file ele10sec6def.txt -time -ele 10 section 6 deformation
recorder Element -file ele10sec1superior.txt -time -ele 10 section 1 fiber 0 $y1c
stressStrain
recorder Element -file ele10sec6superior.txt -time -ele 10 section 6 fiber 0 $y1c
stressStrain
recorder Element -file ele10sec1inferior.txt -time -ele 10 section 1 fiber 0 -$y1c
stressStrain
recorder Element -file ele10sec6inferior.txt -time -ele 10 section 6 fiber 0 -$y1c
stressStrain
recorder Element -file ele40sec1Force.txt -time -ele 40 section 1 force
recorder Element -file ele40sec6Force.txt -time -ele 40 section 6 force
recorder Element -file ele40sec1def.txt -time -ele 40 section 1 deformation
recorder Element -file ele40sec6def.txt -time -ele 40 section 6 deformation
recorder Element -file ele40sec1superior.txt -time -ele 40 section 1 fiber 0 $y1c
stressStrain
recorder Element -file ele40sec6superior.txt -time -ele 40 section 6 fiber 0 $y1c
stressStrain
recorder Element -file ele40sec1inferior.txt -time -ele 40 section 1 fiber 0 -$y1c
stressStrain
recorder Element -file ele40sec6inferior.txt -time -ele 40 section 6 fiber 0 -$y1c
stressStrain

constraints Transformation
numberer Plain
system BandGeneral
test EnergyIncr 1.e-8 6 0
algorithm Newton
integrator LoadControl 1
analysis Static
analyze 1
puts "ANALISIS GRAVITACIONAL COMPLETADO CON EXITO"

```

Antes de ejecutar el siguiente análisis se congela el análisis gravitacional con el comando “loadConst”. Se define el análisis pushover, en este caso la carga lateral aplicada será triangular. Al igual que para el análisis gravitacional se definen las opciones para la ejecución del pushover. Como se trata de un pushover controlado por el desplazamiento es necesario definir el desplazamiento máximo esperado y el incremento de desplazamiento para cada paso.

```

# Se reinicia para el siguiente caso de análisis
# -----

```

```
loadConst -time 0 ; # se usa para evitar que las cargas gravitacionales sigan incrementando
wipeAnalysis ; # este comando se usa para empezar un nuevo tipo de analisis
```

```
# Análisis: PUSHOVER
# ++++++
# Se define el patrón de carga
# -----
pattern Plain 2 Linear {

load 37 0.16666666667 0 0 0 0
load 38 0.33333333333 0 0 0 0
load 39 0.5 0 0 0 0
}

# Se define las opciones de análisis
# -----

set IDctrlNode 39 ; # nudo en que se controlara el desplazamiento maximo
set IDctrlDOF 1 ; # dirección en la que se aplicara la carga
set Dmax 0.2 ; # desplazamiento maximo esperado
set Dincr 0.002 ; # incremento de desplazamiento
constraints Transformation
numberer Plain
system BandGeneral
test EnergyIncr 1.e-4 500 0
algorithm Newton
integrator DisplacementControl $IDctrlNode $IDctrlDOF $Dincr
analysis Static

# EJECUCION DEL ANALISIS PUSHOVER ESTÁTICO
set Nsteps [expr int($Dmax/$Dincr)] ; # Se define el numero de pasos necesario para
alcanzar el desplazamiento maximo
set ok [analyze $Nsteps]
puts "ANALISIS PUSHOVER COMPLETADO CON EXITO"
```

Es código a continuación se usa en caso de tener problemas de convergencia en la solución.

```
# En caso de problemas de convergencia
if {$ok != 0} {
# Se cambia algunos parámetros del análisis para lograr la convergencia
# performance is slower inside this loop
set ok 0;
set controlDisp 0.0;    # se empieza desde cero
```

```

set D0 0.0; # se empieza desde cero
set Dstep [expr ($controlDisp-$D0)/($Dmax-$D0)]
while {$Dstep < 1.0 && $ok == 0} {
set controlDisp [nodeDisp $IDctrlNode $IDctrlDOF ]
set Dstep [expr ($controlDisp-$D0)/($Dmax-$D0)]
set ok [analyze 1 ]
if {$ok != 0} {
test NormDispIncr 1.e-4 2000 0
algorithm Newton - initial
set ok [analyze 1 ]
test EnergyIncr 1.e-4 $maxNumIter 0
algorithm Newton
}
if {$ok != 0} {
algorithm Broyden 8
set ok [analyze 1 ]
algorithm Newton
}
if {$ok != 0} {
algorithm NewtonLineSearch .8
set ok [analyze 1 ]
algorithm Newton
}
}
}; # end if

```

El siguiente paso es la ejecución de la simulación y por último el análisis de resultados. El archivo `dispx39.txt` contiene 2 columnas la primera es la fuerza (necesaria para alcanzar el desplazamiento meta = 0.2), y la segunda el desplazamiento en sentido X. Este orden viene dado por la forma en la que se creó el recorder. Se puede procesar con excel y obtener la gráfica del pushover.

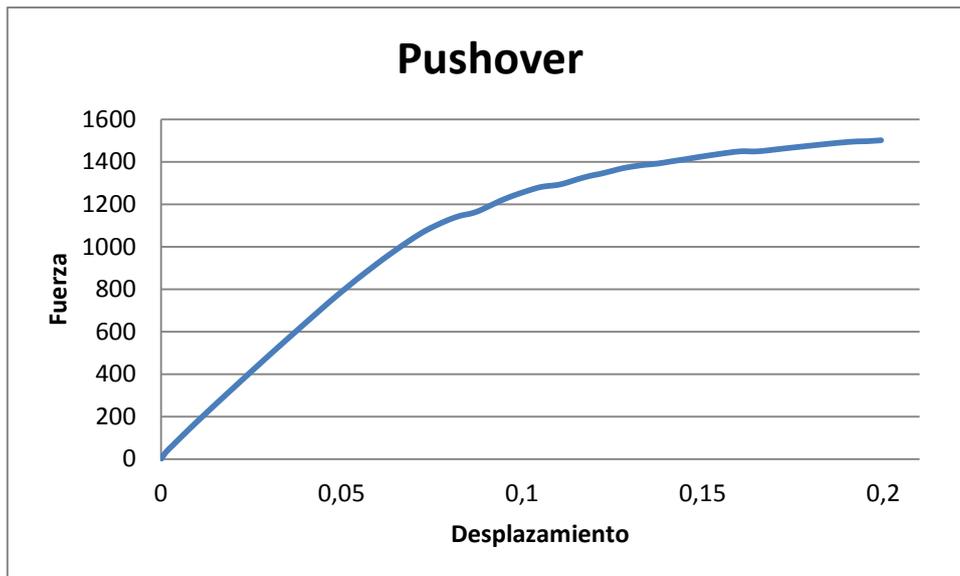


FIGURA 43 – Gráfica Pushover

También si se procesan el resto de archivos pueden obtenerse gráficas de Momento, fuerza axial, cortante, curvatura, esfuerzo tanto en la fibra superior como en la inferior de los dos extremos del elemento. A continuación se presentan las obtenidas para el elemento 40 (viga).

Para la obtención de las cuatro graficas siguientes fue necesario procesar los archivos:

- ele40sec1superior.txt
- ele40sec1inferior.txt
- ele40sec6superior.txt
- ele40sec6inferior.txt

Estos contienen la información de esfuerzos de las fibras superior e inferior en cada extremo del elemento (extremo inicial “sección 1” y extremo final “sección 6”), tal como lo indica el nombre mismo del archivo. Cada archivo de datos contiene tres columnas, la primera al igual que para el archivo “dispx39.txt” corresponde a la fuerza necesaria para alcanzar el desplazamiento meta, esta columna es común para todos los archivos. Y la tercera columna es el esfuerzo en la fibra.

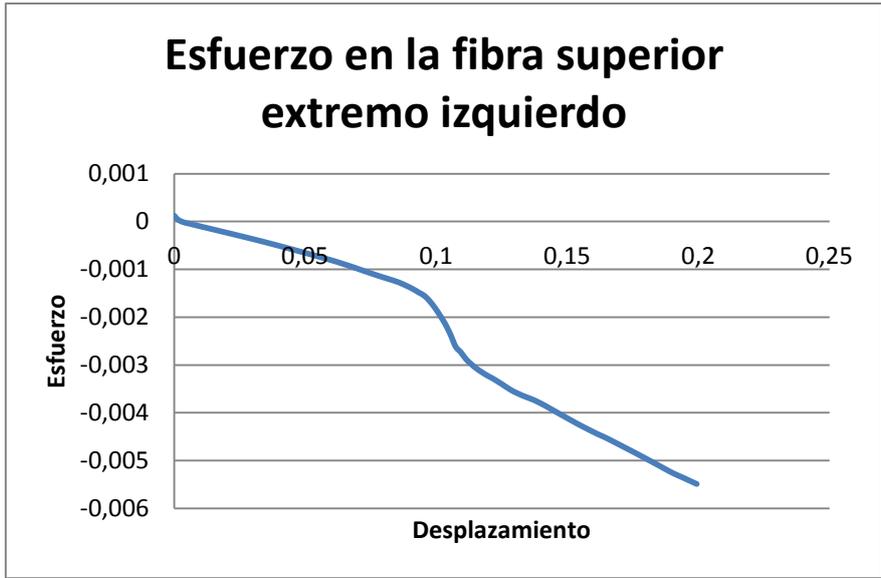


FIGURA 44 – Esfuerzo en fibra superior izquierda

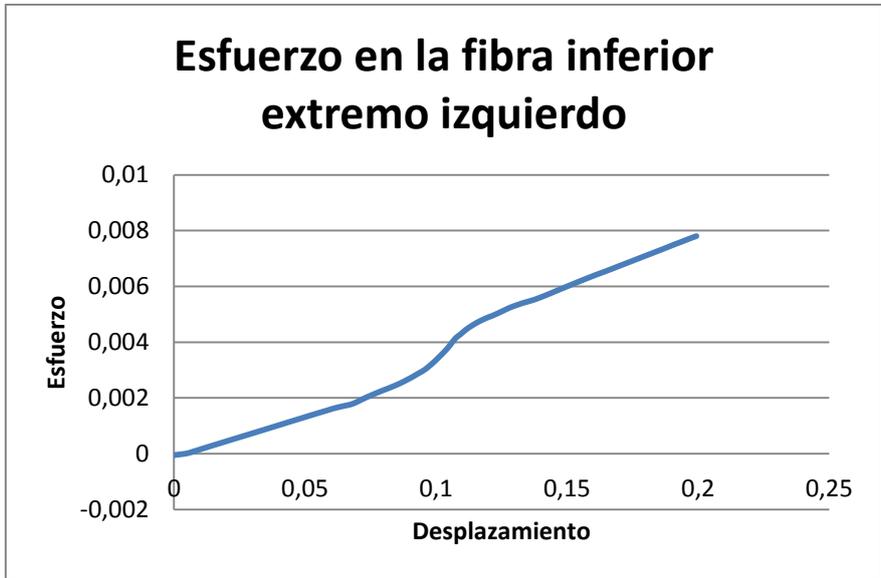


FIGURA 45 – Esfuerzo en fibra inferior izquierda



FIGURA 46 – Esfuerzo en fibra superior derecha

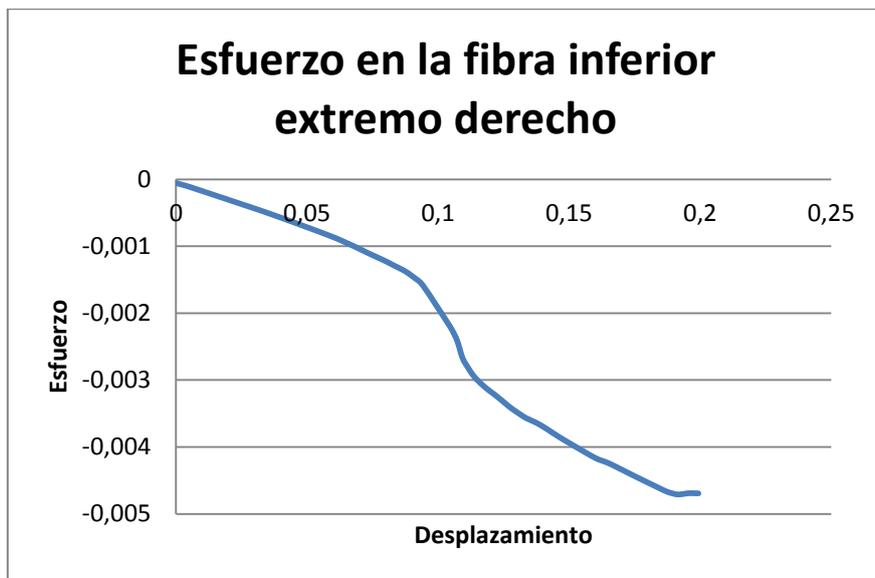


FIGURA 47 – Esfuerzo en fibra inferior derecha

Los archivos:

- ele40sec1Force.txt
- ele40sec1def.txt
- ele40sec6Force.txt
- ele40sec6def.txt

Corresponden a los resultados de fuerza y deformación en los extremos del elemento. Este orden viene dado por la forma en la que se creó el recorder. Los archivos de fuerza contienen la respuesta de fuerza y momento, de aquí se pueden obtener gráficas de fuerza axial y momento. Así mismo del archivo de deformaciones se pueden obtener los

diagramas de curvatura. A continuación se muestra las gráficas de momento y de curvatura en los extremos.

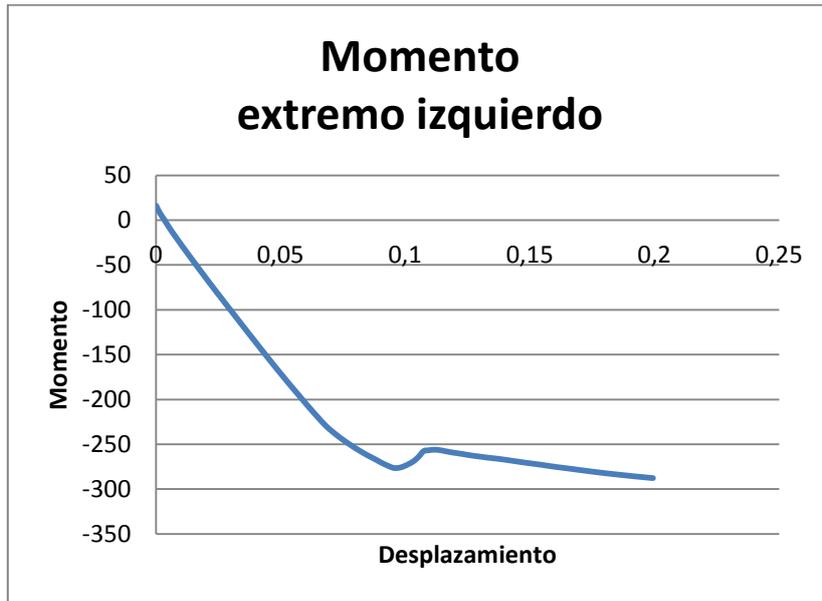


FIGURA 48 – Momento en el extremo izquierdo

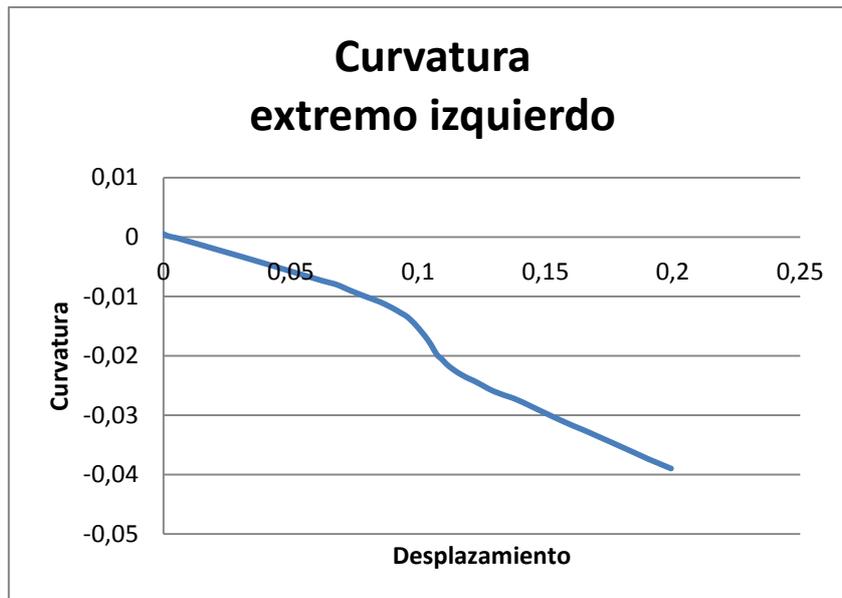


FIGURA 49 – Curvatura en el extremo izquierdo

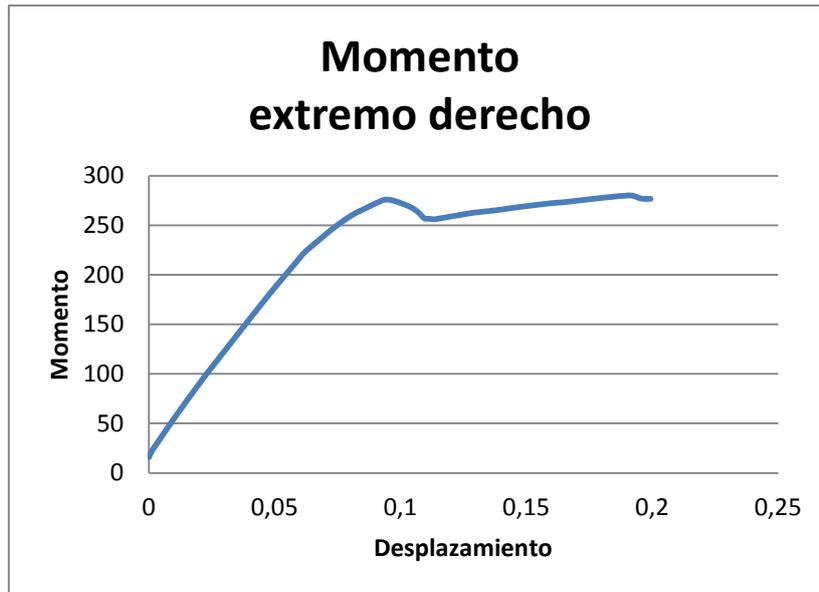


FIGURA 50 – Momento en el extremo derecho



FIGURA 51 – Curvatura en el extremo derecho

Los demás recorders generados corresponden a la respuesta de una columna del primer piso (elemento 10), los cuales al igual que para la viga se pueden procesar con Excel y obtener gráficas similares a las realizadas para la viga.

Como se mencionó al inicio de este trabajo OpenSees permite resolver un mismo problema de maneras distintas, es así que el ejemplo presentado puede modelarse también usando elementos elásticos viga – columna conectados por elementos de longitud cero los cuales se usarán para representar el comportamiento no lineal de la estructura.

6.3.PÓRTICOS EN PUENTES

El riesgo frente a eventos naturales de la infraestructura del transporte en el Ecuador es variable y depende de la amenaza, vulnerabilidad y exposición. Los terremotos son una amenaza real para los puentes y otras estructuras de la red de transporte. Esta amenaza está presente en la mayor parte del territorio nacional y varía según el sitio de implantación de la estructura, siendo las provincias del centro-norte del país las de mayor exposición. Por ello es necesario que los Ingenieros Civiles tengan una formación sólida y actualizada en el diseño de puentes de manera que puedan producir diseños eficientes y seguros.

En los años recientes, el análisis no lineal de puentes ha ganado un gran espacio, por la necesidad de un comportamiento inelástico bajo cargas de sismo. La filosofía común para el diseño sísmico ordinario de un puente es que este sufra un grado de daño sin colapsarse.

Es requerido un control y evaluación del daño con un análisis no lineal post-elástico. Un análisis no lineal es complejo y envuelve muchas simplificaciones que se asumen. Los ingenieros deben estar familiarizados con estas complejidades y asunciones para un diseño de un puente tal que el puente sea seguro y económico.

Muchos factores contribuyen al comportamiento no lineal de un puente. Estos factores incluyen consideraciones como la inelasticidad del material, geometrías o efectos de segundo orden, la interacción no lineal del suelo - fundación - estructura, juntas abriendo y cerrando a las rotulas, en la localización de estribos, efectos diferidos debido al acortamiento y fisuración del concreto, etc. El tema de análisis no lineal es extremadamente extenso, es por ello que en el ejemplo que se presentará más abajo solamente se tomarán en cuenta algunas formulaciones básicas de un análisis estático no lineal para el caso de sismo.

6.3.1. Definición y elementos de un puente

Un puente es una construcción, por lo general artificial, que permite salvar un accidente geográfico o cualquier otro obstáculo físico como un río, un cañón, un valle, un camino, una vía férrea, un cuerpo de agua, o cualquier obstrucción. El diseño de cada puente varía dependiendo de su función y la naturaleza del terreno sobre el que el puente es construido. Su proyecto y su cálculo pertenecen a la ingeniería estructural, siendo numerosos los tipos de diseños que se han aplicado a lo largo de la historia, influidos por los materiales disponibles, las técnicas desarrolladas y las consideraciones económicas, entre otros factores.

Cuando se habla de puentes es común hablar de la superestructura que consiste en un tablero que es quien soporta directamente las cargas, y de la subestructura que la

conforman las pilas, los estribos y los cimientos. Las pilas están formadas por una o más columnas y son las que soportan la superestructura, la misma que puede estar monolítica o simplemente apoyada. De igual forma el otro extremo de las pilas pueden estar simplemente apoyadas en el suelo o embebidas sobre éste con pilotes prebarrenados, dependiendo de las condiciones del suelo. Los estribos son los elementos que soportan el tablero en cada uno de los extremos del puente. Los ciminetos o apoyos de estribos y pilas se encargan de transmitir al terreno todos los esfuerzos que se producen en el puente.

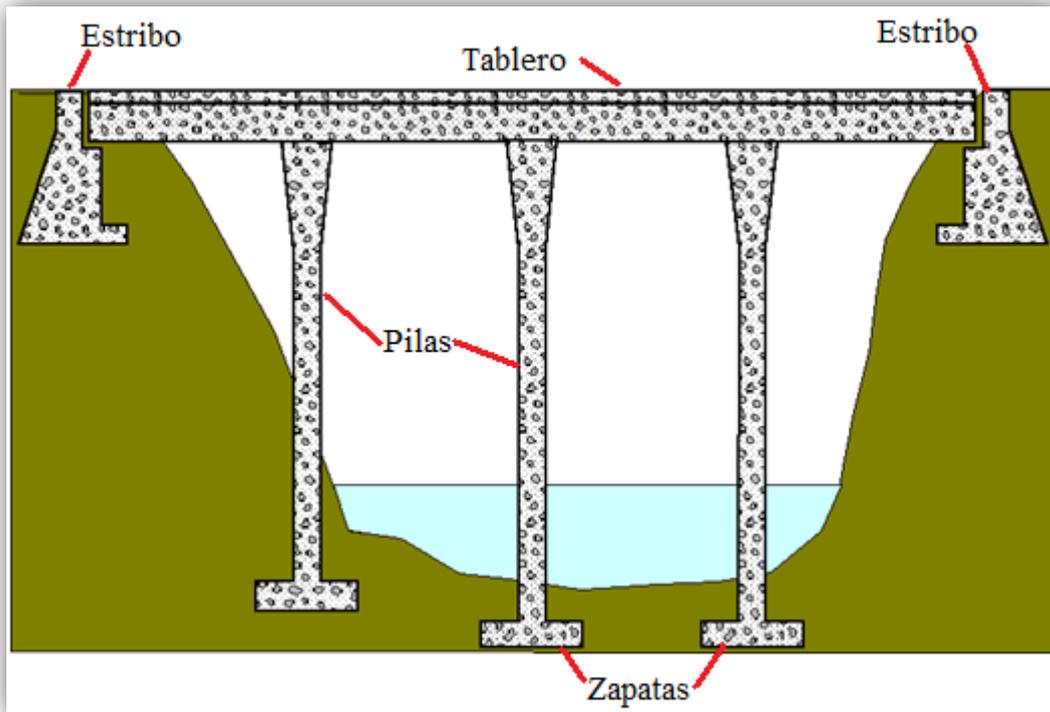


FIGURA 52 – Elementos de un puente

A continuación se presenta un ejemplo en el que se analizará el comportamiento de un puente, este ejemplo puede ser comprobado con ITHA-Bridge, la cual es una herramienta informática que está programada utilizando como base la plataforma OpenSees. Con ITHA-Bridge se puede comprobar si el comportamiento del puente durante la ocurrencia de un sismo es el esperado en la etapa de diseño. Se puede acceder a esta herramienta virtual de servicio público y mundial, a través de la página de internet del VLEE: <http://www.utpl.edu.ec/vlee>.

6.4.EJEMPLO. ANÁLISIS NO LINEAL DE UN PUENTE DE 3 VANOS Y LUZ 90M.

En el presente ejemplo se construirá un puente sobre el cual se ejecutará un análisis no-lineal. En este ejemplo se introducen nuevos conceptos y se hace uso de nuevos comandos que se detallarán en el transcurso del ejemplo.

Al igual que en los ejemplos anteriores es necesario idealizar la estructura. Así, nuestro modelo del puente se diseña con las siguientes características:

- El pórtico 3D del puente se basa en la geometría de la FIGURA 53. El puente consta de 3 vanos de 30m cada uno, dando una longitud total de 90m.
- La super – estructura es continua de concreto. Su peso es de 135kN/m. La altura desde la base de la super – estructura hasta el centro de gravedad es de 1.4m. Una sección transversal de 12m².
- Se ha considerado que las pilas están formadas de una columna integral. Las pilas tienen una altura uniforme de 7m, la unión con la super – estructura es monolítica. El diámetro de las pilas es de 1.6m. El ancho de la viga cabezal es de 12m y su peso es de 500kN/m.
- El concreto con el que se construirán las pilas del puente tiene una resistencia a la compresión $f'c= 30\text{MPa}$, peso unitario de 24kN/m^3 . El esfuerzo de fluencia del acero de refuerzo es igual a 420MPa .
- El modelo tiene 30 nudos. Tal como se ve en la FIGURA 54. Cada vano se dividirá en 5 partes para un mejor análisis. El modelo completo consta de 25 elementos.
- Se modelará las pilas como elementos no lineales. Se crean secciones de fibras para estos elementos.
- El módulo elástico del hormigón se toma como base para definir las propiedades mecánicas de las secciones.
- La masa de la estructura se distribuirá apropiadamente en sus nudos.
- Las propiedades del concreto confinado serán calculadas en tiempo de ejecución utilizando el modelo de Mander.
- Se ejecutará un análisis de carga muerta, luego un análisis sísmico en la dirección “X” y “Y”.

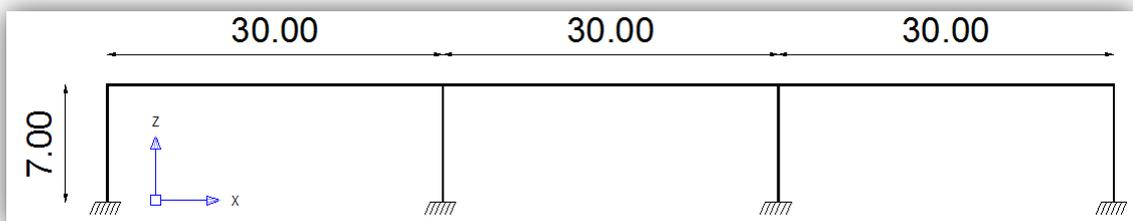


FIGURA 53 – Elevación plano XZ del modelo

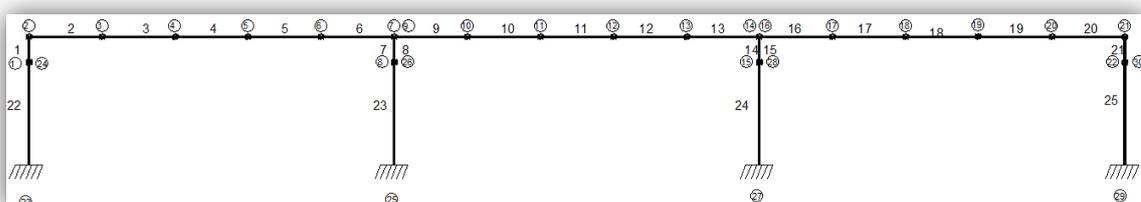


FIGURA 54 – Nudos y elementos

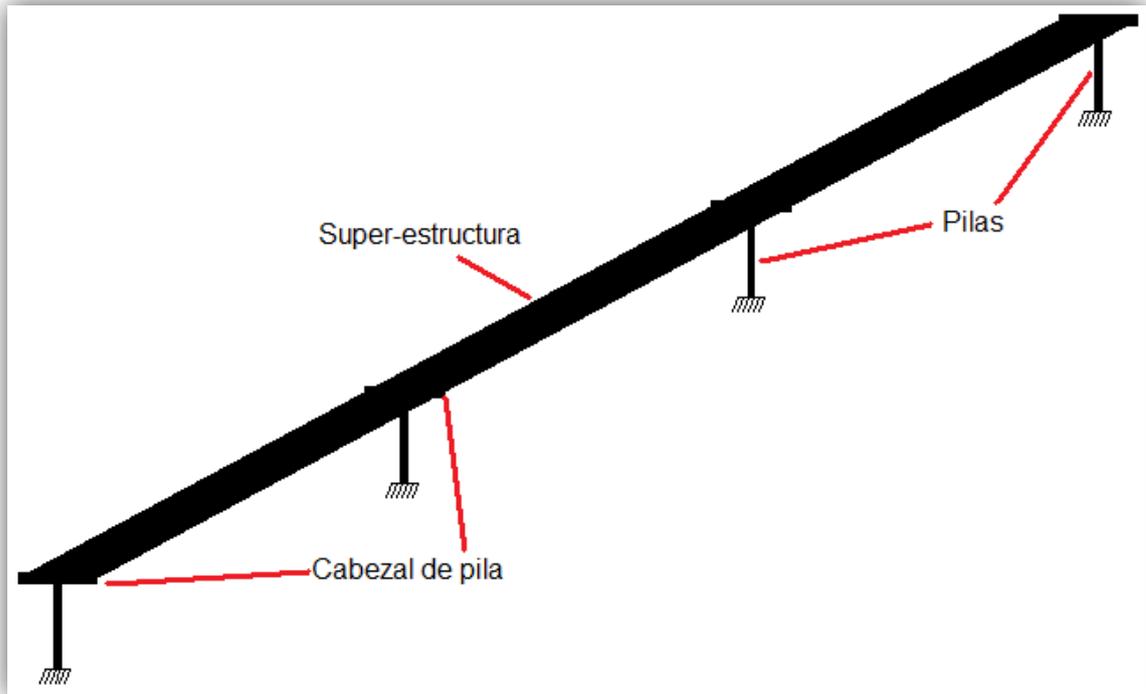


FIGURA 55 – Vista 3D del modelo

Empezamos con la definición del modelo de 3 dimensiones con 6 grados de libertad. Definimos las constantes como variables, e ingresamos los datos necesarios para la generación de nuestro puente, como son el número de vanos, la longitud total, las características de la super – estructura, de la subestructura y las propiedades de los materiales.

```
#####
#Ejemplo8.tcl #
#Análisis no-lineal en 3D de un puente de 3 vanos, con una luz de 90m #
#Unidades: m, s, kN #
#####

# CREACION DEL MODELO EN 3 DIMENSIONES Y 6 GRADOS DE LIBERTAD
wipe
model BasicBuilder -ndm 3 -ndf 6

# Se definen las constantes a usar
set g 9.81
set pi [expr acos(-1)]

# SE INGRESAN LOS DATOS DEL PUENTE:
# -----
# -----
set nspans 3; #Numero de vanos del puente
```

set slength 90; #Longitud total del puente (m)

PROPIEDADES DE LA SUPER-ESTRUCTURA:

set superW 135; #Peso de la super-estructura (kN/m)

set superH 1.4; #Distancia desde el centroide hasta la parte inferior de la super-estructura (m)

set superIz 132; #Inercia con respecto al eje Z (m⁴)

set superIy 23; #Inercia con respecto al eje Y (m⁴)

set superE 27500; #Modulo de elasticidad del concreto (MPa)

set superA 12; #Area de la sección transversal del claro (m²)

set superJ 15; #Constante de torsión (m³)

set superNPS 5; #Número de subdivisiones de cada vano

PROPIEDADES DE LA SUB-ESTRUCTURA:

CARACTERISTICAS COMUNES DE LAS PILAS

set DIA 1.6; # Diametro de la columna (m)

set H 7; # Altura libre de la columna (m)

set NLBAR 20; # Numero de barras longitudinales

set DLBAR 0.042; # Diametro de las barras longitudinales(m)

set DHBAR 22; # Diametro del refuerzo transversal. (mm)

set HBARS 150; # Espaciamiento del refuerzo transversal(mm)

set COV 50; # Recubrimiento del refuerzo transversal (mm)

set CBW 500; # Peso de la viga cabezal(kN)

PROPIEDADES DE LOS MATERIALES

set WC 24; # Peso unitario del concreto. (kN/m³)

set FPC 30; # Resistencia a la compresion del concreto no confinado. (MPa)

set FY 420; # Esfuerzo de fluencia de las barras longitudinales. (MPa)

set FUR 1.4; # Relacion de ultima fluencia para las barras de refuerzo longitudinal.

set ESU 0.06; # Deformación unitaria del acero longitudinal.

set FYH 420; # Esfuerzo de fluencia del refuerzo transversal. (MPa)

set FURH 1.4; # Relacion de ultima fluencia para el refuerzo transversal.

set ESUH 0.06; # Deformación unitaria del acero transversal.

set Es 200000; # Modulo de elasticidad del acero ()

set poisson 0.2; # Relacion de poisson para el concreto

Ingresamos las coordenadas de cada uno de los nudos tal como se mostró en la FIGURA 54. Empotramos las bases de las pilas en el suelo. Con el comando “equalDOF” restringimos los grados de libertad 1, 2, 3, 4, 5, 6 (es decir: DespX, DespY, DespZ, Mx, My, Mz) de los nudos 1, 8, 15, y 22 de la misma manera que para los nudos 24, 26, 28 y 30. Este comando es usado para restringir un mismo grado de libertad a varios puntos. Su sintáxis es:

equalDOF \$rNodeTag \$cNodeTag \$dof1 \$dof2 ...

Donde:

\$rNodeTag	etiqueta o nombre del nudo master (rNode).
\$cNodeTag	etiqueta del nudo a inhibir, nudo esclavo (cNode).
\$dof1 \$dof2 ...	grado o grados de libertad que se restringiran en el nudo cNode de igual forma que se hará con el nudo rNode

El rango válido para ello es desde 1 hasta 6, es decir depende el número de grados de libertad del modelo, por ejemplo si el modelo es en 2 dimensiones el rango será de 1 a 3, si en tres dimensiones será de 1 a 6.

```

# SE DEFINE LA GEOMETRIA
# SE CREA LOS NUDOS QUE CONFORMAN EL PUENTE.
node 1 0 0 0
node 2 0 0 1.4
node 3 6.0 0 1.4
node 4 12.0 0 1.4
node 5 18.0 0 1.4
node 6 24.0 0 1.4
node 7 30.0 0 1.4
node 8 30.0 0 0
node 9 30 0 1.4
node 10 36.0 0 1.4
node 11 42.0 0 1.4
node 12 48.0 0 1.4
node 13 54.0 0 1.4
node 14 60.0 0 1.4
node 15 60.0 0 0
node 16 60 0 1.4
node 17 66.0 0 1.4
node 18 72.0 0 1.4
node 19 78.0 0 1.4
node 20 84.0 0 1.4
node 21 90.0 0 1.4
node 22 90.0 0 0
node 23 0 0 -7
node 24 0 0 0
node 25 30 0 -7
node 26 30 0 0
node 27 60 0 -7
node 28 60 0 0
node 29 90 0 -7
node 30 90 0 0

# SE FIJA LOS NUDOS (DEFINICION DE RESTRICCIONES)
# BASE DE LAS PILAS
fix 23 1 1 1 1 1 1
fix 25 1 1 1 1 1 1

```

```

fix 27 1 1 1 1 1 1
fix 29 1 1 1 1 1 1
# CABEZA DE LAS PILAS
equalDOF 24 1 1 2 3 4 5 6
equalDOF 26 8 1 2 3 4 5 6
equalDOF 28 15 1 2 3 4 5 6
equalDOF 30 22 1 2 3 4 5 6

```

Se definen las masas en cada uno de tal manera que se distribuya todo el peso de la super – estructura en ellos, en este caso se han definido las masas en los grados de libertad uno y dos, por cuanto se va aplicar un análisis dinámico (análisis sísmico) en esos dos sentidos de desplazamiento (X y Y). Sin embargo para evitar problemas de convergencia en la solución del análisis se puede asignar a los grados de libertad libres de cada nudo valores de masa aunque sea muy pequeños.

```

# SE DEFINEN LAS MASAS DE LOS NUDOS
# LA MASA EN CADA NUDO ES IGUAL AL PESO DE LA SUPER-
ESTRUCTURA
# POR LA DISTANCIA ENTRE ESTACIONES Y DIVIDIDA ENTRE DOS.
# ADEMAS SE DIVIDE POR LA GRAVEDAD PARA CONVERTIR EL PESO EN
MASA.
# SE HA CONSIDERA QUE CADA VANO ESTA DIVIDIDO EN 5 PARTES
# COMO CADA VANO TIENE 30M, POR LO TANTO 6m ES LA DISTANCIA
ENTRE
# CADA ESTACION.
set lstation [expr ($slength/$nspans)/$superNPS]; #Longitud entre cada estacion (m)

# Masa de los nudos del puente
mass 2 [expr $superW*$lstation/2/$g] [expr $superW*$lstation/2/$g] 0 0 0 0
mass 3 [expr $superW*$lstation/2/$g] [expr $superW*$lstation/2/$g] 0 0 0 0
mass 4 [expr $superW*$lstation/2/$g] [expr $superW*$lstation/2/$g] 0 0 0 0
mass 5 [expr $superW*$lstation/2/$g] [expr $superW*$lstation/2/$g] 0 0 0 0
mass 6 [expr $superW*$lstation/2/$g] [expr $superW*$lstation/2/$g] 0 0 0 0
mass 7 [expr $superW*$lstation/2/$g] [expr $superW*$lstation/2/$g] 0 0 0 0
mass 9 [expr $superW*$lstation/2/$g] [expr $superW*$lstation/2/$g] 0 0 0 0
mass 10 [expr $superW*$lstation/2/$g] [expr $superW*$lstation/2/$g] 0 0 0 0
mass 11 [expr $superW*$lstation/2/$g] [expr $superW*$lstation/2/$g] 0 0 0 0
mass 12 [expr $superW*$lstation/2/$g] [expr $superW*$lstation/2/$g] 0 0 0 0
mass 13 [expr $superW*$lstation/2/$g] [expr $superW*$lstation/2/$g] 0 0 0 0
mass 14 [expr $superW*$lstation/2/$g] [expr $superW*$lstation/2/$g] 0 0 0 0
mass 16 [expr $superW*$lstation/2/$g] [expr $superW*$lstation/2/$g] 0 0 0 0
mass 17 [expr $superW*$lstation/2/$g] [expr $superW*$lstation/2/$g] 0 0 0 0
mass 18 [expr $superW*$lstation/2/$g] [expr $superW*$lstation/2/$g] 0 0 0 0
mass 19 [expr $superW*$lstation/2/$g] [expr $superW*$lstation/2/$g] 0 0 0 0
mass 20 [expr $superW*$lstation/2/$g] [expr $superW*$lstation/2/$g] 0 0 0 0

```

```

mass 21 [expr $superW*$lstation/2/$g] [expr $superW*$lstation/2/$g] 0 0 0 0
# Masa en la cabeza de las pilas
mass 24 [expr 0.875*$DIA**2*$H/3*$WC/$g +$CBW/$g] [expr
0.875*$DIA**2*$H/3*$WC/$g +$CBW/$g] [expr 0.875*$DIA**2*$H/3*$WC/$g
+$CBW/$g] 0 0 0
mass 26 [expr 0.875*$DIA**2*$H/3*$WC/$g +$CBW/$g] [expr
0.875*$DIA**2*$H/3*$WC/$g +$CBW/$g] [expr 0.875*$DIA**2*$H/3*$WC/$g
+$CBW/$g] 0 0 0
mass 28 [expr 0.875*$DIA**2*$H/3*$WC/$g +$CBW/$g] [expr
0.875*$DIA**2*$H/3*$WC/$g +$CBW/$g] [expr 0.875*$DIA**2*$H/3*$WC/$g
+$CBW/$g] 0 0 0
mass 30 [expr 0.875*$DIA**2*$H/3*$WC/$g +$CBW/$g] [expr
0.875*$DIA**2*$H/3*$WC/$g +$CBW/$g] [expr 0.875*$DIA**2*$H/3*$WC/$g
+$CBW/$g] 0 0 0

```

Las propiedades del hormigón confinado se calculan durante la ejecución del análisis, así como la deformación unitaria del refuerzo trnsversal que causará su ruptura. Una vez definidas las propiedades del hormigón se crean los materiales que se usaran para formar la sección que tendrán las pilas. Se crean el material para el concreto sin confinamiento, el concreto confinado, para el acero de refuerzo longitudinal y dos materiales elásticos de los cuales se obtendrán las respuestas a cortante y torsión para que las pilas se asemejan más a la realidad, y de igual forma obtener resultados más confiables.

```

# Propiedades del hormigon confinado a usar en la sección de acuerdo al modelo de
mander,
set eco 0.002; # Deformacion unitaria a maxima resistencia del concreto
set Ec [expr 4700.0*sqrt($FPC)]
set Dc [expr ($DIA*1000-2*$COV)]
set sp [expr $HBARS-$DHBAR]
set rcc [expr $DLBAR**2*$NLBAR/$Dc**2]
set ke [expr (1-$sp/2/$Dc)/(1-$rcc)]
set rs [expr $pi*$DHBAR**2/$Dc/$HBARS]
set fpl [expr 1/2.0*$ke*$rs*$FY]
# Se calcula la deformacion unitaria que causa la ruptura del refuerzo transversal.
set fpcc [expr (-1.254 + 2.254 * (1 + 7.94 * $fpl / $FPC) ** 0.5 - 2 * $fpl / $FPC) *
$FPC]
set ecc [expr $eco * (1 + 5 * ($fpcc / $FPC - 1))]
set Esec [expr $fpcc / $ecc]
set r [ expr $Ec / ($Ec - $Esec)]
set ecu [ expr 1.5 * (0.004 + 1.4 * $rs * $FYH * $ESU / $fpcc) ]
set x [expr $ecu / $ecc ]
set fcu [expr $fpcc * $x * $r / ($r - 1 + $x ** $r)]

# SE DEFINEN LOS MATERIALES:

```

```

uniaxialMaterial Concrete01 1 [expr -$FPC*1000] -$Seco 0 -0.004; # Material para
fibras de concreto sin confinamiento
uniaxialMaterial Concrete01 2 [expr -$fpcc*1000] -$Secc [expr -$fcu*1000] -$Secu; #
Material para fibras de concreto confinado
set b [expr ($FUR-1)*$FY/($ESU-0.002)/$Es]
uniaxialMaterial Steel01 3 [expr $FY*1000] [expr $Es*1000] $b; # Material para acero
de refuerzo longitudinal
# Para la rigidez de cortante.
# La capacidad de cortante de la columna por CALTRANS.
set G [expr $Ec/2.0/(1.0+$poisson)]
set J [expr $pi*pow($DIA,4)/32]
uniaxialMaterial Elastic 4 [expr $G*1000]
uniaxialMaterial Elastic 5 $J

```

Igual que en el ejemplo anterior se define el número de fibras que sormarán la sección, con la única diferencia que ahora la sección es circular. El procedimiento ha seguir es el mismo. A continuación se define las transformaciones de coordenadas para orientar los ejes locales de los elementos con respecto de los ejes globales de la estructura. Para los elementos de la super – estructura se crean transformaciones lineales y para las pilas una transformación P-Delta.

Los efectos P-Delta corresponden a los efectos adicionales, en las dos direcciones principales de la estructura, causados por efectos de segundo orden que producen un incremento en las fuerzas internas (fuerzas cortantes y axiales, y a los momentos flectores) y derivas de la estructura, por lo tanto deben considerarse en la evaluación de la estabilidad estructural global y en el diseño mismo de la estructura. Es un efecto no lineal que está asociado con la magnitud de la carga axial aplicada (P) y el desplazamiento (delta).

Hay dos tipos de efectos P-Delta:

- P-“GRANDES” Delta (P- Δ) – efectos sobre la estructura.
- P-“pequeños” Delta (P- δ) – efectos sobre los miembros

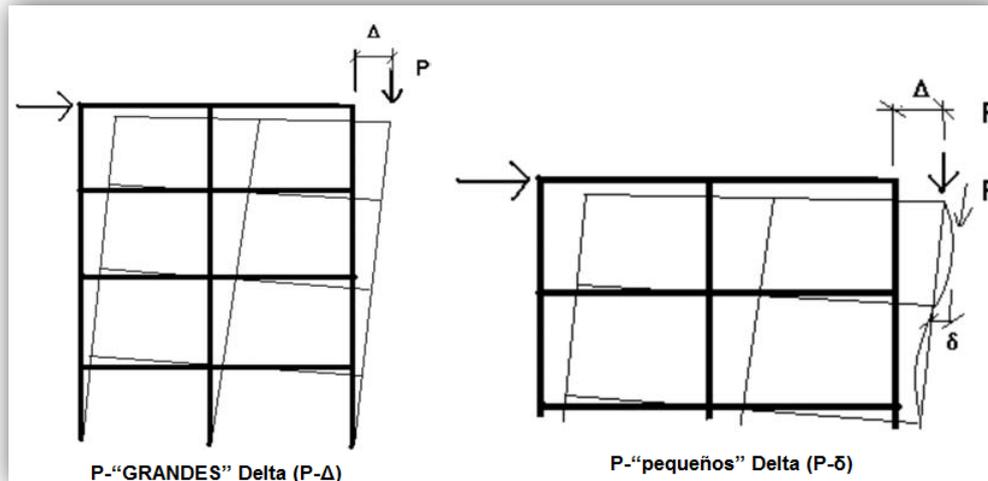


FIGURA 56 – Efectos P-Delta

Esta transformación de coordenadas por lo tanto busca transmitir estos efectos desde el sistema local del elemento al sistema global de coordenadas de la estructura.

Su sintáxis es:

Para un modelo en dos dimensiones:

geomTransf PDelta \$transfTag <-jntOffset \$dXi \$dYi \$dXj \$dYj>

Para un modelo en tres dimensiones:

geomTransf PDelta \$transfTag \$vecxzX \$vecxzY \$vecxzZ <-jntOffset \$dXi \$dYi \$dZi \$dXj \$dYj \$dZj>

Donde:

\$transfTag	Identificador o etiqueta de la transformación de coordenadas
\$vecxzX \$vecxzY \$vecxzZ	Componentes del vector vecxz (vector usado para definir el sistema local de coordenadas). Estos componentes son especificados en base al sistema global de coordenadas X, Y, Z y definen al vector que esta en el plano paralelo al plano XZ del sistema local de coordenadas.
\$dXi \$dYi \$dZi	valores opcionales que se especifican de acuerdo al sistema global de coordenadas para el nudo “i” al final de elemento.
\$dXj \$dYj \$dZj	valores opcionales que se especifican de acuerdo al sistema global de coordenadas para el nudo “j” al final de elemento.

SE DEFINE LAS FIBRAS DE LA SECCION.

set numSubdivCirc 16; # Numero de fibras en la direccion circunferencial

set numSubdivRadcover 4; # Numero de fibras en la direccion radial para el nucleo

```

set numSubdivRadcore 8; # Numero de fibras en la direccion radial para el
recubrimiento

# Se calcula el radio interior y exterior de la seccion.
set intRad [expr $Dc/2000.0]
set extRad [expr $DIA/2.0]
# Area de las varillas de refuerzo transversal
set asb [expr $pi*($DLBAR/1000.0)**2/4]

# SE CREAN LAS SECCIONES QUE FORMARAN LAS PILAS CIRCULARES:
section Fiber 1 {
patch circ 1 $numSubdivCirc $numSubdivRadcover 0 0 $intRad $extRad 0 360; #
Parche para el recubrimiento
patch circ 2 $numSubdivCirc $numSubdivRadcore 0 0 0 $intRad 0 360; # Parche para
el nucleo (hormigon confinado)
layer circ 3 $NLBAR $asb 0 0 $intRad 0 [expr 360-360/$NLBAR]
}
section Aggregator 2 4 Vz 4 Vy 5 T -section 1

#DEFINE LA ORIENTACION DE LOS EJES LOCALES DE UN ELEMENTO CON
RESPECTO A LOS EJES GLOBALES DE LA ESTRUCTURA.
# Transformacion de coordenadas Lineal para los elementos horizontales
geomTransf Linear 1 0 0 1
# Transformacion de coordenadas Lineal para los elementos verticales
geomTransf Linear 2 1 0 0
# Transformacion de coordenadas PDelta para las pilas
geomTransf PDelta 3 1 0 0

```

Se crean los elementos, para la super – estructura elementos elásticos viga – columna, y para las pilas elementos no lineales viga – columna. En OpenSees se pueden crear básicamente dos tipos de elementos no lineales viga – columna:

- **Elementos basados en fuerzas:** las características de estos son: distribuyen la plasticidad a lo largo de toda su longitud “*nonlinearBeamColumn*” y que concentran la plasticidad en los extremos con un interior elástico “*beamWithHinges*”, tal como se lo explico en el ejemplo anterior.
- **Elementos basados en desplazamientos:** a diferencia de los anteriores distribuyen la plasticidad con una curvatura lineal “*dispBeamColumn*”

Para este ejemplo se ha considerado que la plasticidad del elemento está distribuida en toda su longitud *nonlinearBeamColumn*. Este elemento está baso en un proceso no-iterativo o iterativo de fuerzas, la integración para el proceso usa la regla cuadrática de Gauss-Lobatto (dos puntos de integración al final del elemento). Su sintáxis es:

**element nonlinearBeamColumn \$eleTag \$iNode \$jNode \$numIntgrPts
\$secTag \$transfTag <-mass \$massDens> <-iter \$maxIters \$tol>**

Donde:

\$eleTag	etiqueta única del elemento.
\$iNode \$jNode	nudos de inicio y fin del elemento.
\$numIntgrPts	número de puntos de integración a lo largo del elemento.
\$secTag	etiqueta de la sección previamente definida a usar en el elemento.
\$transfTag	etiqueta de transformación de coordenadas previamente definida.
\$massDens	elemento masa densidad (por unidad de longitud) para formar la matriz de masa (opcional, por defecto 0).
\$maxIters	número máximo de iteraciones hasta lograr la convergencia del elemento (opcional, por defecto 1).
\$tol	tolerancia permitida para satisfacer la convergencia (opcional, por defecto 1).

Hay que recordar que al igual que en el ejemplo anterior cuando se crearon los recorders para los elementos **recorder Element** de vigas y columnas, el parámetro **\$secNum** del recorder se refiere al punto de integración del cual se quieren obtener los resultados, mas no la etiqueta de la sección que forma el elemento.

En este ejemplo se analizaran 4 secciones, por tanto los elementos de las pilas tendrán la siguiente forma:

```

                $eleTag $iNode $jNode $numIntgrPts $secTag $transfTag
element nonlinearBeamColumn 22 23 24 4 2 3

```

```

# SE CREAN LOS ELEMENTOS
# Se crea elementos elasticos que corresponden a la super-estructura.
element elasticBeamColumn 1 1 2 $superA [expr $superE*1000] [expr $superE*1000]
$superJ $superIy $superIz 2
element elasticBeamColumn 2 2 3 $superA [expr $superE*1000] [expr $superE*1000]
$superJ $superIy $superIz 1
element elasticBeamColumn 3 3 4 $superA [expr $superE*1000] [expr $superE*1000]
$superJ $superIy $superIz 1
element elasticBeamColumn 4 4 5 $superA [expr $superE*1000] [expr $superE*1000]
$superJ $superIy $superIz 1
element elasticBeamColumn 5 5 6 $superA [expr $superE*1000] [expr $superE*1000]
$superJ $superIy $superIz 1
element elasticBeamColumn 6 6 7 $superA [expr $superE*1000] [expr $superE*1000]
$superJ $superIy $superIz 1
element elasticBeamColumn 7 7 8 $superA [expr $superE*1000] [expr $superE*1000]
$superJ $superIy $superIz 2

```

```

element elasticBeamColumn 8 8 9 $superA [expr $superE*1000] [expr $superE*1000]
$superJ $superIy $superIz 2
element elasticBeamColumn 9 9 10 $superA [expr $superE*1000] [expr $superE*1000]
$superJ $superIy $superIz 1
element elasticBeamColumn 10 10 11 $superA [expr $superE*1000] [expr
$superE*1000] $superJ $superIy $superIz 1
element elasticBeamColumn 11 11 12 $superA [expr $superE*1000] [expr
$superE*1000] $superJ $superIy $superIz 1
element elasticBeamColumn 12 12 13 $superA [expr $superE*1000] [expr
$superE*1000] $superJ $superIy $superIz 1
element elasticBeamColumn 13 13 14 $superA [expr $superE*1000] [expr
$superE*1000] $superJ $superIy $superIz 1
element elasticBeamColumn 14 14 15 $superA [expr $superE*1000] [expr
$superE*1000] $superJ $superIy $superIz 2
element elasticBeamColumn 15 15 16 $superA [expr $superE*1000] [expr
$superE*1000] $superJ $superIy $superIz 2
element elasticBeamColumn 16 16 17 $superA [expr $superE*1000] [expr
$superE*1000] $superJ $superIy $superIz 1
element elasticBeamColumn 17 17 18 $superA [expr $superE*1000] [expr
$superE*1000] $superJ $superIy $superIz 1
element elasticBeamColumn 18 18 19 $superA [expr $superE*1000] [expr
$superE*1000] $superJ $superIy $superIz 1
element elasticBeamColumn 19 19 20 $superA [expr $superE*1000] [expr
$superE*1000] $superJ $superIy $superIz 1
element elasticBeamColumn 20 20 21 $superA [expr $superE*1000] [expr
$superE*1000] $superJ $superIy $superIz 1
element elasticBeamColumn 21 21 22 $superA [expr $superE*1000] [expr
$superE*1000] $superJ $superIy $superIz 2
# Se crea elementos no-lineales para las pilas
element nonlinearBeamColumn 22 23 24 4 2 3
element nonlinearBeamColumn 23 25 26 4 2 3
element nonlinearBeamColumn 24 27 28 4 2 3
element nonlinearBeamColumn 25 29 30 4 2 3

```

Como en todo análisis primero se ejecuta un análisis estático, para ello definimos las cargas que representan el peso propio de la estructura, estas cargas se aplicarán como una carga uniformemente distribuida sobre los elementos de la super – estructura. No vamos a cargar el modelo con cargas laterales, puesto que será en este ejemplo la aceleración sísmica en la base de la estructura la que genere fuerzas internas en los elementos y provoque la respuesta lateral.

El siguiente paso es la creación de los recorders. En este ejemplo analizaremos los desplazamientos máximos y mínimos de la super – estructura y el comportamiento de las pilas en cuatro secciones. Para ello definimos el **recorder EnvelopeNode** que grabará los valores máximos positivos, negativos y absolutos de desplazamiento en la

dirección X y Y (2, 9, 16 y 21), así mismo definimos el **recorder Node** para grabar la historia de desplazamientos de los mismos nudos durante la ejecución del análisis completo.

Para el caso de las pilas, los recorders guardaran la historia de deformación de cada una de las secciones en que fueron divididas. Si recordamos, en la definición de los elementos colocamos 4 puntos de integración, esto dividió en ese número de secciones al elemento, es así que analizaremos cada una de ellas, el tipo de respuesta a obtener es la deformación. Además se crearán recorders para obtener las fuerzas locales de los elementos (este recorder no incluye fuerzas inerciales).

```
# SE DEFINE LA CARGA MUERTA.
# Se definen las cargas
pattern Plain 1 Linear {
# Se cargan los elementos de la super-estructura
eleLoad -ele 2 3 4 5 6 9 10 11 12 13 16 17 18 19 20 -type beamUniform 0 -$superW 0
# Se cargan las pilas
load 24 0 0 -[expr 0.875*$DIA**2*$H/3*$WC+$CBW] 0 0 0
load 26 0 0 -[expr 0.875*$DIA**2*$H/3*$WC+$CBW] 0 0 0
load 28 0 0 -[expr 0.875*$DIA**2*$H/3*$WC+$CBW] 0 0 0
load 30 0 0 -[expr 0.875*$DIA**2*$H/3*$WC+$CBW] 0 0 0
}

# SE CREAN LOS RECORDERS
# Graba la envolvente de los nudos del puente
recorder EnvelopeNode -file EnvelopeDisp.txt -time -node 2 9 16 21 -dof 1 2 disp
# Graba la respuesta de los nudos del puente
recorder Node -file NodeDisp.txt -time -node 2 9 16 21 -dof 1 2 disp
# Graba la respuesta de la pila 1
recorder Element -file Pila1_LForce.txt -time -ele 22 localForce
recorder Element -file Pila1_Def1.txt -time -ele 22 section 1 deformation
recorder Element -file Pila1_Def2.txt -time -ele 22 section 2 deformation
recorder Element -file Pila1_Def3.txt -time -ele 22 section 3 deformation
recorder Element -file Pila1_Def4.txt -time -ele 22 section 4 deformation
# Graba la respuesta de la pila 2
recorder Element -file Pila2_LForce.txt -time -ele 23 localForce
recorder Element -file Pila2_Def1.txt -time -ele 23 section 1 deformation
recorder Element -file Pila2_Def2.txt -time -ele 23 section 2 deformation
recorder Element -file Pila2_Def3.txt -time -ele 23 section 3 deformation
recorder Element -file Pila2_Def4.txt -time -ele 23 section 4 deformation
# Graba la respuesta de la pila 3
recorder Element -file Pila3_LForce.txt -time -ele 24 localForce
recorder Element -file Pila3_Def1.txt -time -ele 24 section 1 deformation
recorder Element -file Pila3_Def2.txt -time -ele 24 section 2 deformation
recorder Element -file Pila3_Def3.txt -time -ele 24 section 3 deformation
recorder Element -file Pila3_Def4.txt -time -ele 24 section 4 deformation
```

```
# Graba la respuesta de la pila 4
recorder Element -file Pila4_LForce.txt -time -ele 25 localForce
recorder Element -file Pila4_Def1.txt -time -ele 25 section 1 deformation
recorder Element -file Pila4_Def2.txt -time -ele 25 section 2 deformation
recorder Element -file Pila4_Def3.txt -time -ele 25 section 3 deformation
recorder Element -file Pila4_Def4.txt -time -ele 25 section 4 deformation
```

Al igual que en los ejemplos anteriores se definen los parámetros para la ejecución del análisis debido a las cargas gravitacionales. Antes de aplicar el registro de aceleración en la base de la estructura es necesario congelar el patron de cargas gravitacionales para que no continúe incrementándose durante el sismo tal como se venía haciendo en los ejemplos anteriores.

A continuación se definen las variables que contendrán los registros sísmicos, el factor multiplicador para esos registros, el intervalo de tiempo que existe entre cada registro y el número de puntos que contiene el archivo. El sismo que se aplicará tiene una duración de 27.5 segundos, cada punto del registro es aceleración (g), el intervalo de tiempo entre cada punto es 0.005s, el archivo contiene 5501 registros, se aplicará un sismo en cada una de las dos direcciones horiozntales X, Y. Los archivos que contienen los registros sísmicos deben estar en la misma carpeta que el tcl.

El registro de aceleraciones se muestra en la siguiente figura:

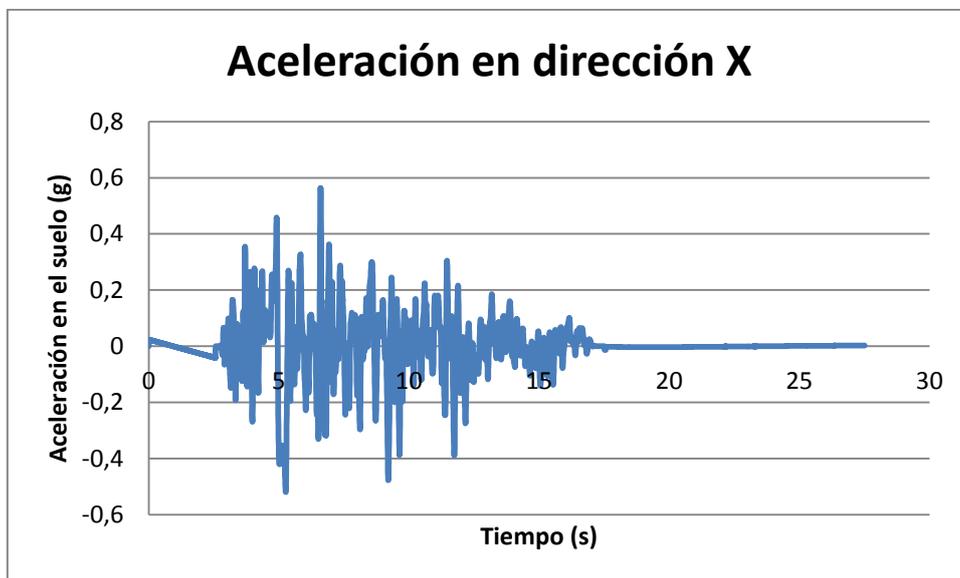


FIGURA 57 – Registro de aceleración en dirección X

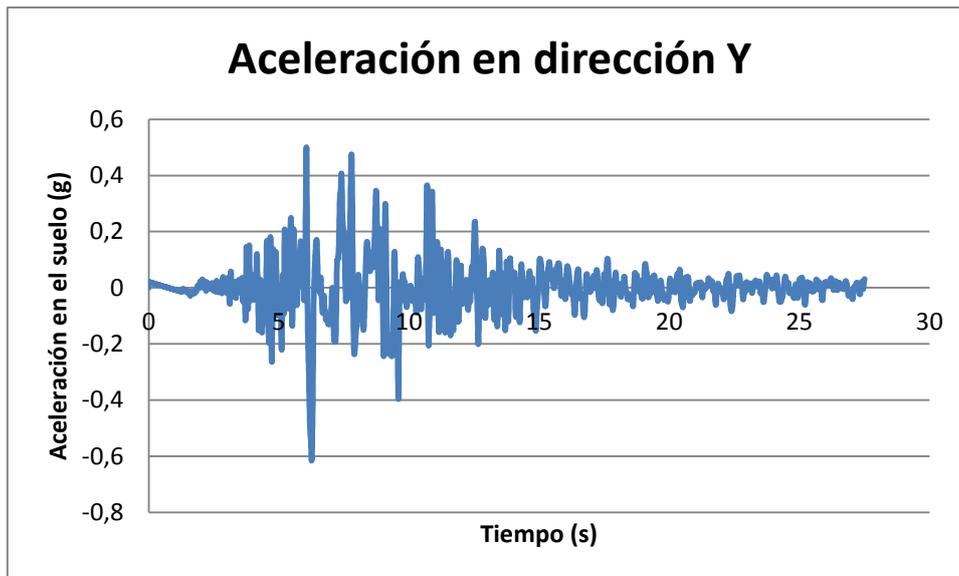


FIGURA 58 – Registro de aceleración en dirección Y

El análisis debe ser de tipo transitorio. El paso del análisis debe seleccionarse adecuadamente para evitar problemas de convergencia, para lo cual se define un contador de tiempo que se detendrá cuando el tiempo de duración del sismo se complete. A más de esto se añade un código que se ejecutará cuando se encuentran problemas en la convergencia de la solución, esto se hará hasta completar el análisis.

```
# ANALISIS GRAVITACIONAL
```

```
test NormDispIncr 1.0e-8 20 0
```

```
algorithm Newton
```

```
system ProfileSPD
```

```
constraints Transformation
```

```
numberer RCM
```

```
integrator LoadControl 1
```

```
analysis Static
```

```
analyze 1
```

```
# Se detiene y congela el analisis gravitacional
```

```
loadConst -time 0; # evita que las cargas gravitacionales sigan incrementando
```

```
# ANALISIS SISMICO
```

```
# Se definen las variables para el analisis sismico
```

```
set sismox "sismox.txt"
```

```
set sismoy "sismoy.txt"
```

```
set Eq_factor 9.81
```

```
set dt 0.005
```

```
set npts 7629
```

```
# Abre los archivos sismicos.
```

```
set serie1 "Series -dt $dt -filePath sismox.txt -factor $Eq_factor"
```

```

set serie2 "Series -dt $dt -filePath sismoy.txt -factor $Eq_factor"

# Crea la carga de diseño con excitacion uniforme
pattern UniformExcitation 2 1 -accel $serie1
pattern UniformExcitation 3 2 -accel $serie2

# Para asignar amortiguamiento Raleigh, tenemos que obtener los periodos de los dos
modos de vibracion
set lambda1 0.02 ;# amortiguamiento asignado al primer modo de vibración
set lambda2 0.02 ;# amortiguamiento asignado al segundo modo de vibración
set eigenvalues [eigen 2] ;# Obtenemos los valores propios de los primeros 2 modos de
vibracion
set omega1 [expr sqrt([lindex $eigenvalues 0])] ;# Las frecuencias de vibracion son la
raiz cuadrada de los valores propios
set omega2 [expr sqrt([lindex $eigenvalues 1])] ;# Los valores propios estan en una
lista, el comando lindex extrae elementos de la lista
set alpha [expr (2*$omega1*$omega2*($omega1*$lambda2-
$omega2*$lambda1))/($omega1**2-$omega2**2)]
set beta [expr 2*($omega1*$lambda1-$omega2*$lambda2)/($omega1**2-
$omega2**2)]

# Inicia el analisis tipo Newmark
integrator Newmark 0.5 0.25 $alpha $beta 0 0
analysis Transient

# Se define el numero de pasos para el analisis
# Inicia un contador de tiempo
set startT [clock seconds]
set res [analyze $npts $dt]
set tFinal [expr $npts * $dt]

# =====
# =====
# EN CASO DE NO CONVERGENCIA, EL CODIGO SIGUIENTE CHEQUEA
# LA FALLA E ITERA SI ES NECESARIO PARA COMPLETAR EL ANALISIS.
set useInitialTangent 0
set ok 0
set tCurrent [getTime]
set dtnew $dt
set nsteps 1

# now check for failure and iterate as necessary to complete record
while { $ok == 0 && $tCurrent < $tFinal } {
  if { $useInitialTangent == 1 } {
    algorithm Newton -initial
    test EnergyIncr 1.0e-12 2500 2
  } else {
    algorithm Newton

```

```

test EnergyIncr 1.0e-12 25 0
}
set res [analyze 1 $dt]
set tCurrent [getTime]
# If analysis fails, try different algorithms
if { $res >= 0 } {
    set useInitialTangent 0
} else {
    if { $useInitialTangent == 1 } {
        # reduce time step size for 10 increments until convergence, otherwise give up
        set quickexit 0
        while { $quickexit == 0 & $dtnew > $dt/5.0e2 } {
            set dtnew [expr $dtnew/2.0]
            set nsteps [expr $nsteps*2]
            if { $nsteps > 150 } {
                set nsteps 150
            }
            puts "reducing step size to $dtnew"
            set res [analyze $nsteps $dtnew]
            set tCurrent [getTime]
            if { $res >= 0 } {
                set useInitialTangent 0
                set quickexit 1
                set nsteps 1
                set dtnew $dt
                puts "resuming old step sizes of $dt"
            }
        }
        if { $quickexit == 0 } {
            puts "Transient: analysis failed -- end of record not reached"
            puts $log "Transient: analysis failed -- end of record not reached"
            set ok -1;
        }
    } else {
        algorithm KrylovNewton
        set res [analyze 1 $dt]
        set tCurrent [getTime]
        if { $res >= 0 } {
            set useInitialTangent 0
        } else {
            set useInitialTangent 1
        }
    }
}
}
# =====
# =====
# FINALIZA EL CONTADOR DE TIEMPO.

```

```
set endT [clock seconds]
puts "Tiempo de ejecucion de analisis: [expr $endT-$startT] seconds."
```

Una vez ejecutado el análisis, obtenemos los siguientes archivos de resultados:

- NodeDisp.txt
- EnvelopeDisp.txt
- Pila1_Def1.txt
- Pila1_Def2.txt
- Pila1_Def3.txt
- Pila1_Def4.txt
- Pila2_Def1.txt
- Pila2_Def2.txt
- Pila2_Def3.txt
- Pila2_Def4.txt
- Pila3_Def1.txt
- Pila3_Def2.txt
- Pila3_Def3.txt
- Pila3_Def4.txt
- Pila4_Def1.txt
- Pila4_Def2.txt
- Pila4_Def3.txt
- Pila4_Def4.txt

Para analizar los resultados hay que recordar la geometría de la estructura, la orientación de los elementos, de los ejes locales y la orientación de los ejes globales, así como también la forma como se definieron los recorders.

Con el archivo de desplazamientos podemos obtener las siguientes gráficas de desplazamiento en función del tiempo para los nudos 2, 9 , 16 y 21, en las dos direcciones horizontales X, Y.



FIGURA 59 – Historia de desplazamiento en dirección X del nudo 2



FIGURA 60 – Historia de desplazamiento en dirección Y del nudo 2



FIGURA 61 – Historia de desplazamiento en dirección X del nudo 9



FIGURA 62 – Historia de desplazamiento en dirección Y del nudo 9

Los valores máximos de respuesta pueden obtenerse del análisis detallado de las graficas presentadas o del archivo EnvelopeDisp.txt que contiene la envolvente de desplazamientos.

De acuerdo a como se especificó en el recorder, las cuatro primeras columnas corresponden al nudo 2, de la 5 – 8 al nudo 9, de la 9 – 12 al nudo 16 y la 13 – 16 al nudo 21. La primera fila contiene los valores máximos negativos (es decir horizontales hacia la izquierda), la segunda los máximos positivos (es decir horizontales hacia la derecha) y la tercera los máximos absolutos (sin importar que fuesen hacia la izquierda o derecha).

tiempo x2 tiempo y2 tiempo x9 tiempo y9

6.73	-0.193232	9.025	-0.701764	6.73	-0.193211	9.02	-0.690174
5.775	0.355379	7.02	0.22586	5.775	0.355394	7.02	0.229236
5.775	0.355379	9.025	0.701764	5.775	0.355394	9.02	0.690174
...							
tiempo	x16	tiempo	y16	tiempo	x21	tiempo	y21
6.73	-0.193193	8.935	-0.679217	6.73	-0.193197	8.835	-0.672232
5.775	0.355412	7.02	0.2326	5.775	0.355412	7.015	0.236102
5.775	0.355412	8.935	0.679217	5.775	0.355412	8.835	0.672232

Por ejemplo se lee que el máximo desplazamiento ocurriónn en el nudo 2 en la dirección Y y fue de 0.70m hacia la izquierda y ocurrió en el instante 9.025s durante el sismo. Esto concuerda con lo que se observa en las graficas que muestran la historia de desplazamiento.

De acuerdo al los recorderes definidos en el modelo, los archivos Pila1_LForce.txt, Pila2_LForce.txt , Pila3_LForce.txt y Pila4_LForce.txt contiene las fuerzas en las pilas en sus ejes locales. La primera fila del archivo contiene las fuerzas al final de la aplicación de las cargas gravitacionales y el resto de filas corresponde a la historia de fuerzas durante la aplicación del sismo.

Ya que en el archivo se registraron fuerzas locales, cada archivo contiene trece columnas, donde luego de la primera columna de tiempo, las siguientes sies corresponden a las fuerzas en en nudo i (F_x , F_y , F_z , M_x , M_y y M_z), y lo mismo para las seis columnas restantes pero esta vez para el nudo j. Esta información puede exportarse a Excel para su procesamiento. Por ejemplo, a continuación se presenta la historia de fuerza axial F_x en el extremo inicial (izquierdo) de la pila 3:

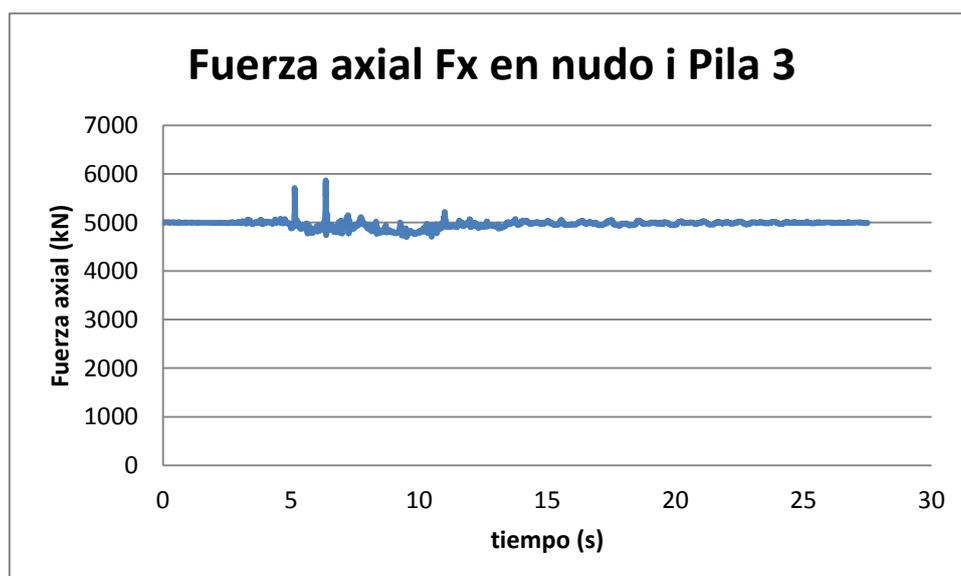


FIGURA 63 – Fuerza axial F_x en el nudo i de la Pila 3

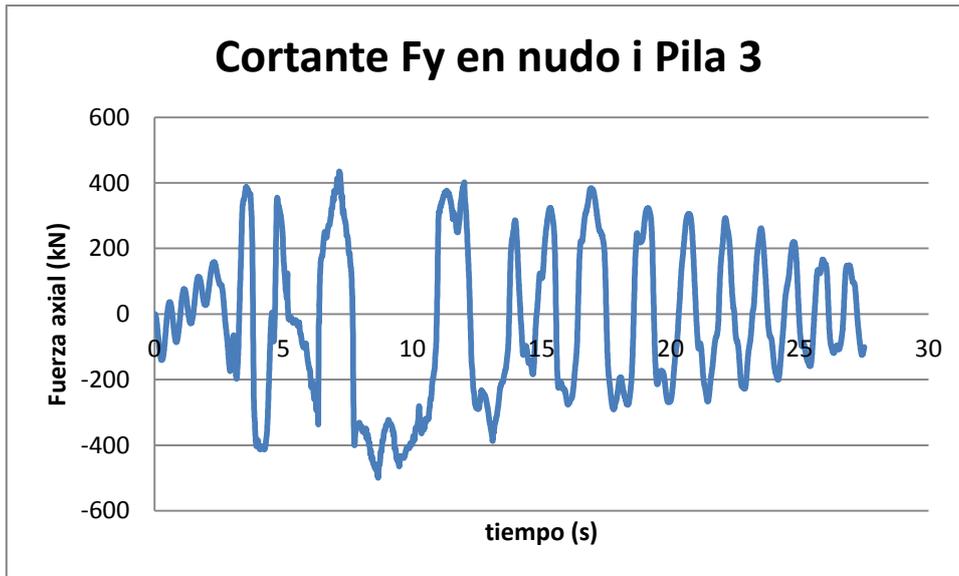


FIGURA 64 – Cortante F_y en el nudo i de la Pila 3

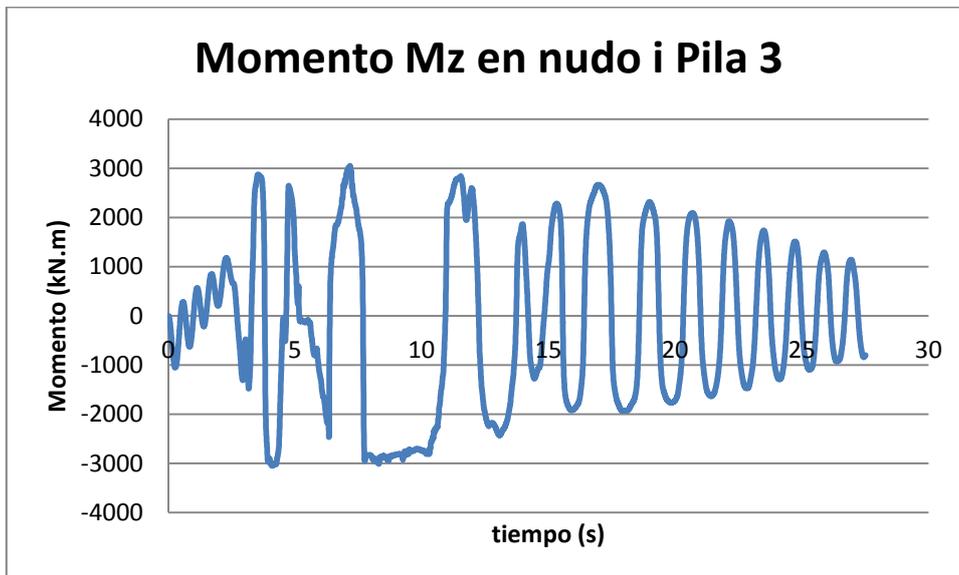


FIGURA 65 – Momento M_z en el nudo i de la Pila 3

De la misma manera que se obtuvieron estas gráficas se pueden obtener para el resto de fuerzas, y el resto de pilas.

En los recorders restantes, tal como se definió en el modelo están registradas las respuestas de deformación en cada una de las secciones en que se dividió las pilas. Los archivos contienen 7 columnas donde luego de la primera columna de tiempo, las tres siguientes corresponden al extremo inicial de la sección y las otras tres al extremo final de la sección, la segunda columna de las tres de cada extremo es la deformación y la tercera la curvatura. A continuación se presentan la historia de deformaciones y curvatura para cada sección de la pila 4.



FIGURA 66 – Deformación en extremo i de cada sección de la Pila 4



FIGURA 67 – Deformación en extremo j de cada sección de la Pila 4

De las gráficas se puede ver claramente que la mayor deformación se encuentra en el extremo inicial de la sección 1, es decir en la parte inferior de la pila, esto es lógico debido a que son las bases de las pilas las que están sometidas principalmente a la acción del sismo.

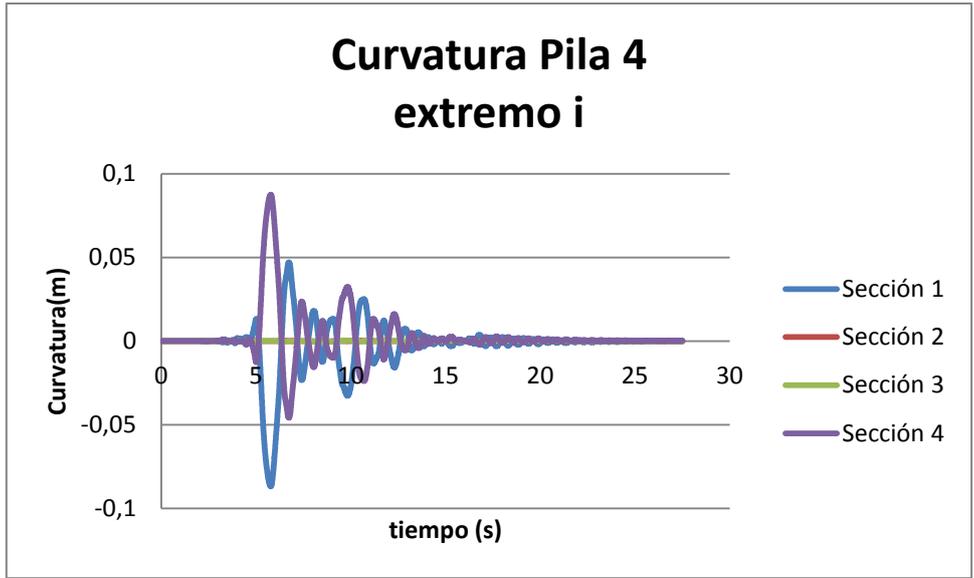


FIGURA 68 – Curvatura en extremo i de cada sección de la Pila 4

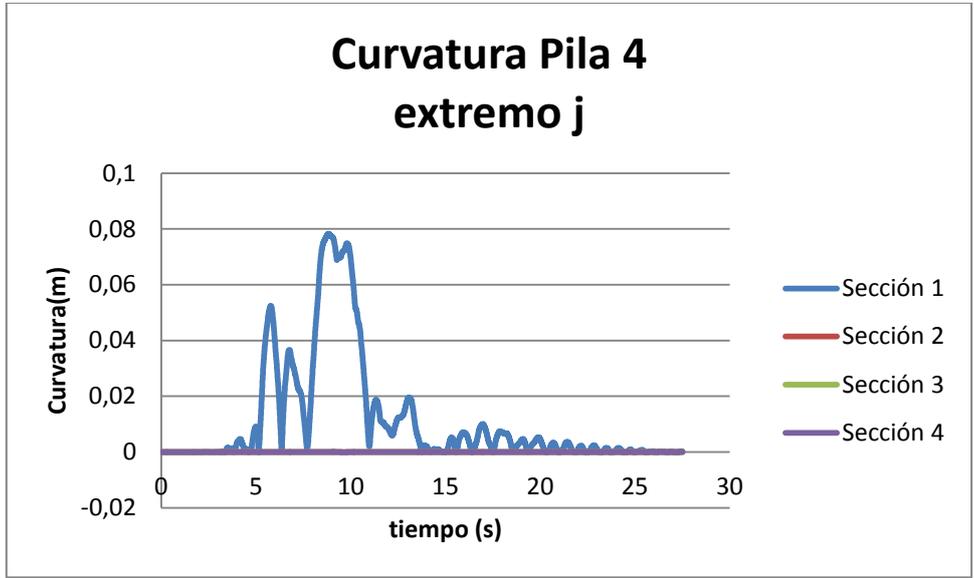


FIGURA 69 – Curvatura en extremo j de cada sección de la Pila 4

De igual forma que las deformaciones se nota la curvatura mayor en la sección 1, de manera que se comprueba la consistencia de los resultados con lo que ocurre en la realidad.

CAPÍTULO VII

ANÁLISIS INELÁSTICO BAJO CARGAS DINÁMICAS

7. ANÁLISIS DINÁMICOS

7.1. INTRODUCCIÓN

El enfoque de este capítulo está orientado al caso de estructuras sujetas a acciones laterales ocasionadas por movimientos en la base de la estructura. El análisis dinámico es importante porque ese movimiento oscilatorio produce una modificación de las tensiones y deformaciones existentes que deben tenerse en cuenta por ejemplo para lograr un diseño sísmico adecuado.

Los análisis dinámicos se dividen usualmente en tres grandes grupos:

- Análisis modal espectral, de uso ingenieril más común.
- Análisis de historia en el tiempo.
- Análisis en el dominio de las frecuencias.

Donde el más completo o realista es el de historia en el tiempo, ya que durante el movimiento de una edificación por la acción sísmica, las sollicitaciones sobre aquella son realmente de dirección diversa, y puede a través de este análisis observarse cual es el comportamiento que tiene la estructura conforme el paso del tiempo.

Se ha llegado a considerar que el movimiento del suelo tiene seis componentes de movimiento independientes, tres traslacionales y tres rotacionales. Dentro de estas componentes, las traslacionales en las direcciones horizontales suelen ser tomadas en cuenta, en forma independiente, para fines de tener condiciones de carga en los análisis, dado que por lo general son los más importantes.

Los programas de análisis estructurales más comunes no realizan análisis incluyendo efectos de segundo orden (denominado también no linealidad geométrica), es decir, la consideración de esfuerzos adicionales debidos a la modificación de los ejes causada por las deformaciones. Esto no ocurre en OpenSees, la consideración de estos efectos así como la creación de elementos con comportamiento no lineal, cualquier tipo de geometría y variados tipos de materiales si son posibles realizarlos con OpenSees.

7.2. ANÁLISIS DINÁMICOS EN OPENSEES

OpenSees cuenta con cinco tipos de análisis dinámicos disponibles para incorporar en los modelos, los cuales son:

- Análisis dinámico bajo excitación sísmica uniforme “Dynamic Uniform Earthquake Ground Motion (typical)”, es el más típico que se realiza. Lo que hace es aplicar un registro de sísmico de aceleración sobre todos los nudos de la base de la estructura en una misma dirección.

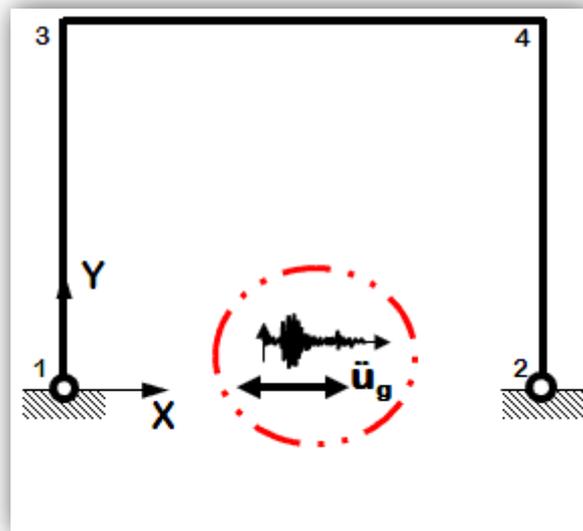


FIGURA 70 – Análisis dinámico bajo excitación sísmica uniforme

- Análisis dinámico bajo excitación sísmica bidireccional “Dynamic Bidirectional Earthquake Ground Motion (typical)”, al igual que el anterior también es un

análisis que se realiza con frecuencia. A diferencia del anterior aplica dos excitaciones sobre la base de la estructura, una en cada dirección “birreccional”.

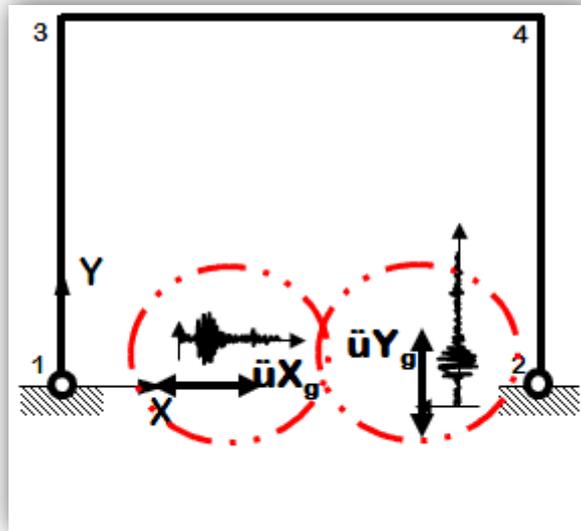


FIGURA 71 – Análisis dinámico bajo excitación sísmica bidireccional

- Análisis dinámico bajo excitación sísmica múltiple “Dynamic Multiple-Support Earthquake Ground Motion”, permite asignar n registros sísmicos distintos a los “n” nudos de la base de la estructura, los cuales son aplicados en la dirección deseada.

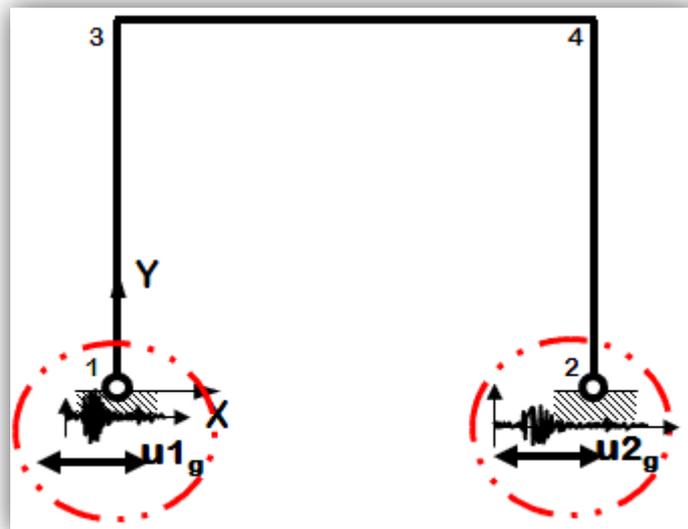


FIGURA 72 – Análisis dinámico bajo excitación sísmica múltiple

- Análisis dinámico bajo excitación senoidal uniforme “Dynamic Uniform Sine-Wave Ground Motion”, de manera similar al primer tipo este también se usa para aplicar una excitación uniforme sobre todos los nudos de la base de la

estructura en una misma dirección, con la diferencia que la excitación para este caso corresponde a la de una función senoidal.

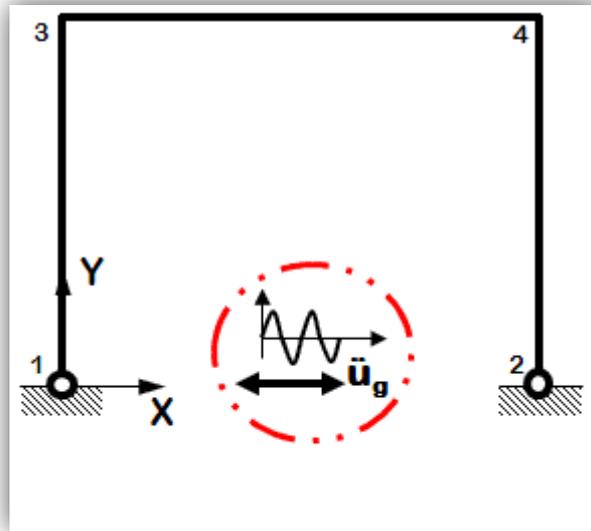


FIGURA 73 – Análisis dinámico bajo excitación senoidal

- Análisis dinámico bajo excitación senoidal múltiple “Dynamic Multiple-Support Sine-Wave Ground Motion”, se usa para asignar n funciones senoidales como nudos base tiene el modelo, aplicados en la dirección deseada.

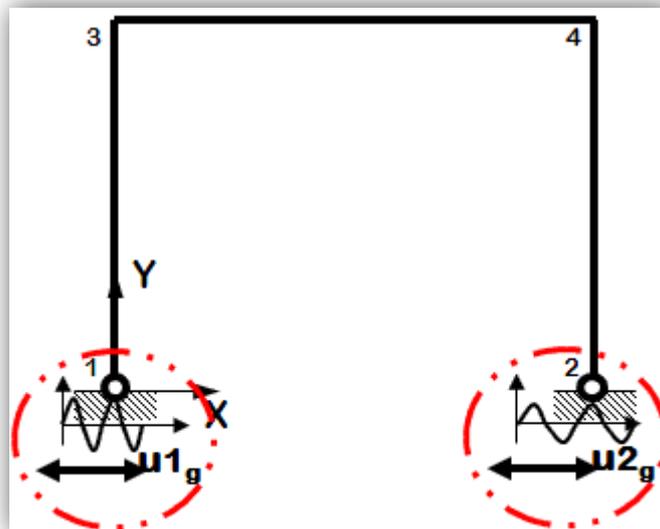


FIGURA 74 – Análisis dinámico bajo excitación senoidal múltiple

A continuación se presenta un modelo de un pórtico en sobre el que se ejecutarán los cinco tipos de análisis dinámicos con que dispone OpenSees.

7.3.EJEMPLO. ANÁLISIS DINÁMICOS DE UN PÓRTICO EN 2D

El pórtico que se analizará tiene la geometría que se muestra en la siguiente figura. **No se encuentra el origen de la referencia.**, las columnas tienen una altura de 3m y viga una longitud de 5m, se modelarán como elementos inelásticos con y sin fibras las columnas y la viga como un elemento elástico. Se aplicará una carga uniformemente distribuida sobre la viga, registros de aceleración sísmicos y ondas senoidales en la base de la estructura. Al tratarse de un análisis dinámico se debe asignar masas a todos los grados de libertad libres en el modelo, el peso de la estructura estará distribuida uniformemente en los nudos.

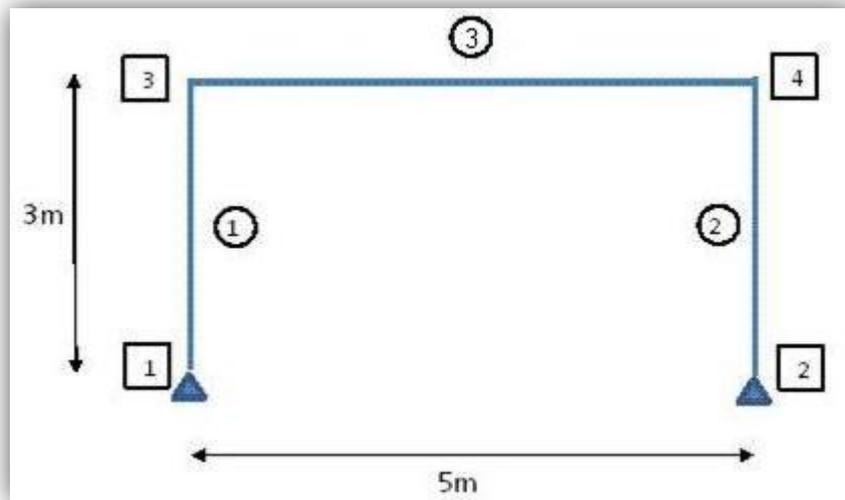


FIGURA 75 – Geometría del modelo

El registro sísmico a aplicar tiene una duración de 27.5 segundos, pero debido al tiempo que tarde tarde el programa hasta la finalización del análisis solamente se ejecutará para los 10 primeros segundos de duración del sismo. La onda senoidal también se la tendrá una duración de 10 segundos.

Para este ejemplo se crearon 6 archivos Tcl, donde el primero es el que contiene el modelo en sí (desde la geometría hasta la aplicación de las cargas gravitacionales), los cinco restantes contienen las características y parámetros necesarios para la ejecución de cada tipo de análisis dinámico.

A continuación se presenta el código Tcl de cada uno de estos archivos creados para este ejemplo con comentarios que ayudan a explicar el modelo:

```
#####  
#Ejemplo9.tcl #  
#Análisis Inelástico de un Pórtico plano con Elementos no lineales #  
#con o sin sección de fibras #  
#Unidades: kN m s #
```

```
#####
wipe                ;# Este comando borra todos los objetos existentes
model basic -ndm 2 -ndf 3    ;# 2 dimensiones; 3 GDL por nudo
set pi [expr acos(-1.0)]
set g 9.81
```

Como se puede ver en este ejemplo se está generando el modelo de manera distinta a lo que se venía haciendo en los ejemplos anteriores, aquí el modelo no está restringido únicamente a la altura de 3m y ancho de 5m, estos valores pueden modificarse en cualquier momento y obtener un pórtico con dimensiones distintas ya que las coordenadas cambiarán automáticamente de acuerdo a los valores nuevos ingresados.

```
# Se define la geometria-----
set LCol 3;           # Altura de las columnas (m)
set LBeam 5;         # longitud de la viga (m)
set Weight 100;      # Carga sobre la viga (N)
# Se define la geometria de la seccion
set HCol 0.5;        # Altura de columna
set BCol 0.4;        # Base de columna
set HBeam 0.6;       # Altura de viga
set BBeam 0.4;       # Base de viga

# Calculo de peso y masa
set PCol [expr $Weight/2];           # Peso en la cabeza de cada columna
set Mass [expr $PCol/$g];           # Masa en la cabeza de la columna

# Calculo area e inercia de los elementos
set ACol [expr $BCol*$HCol]
set ABeam [expr $BBeam*$HBeam]
set IzCol [expr 1./12.*$BCol*pow($HCol,3)]
set IzBeam [expr 1./12.*$BBeam*pow($HBeam,3)]

# COORDEANDAS DE LOS NUDOS
node 1 0 0
node 2 $LBeam 0
node 3 0 $LCol
node 4 $LBeam $LCol

# CONDICIONES DE BORDE
fix 1 1 1 0
fix 2 1 1 0
fix 3 0 0 0
fix 4 0 0 0

# MASA EN LOS NUDOS:
```

```
mass 3 $Mass 0 0
mass 4 $Mass 0 0
```

Se crean los elementos inelásticos para las columnas con 2 opciones, la primera de sección sin fibras y la segunda sección con fibras. Para seleccionar el tipo de elemento que se desea generar las columnas basta con cambiar la variable **ElemTipo** entre 1 o 2.

No se dará explicación sobre la creación de estos elementos puesto que en ejemplos anteriores ya se lo ha hecho.

```
# SE CREAM LOS ELEMENTOS INELASTICOS
set ElemTipo 1
switch $ElemTipo {
  1 {
    puts " --- El analisis se realizara con elementos no lineales sin fibras ---"
    # Definicion de materiales, secciones y elementos-----
    set fc 24000; # resistencia a la compresión no confinada del concreto (kPa)
    set Ec [expr 4700*sqrt($fc/1000)*1000]; # Modulo de elasticidad del concreto

# SECCION DE COLUMNAS:
# Calculo de los parametros de rigidez
set MyCol 130000;          # Momento de fluencia
set PhiYCol 0.65e-4;     # Curvatura de fluencia
set EIColCrack [expr $MyCol/$PhiYCol];    # Inercia agrietada de la seccion
set b 0.01;              # Coeficiente post - fluencia
uniaxialMaterial Steel01 2 $MyCol $EIColCrack $b;          # comportamiento
bilineal para flexion
uniaxialMaterial Elastic 3 [expr $Ec*$ACol];              # este material se usara
para obtener la respuesta axial-force-strain
section Aggregator 1 3 P 2 Mz; # combina el comportamiento axial y a flexion en una
seccion (no P-M )

# SECCION DE VIGA:
section Elastic 2 $Ec $ABeam $IzBeam;          # seccion elastica para la viga

# Transformacion de coordenadas:
geomTransf PDelta 1 ; # Solamente las columnas pueden tener efectos PDelta (efectos
gravitacionales)
geomTransf Linear 2

# Conectividad de los elementos:
set numIntgrPts 5;      # numero de puntos de integracion para los elementos basados
en fuerzas
element nonlinearBeamColumn 1 1 3 $numIntgrPts 1 1
element nonlinearBeamColumn 2 2 4 $numIntgrPts 1 1
```

```

element nonlinearBeamColumn 3 3 4 $numIntgrPts 2 2
}
2 {
puts " --- El analisis se realizara con elementos no lineales con fibras ---"
# Definicion de materiales, secciones y elementos-----
#Propiedades del hormigón y acero:
set fc 24000; # resistencia a la compresión no confinada del concreto (kPa)
set Ec [expr 4700*sqrt($fc/1000)*1000]; # Modulo de elasticidad del concreto
set fy 420000 ; # resistencia de fluencia del acero longitudinal (kPa)
set fyh 602000 ;# resistencia última del acero de estribos (kPa)
set Es 200000000 ; # modulo de elasticidad del acero
set shr 0.003 ; # razón de rigidez post fluencia (parámetro a usar para definir el material Steel01)
# COLUMNAS
set db 0.016 ;# diametro de las barras longitudinales (m)
set nbltb 8 ;# # número de barras arriba y abajo (arriba + abajo)
set nbll 4 ;# # número de barras laterales (todas, izq+der.)
set dbt 0.01 ;# Diametro de las barras en estribos
set nvsx 2 ;# numero de ramas de estribos resistiendo corte en la direccion X
set nvsy 2 ;# numero de ramas de estribos resistiendo corte en la direccion Y
set sv 0.1 ;# espaciamiento de los estribos (m)
set cover 0.04 ; # recubrimiento hasta los estribos (m)
# Propiedades del hormigon confinado a usar en la sección de vigas de acuerdo al
modelo de mander,
set eco 0.002
set epss 0.004
set esm 0.1
set sp [expr $sv-$dbt]
set rs [expr ($nvsx*$dbt**2*(pi/4)*($BCol-
2*$cover)+$nvsy*$dbt**2*(pi/4)*($HCol-2*$cover))/((($BCol-2*$cover)*($HCol-
2*$cover)*$sv)]
set areaBar [expr $db**2*(pi/4)]
set areacon [expr ($BCol-2*$cover)*($HCol-2*$cover)]
set rcc [expr $areaBar*( $nbltb+$nbll)/$areacon]
set wi 0.2
set ke [expr ((1 - $wi**2 / (6 * $areacon)) * (1 - $sv / (2 * $BCol)) * (1 - $sv / (2 *
$HCol))) / (1 - $rcc)]
set fpl [expr 1/2.0*$ke*$rs*$fyh]
set fpcc [expr (-1.254 + 2.254 * (1 + 7.94 * $fpl / $fc) ** 0.5 - 2 * $fpl / $fc) * $fc]
set ecc [expr $eco * (1 + 5 * ($fpcc / $fc - 1))]
set Esec [expr $fpcc / $ecc]
set r [ expr $Ec / ($Ec - $Esec)]
set ecu [ expr 1.5 * (0.004 + 1.4 * $rs * $fyh * $esm / $fpcc) ]
set x [expr $ecu / $ecc ]
set fcu [expr $fpcc * $x * $r / ($r - 1 + $x ** $r)]

# Materiales para columnas

```

```

uniaxialMaterial Concrete01 1 -$fc -$eco 0 -$epss ; # Material para fibras de concreto
sin confinamiento
uniaxialMaterial Steel01 2 $fy $Es $shr ; # Material para acero de refuerzo longitudinal
uniaxialMaterial Concrete01 3 -$fpcc -$ecc -$fcu -$ecu ; # Material para fibras de
concreto confinado
uniaxialMaterial Elastic 4 $Ec

```

```

# Creacion de las secciones
# COLUMNAS
section Fiber 1 {
set y0 [expr $BCol/2]
set y1 [expr $y0-$cover]
set z0 [expr $HCol/2]
set z1 [expr $z0-$cover]
set numSubdivCoreZ 16 ; # número de fibras de concreto confinado en la direccion
local Z
set numSubdivCoreY 16 ; # número de fibras de concreto confinado en la direccion
local Y
set numSubdivCover 2 ; # número de fibras en el recubrimiento
# parches para recubrimiento lateral
patch quad 1 $numSubdivCover [expr $numSubdivCoreZ+2*$numSubdivCover] -$y0 -
$z0 -$y1 -$z0 -$y1 $z0 -$y0 $z0
patch quad 1 $numSubdivCover [expr $numSubdivCoreZ+2*$numSubdivCover] $y1 -
$z0 $y0 -$z0 $y0 $z0 $y1 $z0
# parches para recubrimiento inferior y superior
patch quad 1 $numSubdivCoreY $numSubdivCover -$y1 -$z0 $y1 -$z0 $y1 -$z1 -$y1 -
$z1
patch quad 1 $numSubdivCoreY $numSubdivCover -$y1 $z1 $y1 $z1 $y1 $z0 -$y1 $z0
# parche de fibras para el hormigon confinado
patch quad 3 $numSubdivCoreY $numSubdivCoreZ -$y1 -$z1 $y1 -$z1 $y1 $z1 -$y1
$z1
# Acero superior e inferior
fiber 0 $z1 [expr ($nbltb/2) * $areaBar] 2
fiber 0 -$z1 [expr ($nbltb/2) * $areaBar] 2
# Acero lateral
fiber $y1 0 [expr ($nbll/2) * $areaBar] 2
fiber -$y1 0 [expr ($nbll/2) * $areaBar] 2
}
# Se añade la respuesta a cortante del material elastico 8 a la seccion creada 3
section Aggregator 2 4 Vy -section 1

```

```

# SECCION DE VIGA:
section Elastic 3 $Ec $ABeam $IzBeam; # seccion elastica para la viga

```

```

# Transformacion de coordenadas:
geomTransf Linear 1 ; # Solamente las columnas pueden tener efectos PDelta (efectos
gravitacionales)
geomTransf Linear 2

```

```

# Conectividad de los elementos:
set numIntgrPts 5;      # numero de puntos de integracion para los elementos basados
en fuerzas
element nonlinearBeamColumn 1 1 3 $numIntgrPts 2 1
element nonlinearBeamColumn 2 2 4 $numIntgrPts 2 1
element nonlinearBeamColumn 3 3 4 $numIntgrPts 3 2
}
}
puts "Modelo Cosntruido con exito"

```

Una vez creados los elementos se define los recorders que grabarán la respuesta de los elementos y nudos del modelo

```

# DEFINICION DE LOS RECORDERS -----
recorder Node -file DespLibres.txt -time -node 3 4 -dof 1 2 3 disp;# desplazamiento de
los nudos libres
recorder Node -file DespBase.txt -time -node 1 2 -dof 1 2 3 disp;# desplazamiento de
los nudos de la base
recorder Node -file Reaccion.txt -time -node 1 2 -dof 1 2 3 reaction;# reacciones en los
soportes
recorder Drift -file Deriva.txt -time -iNode 1 2 -jNode 3 4 -dof 1 -perpDirn 2;# deriva
lateral
recorder Element -file FuerzasCol.txt -time -ele 1 2 globalForce;# fuerzas globales en
columnas
recorder Element -file FuerzasViga.txt -time -ele 3 globalForce; # fuerzas globales en
vigas
recorder Element -file FuerzaColSec1.txt -time -ele 1 2 section 1 force; # Fuerzas en la
seccion de la columna (axial y momento) nudo i
recorder Element -file DefColSec1.txt -time -ele 1 2 section 1 deformation;#
Deformaciones en la seccion (axial y curvatura) nudo i
recorder Element -file FuerzaColSec$numIntgrPts.txt -time -ele 1 2 section
$numIntgrPts force; # Fuerzas en la seccion de la columna (axial y momento) nudo j
recorder Element -file DefColSec$numIntgrPts.txt -time -ele 1 2 section $numIntgrPts
deformation; # Deformaciones en la seccion (axial y curvatura) nudo j
recorder Element -file FuerzaVigaSec1.txt -time -ele 3 section 1 force; # Fuerzas en la
seccion de la viga (axial y momento) nudo i
recorder Element -file DefVigaSec1.txt -time -ele 3 section 1 deformation;#
Deformaciones en la seccion (axial y curvatura) nudo i
recorder Element -file FuerzaVigaSec$numIntgrPts.txt -time -ele 3 section
$numIntgrPts force; # Fuerzas en la seccion de la viga (axial y momento) nudo j
recorder Element -file DefVigaSec$numIntgrPts.txt -time -ele 3 section $numIntgrPts
deformation;# Deformaciones en la seccion (axial y curvatura) nudo j

```

Se define la carga gravitacional que actúa sobre la estructura, los parámetros para el análisis gravitacional, y se congela la carga para cualquiera de los siguientes análisis dinámicos.

```
# Cargas gravitacionales -----
set WzBeam [expr $Weight/$LBeam];
pattern Plain 1 Linear {
  eleLoad -ele 3 -type -beamUniform -$WzBeam
}
# PARAMETROS PARA EL ANALISIS GRAVITACIONAL
set Tol 1.0e-8
constraints Plain
numberer Plain
system BandGeneral
test NormDispIncr $Tol 6
algorithm Newton
integrator LoadControl 1
analysis Static
analyze 1

# Se mantiene constante la carga gravitacional y se reinicia el tiempo a cero
loadConst -time 0.0
puts "Análisis gravitacional completado"
```

Con la variable “**Atipo**” se selecciona el tipo de análisis dinámico que se desea ejecutar sobre el modelo creado. Lo que hace ésta variable es llamar al archivo Tcl que contiene los parámetros del análisis dinámico seleccionado.

```
# Seleccionar el tipo de analisis que se desea
set Atipo 1
switch $Atipo {
  1 {
    puts " --- Analisis Dinamico bajo carga sismica Uniforme ---"
    source Eq_Uniforme.tcl}
  2 {
    puts " --- Analisis Dinamico bajo carga sismica Bidireccional ---"
    source Eq_Bidireccional.tcl}
  3 {
    puts " --- Analisis Dinamico bajo carga sismica en Multiples Apoyos ---"
    source Eq_Multiple.tcl}
  4 {
    puts " --- Analisis Dinamico bajo funcion Senoidal Uniforme ---"
    source Senoidal_Uniforme.tcl}
  5 {
```

```
puts " --- Analisis Dinamico bajo funcion Senoidal en Multiples Apoyos ---"
source Senoidal_Multiple.tcl}
}
```

```
#####
#FIN
#####
```

Análisis Tipo 1

A continuación se presenta el código Tcl para el Analisis Dinamico bajo carga sismica Uniforme, el cual está contenido dentro del archivo Eq_Uniforme.tcl:

```
#####
#Example9. #
#Portico 2D. #
#PARAMETROS PARA ANALISIS DINAMICO BAJO EXCITACION SISMICA
UNIFORME #
#####
```

Tal como se hizo en ejemplos anteriores se define los parámetros para el análisis dinámico, se calcula el amortiguamiento viscoso en la estructura con ayuda del modelo de Rayleigh. Esta parte del código es común para todos los tipos de análisis dinámico como se verá más adelante.

```
# PARAMETROS PARA ANALISIS DINAMICO
constraints Transformation
numberer RCM
system BandGeneral
test EnergyIncr 1.e-8 10 0
algorithm ModifiedNewton
integrator Newmark 0.5 0.25
analysis Transient

# Parametros para amortiguamiento por Rayleigh-----
set xDamp 0.02; # amortiguamiento
set lambda [eigen 1]; # se obtiene el primer modo de vibracion
set omega [expr pow($lambda,0.5)];
set alphaM 0.; # M-prop. amortiguamiento; D =
alphaM*M
```

```

set betaKcurr 0.;                # K-proportional amortiguamiento;
+beatKcurr*KCurrent
set betaKcomm [expr 2.*$xDamp/($omega)]; # K-prop. parametro para
amortiguamiento; +betaKcomm*KlastCommitt
set betaKinit 0.;                # initial-stiffness proportional amortiguamiento
+beatKinit*Kini
rayleigh $alphaM $betaKcurr $betaKinit $betaKcomm; # AMORTIGUAMIENTO DE
RAYLEIGH
set T1 [expr 2.0*$pi/$omega] ;# Periodo del primer modo de vibración
puts "T1: $T1 segundos"

```

Se define el patrón de carga como un patrón Uniforme con el comando **pattern UniformExcitation** este comando permite aplicar una excitación uniforme al modelo en una cierta dirección, su sintáxis es:

pattern UniformExcitation \$patternTag \$dir -accel \$tsTag <-vel0 \$ver0>

Donde:

\$patternTag	etiqueta única para el patrón de carga
\$dir	dirección sobre la que actuará la excitación
\$tsTag	etiqueta de la serie que contiene la la historia de aceleración
\$ver0	opcional, lavelocidad inicial por defecto es 0

La respuesta obtenida en los nudos para este tipo de análisis son valores RELATIVOS, y no los absolutos obtenidos del caso de excitación múltiple “pattern MultipleSupport”.

En este análisis se aplicará un registro sísmico en la base de la estructura en la dirección X (horizontal).

El código que está a continuación se usa para realizar un proceso iterativo en caso de la no convergencia del análisis, este código también es común en el resto de archivos tcl.

```

# Parametros del registro sismico (afecta a todos los nudos de la base en una sola
direccion)
# estos parametros son solamente para exitacion Uniforme
set sismox "sismox.txt"
set factorX 1.1; # se amplifica el sismo en un 10%
set TmaxAnalysis 10; # Tiempo de duracion maxima del analisis, el analisis durara 10
segundos
set dt 0.005;                # paso del tiempo para el analisis

# Se crea el patron para la exitacion uniforme
set serie1 "Series -dt $dt -filePath $sismox -factor [expr $factorX*$g]"; # serie en
direccion X
pattern UniformExcitation 2 1 -accel $serie1

```

```

# Se define el numero de pasos para el analisis
set Nsteps [expr int($TmaxAnalysis/$dt)];
set ok [analyze $Nsteps $dt];
# Proceso iterativo en caso de no convergencia
if {$ok != 0} {
    # change some analysis parameters to achieve convergence
    # performance is slower inside this loop
    # Time-controlled analysis
    set ok 0;
    set controlTime [getTime];
    while {$controlTime < $TmaxAnalysis && $ok == 0} {
        set controlTime [getTime]
        set ok [analyze 1 $dt]
        if {$ok != 0} {
            puts "Intentado con Newton with Initial Tangent .."
            test NormDispIncr $Tol 1000 0
            algorithm Newton -initial
            set ok [analyze 1 $dt]
            test EnergyIncr 1.e-8 1000 0
            algorithm ModifiedNewton
        }
        if {$ok != 0} {
            puts "Intentado con Broyden .."
            algorithm Broyden 8
            set ok [analyze 1 $dt]
            algorithm ModifiedNewton
        }
        if {$ok != 0} {
            puts "Intentado con NewtonWithLineSearch .."
            algorithm NewtonLineSearch .8
            set ok [analyze 1 $dt]
            algorithm ModifiedNewton
        }
    }
}; # end if ok !0
puts "Análisis Ground Motion finalizado. Tiempo del análisis: [getTime]"

```

Análisis Tipo 2

A continuación se presenta el código Tcl para el Análisis Dinámico bajo carga sísmica bidireccional, el cual está contenido dentro del archivo Eq_Bidireccional.tcl:

```

#####
#Example9. #
#Portico 2D. #

```

```
#PARAMETROS PARA ANALISIS DINAMICO BAJO EXCITACION SISMICA
BIDIRECCIONAL #
#####
```

```
# PARAMETROS PARA ANALISIS DINAMICO
constraints Transformation
numberer RCM
system BandGeneral
test EnergyIncr 1.e-8 10 0
algorithm ModifiedNewton
integrator Newmark 0.5 0.25
analysis Transient
```

```
# Parametros para amortiguamiento por Rayleigh-----
set xDamp 0.02; # amortiguamiento
set lambda [eigen 1]; # se obtiene el primer modo de vibracion
set omega [expr pow($lambda,0.5)];
set alphaM 0.; # M-prop. amortiguamiento; D =
alphaM*M
set betaKcurr 0.; # K-proportional amortiguamiento;
+beatKcurr*KCurrent
set betaKcomm [expr 2.*$xDamp/($omega)]; # K-prop. parametro para
amortiguamiento; +betaKcomm*KlastCommitt
set betaKinit 0.; # initial-stiffness proportional amortiguamiento
+beatKinit*Kini
rayleigh $alphaM $betaKcurr $betaKinit $betaKcomm; # AMORTIGUAMIENTO DE
RAYLEIGH
set T1 [expr 2.0*$pi/$omega] ;# Periodo del primer modo de vibración
puts "T1: $T1 segundos"
```

Este análisis dinámico bidireccional es igual que el caso anterior, es decir aplica excitación uniforme al modelo, con la diferencia que como su nombre lo dice lo hace en dos direcciones, para ello se requieren dos archivos de registros sísmicos, el que será aplicado en la dirección X y el que se aplicará en la dirección Y. Para crear los patrones de carga que contengan estas aceleraciones se procede de igual manera que en el caso anterior.

```
# Parametros del registro sismico (afecta a todos los nudos de la base en ambas
direcciones)
# estos parametros son solamente para excitacion Uniforme

set sismox "sismox.txt" ; # sismo para la direccion X
set sismoy "sismoy.txt" ; # sismo para la direccion Y
set factorX 1.5; # se amplifica el sismo en un 50%
set factorY 0.5; # se reduce el sismo a un 50%
```

```

set TmaxAnalysis 10; # Tiempo de duracion maxima del analisis, el analisis durara 10
segundos
set dt 0.005;          # paso del tiempo para el analisis

# Se crean los patrones para la excitacion bidireccional
set serie1 "Series -dt $dt -filePath $sismox -factor [expr $factorX*$g]";# serie en
direccion X
set serie2 "Series -dt $dt -filePath $sismoy -factor [expr $factorY*$g]";# serie en
direccion Y
pattern UniformExcitation 3 1 -accel $serie1 ;      # patron para la excitacion en
direccion X
pattern UniformExcitation 4 2 -accel $serie2 ;      # patron para la excitacion en
direccion Y

# Se define el numero de pasos para el analisis
set Nsteps [expr int($TmaxAnalysis/$dt)];
set ok [analyze $Nsteps $dt];
# Proceso iterativo en caso de no convergencia
if {$ok != 0} {
    # change some analysis parameters to achieve convergence
    # performance is slower inside this loop
    # Time-controlled analysis
    set ok 0;
    set controlTime [getTime];
    while {$controlTime < $TmaxAnalysis && $ok == 0} {
        set controlTime [getTime]
        set ok [analyze 1 $dt]
        if {$ok != 0} {
            puts "Intentado con Newton with Initial Tangent .."
            test NormDispIncr $Tol 1000 0
            algorithm Newton -initial
            set ok [analyze 1 $dt]
            test EnergyIncr 1.e-8 1000 0
            algorithm ModifiedNewton
        }
        if {$ok != 0} {
            puts "Intentado con Broyden .."
            algorithm Broyden 8
            set ok [analyze 1 $dt]
            algorithm ModifiedNewton
        }
        if {$ok != 0} {
            puts "Intentado con NewtonWithLineSearch .."
            algorithm NewtonLineSearch .8
            set ok [analyze 1 $dt]
            algorithm ModifiedNewton
        }
    }
}

```

```
}; # end if ok !0
puts "Análisis Ground Motion finalizado. Tiempo del analisis: [getTime]"
```

Análisis Tipo 3

A continuación se presenta el código Tcl para el Analisis Dinamico bajo carga sismica en Múltiples apoyos, el cual está contenido dentro del archivo Eq_Multiple.tcl:

```
#####
#Example9. #
#Portico 2D. #
#PARAMETROS PARA ANALISIS DINAMICO BAJO EXCITACION #
SISMICA EN MULTIPLES APOYOS #
#####

# PARAMETROS PARA ANALISIS DINAMICO
constraints Transformation
numberer RCM
system BandGeneral
test EnergyIncr 1.e-8 10 0
algorithm ModifiedNewton
integrator Newmark 0.5 0.25
analysis Transient

# Parametros para amortiguamiento por Rayleigh-----
set xDamp 0.02; # amortiguamiento
set lambda [eigen 1]; # se obtiene el primer modo de vibracion
set omega [expr pow($lambda,0.5)];
set alphaM 0.; # M-prop. amortiguamiento; D =
alphaM*M
set betaKcurr 0.; # K-proportional amortiguamiento;
+beatKcurr*KCurrent
set betaKcomm [expr 2.*$xDamp/($omega)]; # K-prop. parametro para
amortiguamiento; +betaKcomm*KlastCommitt
set betaKinit 0.; # initial-stiffness proportional amortiguamiento
+beatKinit*Kini
rayleigh $alphaM $betaKcurr $betaKinit $betaKcomm; # AMORTIGUAMIENTO DE
RAYLEIGH
set T1 [expr 2.0*$pi/$omega] ;# Periodo del primer modo de vibración
puts "T1: $T1 segundos"
```

Como se trata de un análisis que aplica excitaciones distintas a cada apoyo, es necesario crear un tipo de patrón para excitación múltiple llamado **pattern MultipleSupport**, este

patrón permite asignar excitaciones prescritas similares o diferentes a ser introducidas en los diferentes apoyos de la estructura, su sintáxis es la siguiente:

pattern MultipleSupport \$patternTag { groundMotion... imposedMotion... ... }

Donde:

\$patternTag etiqueta unica para el patrón de carga
groundMotion... comando que genera la excitación
imposedMotion... .. comando que impone la excitación al apoyo.

Los resultados obtenidos en los nudos son los ABSOLUTOS y no los relativos como en el caso de la excitación uniforme “pattern UniformExcitation”.

El comando **groundMotion** se usa para crear la excitación que luego será impuesta a los apoyos de la estructura con el comando **imposedMotion** para crear el patrón MultipleSupport. Su sintaxis es la siguiente:

groundMotion type? arg1? ...

Los argumentos adicionales que se necesitan definir dependen del tipo de excitación que se cree. Existen 2 tipos:

Plain GroundMotion
Interpolated GroundMotion

El tipo plain se usa para construir una excitación plana. Cada excitación está asociada a una “serie de tiempo” que defien los registros de aceleración, velocidad y desplazamiento para la excitación. Su sintáxis es:

groundMotion \$gMotionTag Plain <-accel (accelSeriesType accelArgs)>
<-vel (velSeriesType velArgs)> <-disp (dispSeriesType dispArgs)> <-
int (IntegratorType intArgs)>

Donde:

\$gMotionTag	etiqueta única para el patrón de excitación
<-accel (accelSeriesType accelArgs)>	parámetros que definen la serie de registros de aceleración, opcional.
<-vel (velSeriesType velArgs)>	parámetros que definen la serie de registros de velocidad, opcional.
<-disp (dispSeriesType dispArgs)>	parámetros que definen la serie de registros de desplazamiento, opcional.
IntegratorType	se usa para generar el tipo de integrador numérico, por defecto trapezoidal.

Puede especificarse al mismo tiempo una combinación de los tres tipos, es decir de aceleración, velocidad y desplazamiento.

El comando **imposedMotion** se usa para exigir la respuesta de cierto grado de libertad de un nudo. Su sintáxis es la siguiente:

imposedMotion \$nodeTag \$dirn \$gMotionTag

Donde:

- \$nodeTag** etiqueta del nudo a restringir para aplicar la excitación
- \$dirn** grado de libertad de del que se desea la respuesta. El rango válido de valores es de 1 hasta el ndl del nudo
- \$gMotionTag** etiqueta del patrón de excitación predefinido “\$gmTag”

Para este ejemplo se ha decidido aplicar el archivo de aceleración sismox.txt con un factor de 1.3g al nudo 1 en dirección 1, y con un factor de 1.5g al nudo 2 en dirección 1.

```
# Parametros del registro sismico (cada registro afecta a un nudo diferente en la
direccion deseada)
# estos parametros son solamente para exitacion sismica Multiple
set sismox "sismox.txt"
set factorX 1.3; # se amplifica el sismo en un 30%
set factorY 1.5; # se amplifica el sismo a un 50%
set TmaxAnalysis 10; # Tiempo de duracion maxima del analisis, el analisis durara 10
segundos
set dt 0.005; # paso del tiempo para el analisis

# Se crea el patron para la exitacion multiple
pattern MultipleSupport 5 {
    set DispSeries1 "Series -dt $dt -filePath $sismox -factor [expr $g*$factorX]" ;
# serie para el nudo 1
    set DispSeries2 "Series -dt $dt -filePath $sismox -factor [expr $g*$factorY]" ;
# serie para el nudo 2
    groundMotion 6 Plain -accel $DispSeries1
    imposedMotion 1 1 6 ; # se aplica el registro sismico al nudo 1 en direccion
1=X
    groundMotion 7 Plain -accel $DispSeries2
    imposedMotion 2 1 7 ; # se aplica el registro sismico al nudo 2 en direccion
1=X
}

# Se define el numero de pasos para el analisis
set Nsteps [expr int($TmaxAnalysis/$dt)];
set ok [analyze $Nsteps $dt];

if {$ok != 0} { ; # analysis was not successful.
```

```

# -----
----
# change some analysis parameters to achieve convergence
# performance is slower inside this loop
# Time-controlled analysis
set ok 0;
set controlTime [getTime];
while {$controlTime < $TmaxAnalysis && $ok == 0} {
    set controlTime [getTime]
    set ok [analyze 1 $dt]
    if {$ok != 0} {
        puts "Intentado con Newton with Initial Tangent .."
        test NormDispIncr $Tol 1000 0
        algorithm Newton -initial
        set ok [analyze 1 $dt]
        test EnergyIncr 1.e-8 1000 0
        algorithm ModifiedNewton
    }
    if {$ok != 0} {
        puts "Intentado con Broyden .."
        algorithm Broyden 8
        set ok [analyze 1 $dt]
        algorithm ModifiedNewton
    }
    if {$ok != 0} {
        puts "Intentado con NewtonWithLineSearch .."
        algorithm NewtonLineSearch .8
        set ok [analyze 1 $dt]
        algorithm ModifiedNewton
    }
}
}; # end if ok !0
puts "Analysis Ground Motion finalizado. Tiempo del analisis: [getTime]"

```

Análisis Tipo 4

A continuación se presenta el código Tcl para el Analisis Dinamico bajo la excitación de una onda senoidal uniforme, la cual está contenida dentro del archivo Senoidal_Uniforme.tcl:

```

#####
#Example9. #
#Portico 2D. #
#PARAMETROS PARA ANALISIS DINAMICO BAJO LA EXCITACION #
#DE UNA ONDA SINUSOIDAL UNIFORME #

```

```
#####

# PARAMETROS PARA ANALISIS DINAMICO
constraints Transformation
numberer RCM
system BandGeneral
test EnergyIncr 1.e-8 10 0
algorithm ModifiedNewton
integrator Newmark 0.5 0.25
analysis Transient

# Parametros para amortiguamiento por Rayleigh-----
set xDamp 0.02; # amortiguamiento
set lambda [eigen 1]; # se obtiene el primer modo de vibracion
set omega [expr pow($lambda,0.5)];
set alphaM 0.; # M-prop. amortiguamiento; D =
alphaM*M
set betaKcurr 0.; # K-proportional amortiguamiento;
+beatKcurr*KCurrent
set betaKcomm [expr 2.*$xDamp/($omega)]; # K-prop. parametro para
amortiguamiento; +betaKcomm*KlastCommitt
set betaKinit 0.; # initial-stiffness proportional amortiguamiento
+beatKinit*Kini
rayleigh $alphaM $betaKcurr $betaKinit $betaKcomm; # AMORTIGUAMIENTO DE
RAYLEIGH
set T1 [expr 2.0*$pi/$omega] ;# Periodo del primer modo de vibración
puts "T1: $T1 segundos"
```

A diferencia de los análisis anteriores, este no necesita de un registro de aceleración, puesto que como se trata de una onda es necesario definir los parámetros para definirla, como son el período, la amplitud y la duración.

Al tratarse de una excitación uniforme se crea un patrón UniformExcitation como se realizó en el “Tipo 1”, con la diferencia que la serie contendrá los parámetros de la onda en vez de los del registro sísmico.

La sintaxis de la serie de tiempo para ingresar los parámetros de la onda es:

Sine \$tStart \$tFinish \$period <-shift \$shift> <-factor \$cFactor>

Donde:

\$tStart	Tiempo de inicio cuando el factor de carga será aplicado
\$tFinish	tiempo final de aplicación de la carga
\$period	period característico de la onda senoidal

\$shift fase (en radianes), es opcional, por defecto =0
\$cFactor factor de carga a aplicar, es opcional, por defecto =1

```
# Parametros para la excitacion de la onda sinusoidal uniforme (la onda sinusoidal
afecta a todos los nudos de la base)

set TmaxAnalysis 10; # Tiempo de duracion maxima del analisis, el analisis durara 10
segundos
set dt 0.01; # paso del tiempo para el analisis
set GMSineAccAmpl [expr 0.5*$g]; # amplitud de la onda sinusoidal (esto indica el
movimiento del apoyo, no el movimiento del nudo libre)
set TPeriodSine 0.35; # periodo de la onda sinusoidal
set DurationSine 3; # duracion de la onda sinusoidal
# estos parametros son solamente para la excitacion de una onda sinusoidal
set omegaSine [expr 2*$pi/$TPeriodSine];
set vel0 [expr $GMSineAccAmpl*(-1)/$omegaSine];
set Seno "Sine 0 $DurationSine $TPeriodSine -factor $GMSineAccAmpl "
# Se crea el patron para la excitacion uniforme
pattern UniformExcitation 8 1 -accel $Seno -vel0 $vel0

# Se define el numero de pasos para el analisis
set Nsteps [expr int($TmaxAnalysis/$dt)];
set ok [analyze $Nsteps $dt];

# Proceso iterativo en caso de no convergencia
if {$ok != 0} {
    # change some analysis parameters to achieve convergence
    # performance is slower inside this loop
    # Time-controlled analysis
    set ok 0;
    set controlTime [getTime];
    while {$controlTime < $TmaxAnalysis && $ok == 0} {
        set controlTime [getTime]
        set ok [analyze 1 $dt]
        if {$ok != 0} {
            puts "Intentado con Newton with Initial Tangent .."
            test NormDispIncr $Tol 1000 0
            algorithm Newton -initial
            set ok [analyze 1 $dt]
            test EnergyIncr 1.e-8 1000 0
            algorithm ModifiedNewton
        }
        if {$ok != 0} {
            puts "Intentado con Broyden .."
            algorithm Broyden 8
        }
    }
}
```

```

        set ok [analyze 1 $dt]
        algorithm ModifiedNewton
    }
    if {$ok != 0} {
        puts "Intentado con NewtonWithLineSearch .."
        algorithm NewtonLineSearch .8
        set ok [analyze 1 $dt]
        algorithm ModifiedNewton
    }
}
}; # end if ok !0
puts "Análisis Ground Motion finalizado. Tiempo del análisis: [getTime]"

```

Análisis Tipo 5

A continuación se presenta el código Tcl para el Análisis Dinámico bajo la excitación de una onda senoidal en múltiples apoyos, la cual está contenida dentro del archivo `Senoidal_Multiple.tcl`:

```

#####
#Example9. #
#Portico 2D. #
#PARAMETROS PARA ANALISIS DINAMICO BAJO LA EXCITACION #
#DE UNA ONDA SENOIDAL EN MULTIPLES APOYOS #
#####

# PARAMETROS PARA ANALISIS DINAMICO
constraints Transformation
numberer RCM
system BandGeneral
test EnergyIncr 1.e-8 10 0
algorithm ModifiedNewton
integrator Newmark 0.5 0.25
analysis Transient

# Parametros para amortiguamiento por Rayleigh-----
set xDamp 0.02; # amortiguamiento
set lambda [eigen 1]; # se obtiene el primer modo de vibracion
set omega [expr pow($lambda,0.5)];
set alphaM 0.; # M-prop. amortiguamiento; D =
alphaM*M
set betaKcurr 0.; # K-proporcional amortiguamiento;
+beatKcurr*KCurrent
set betaKcomm [expr 2.*$xDamp/($omega)]; # K-prop. parametro para
amortiguamiento; +betaKcomm*KlastCommitt

```

```

set betaKinit 0.; # initial-stiffness proportional amortiguamiento
+beatKinit*Kini
rayleigh $alphaM $betaKcurr $betaKinit $betaKcomm; # AMORTIGUAMIENTO DE
RAYLEIGH
set T1 [expr 2.0*$pi/$omega] ;# Periodo del primer modo de vibración
puts "T1: $T1 segundos"

```

En este caso como se trata de una excitación múltiple se necesario definir 2 ondas, para lo que procedemos de igual forma que en el “Tipo 4” a definir los parámetros de período, amplitud y duración de cada una de las ondas.

Los patrones de carga para este análisis se los crea de manera similar a como se creó para el análisis “Tipo 3”. En este caso las series contendrán las características de cada onda.

```

# Parametros para la excitacion de la onda senoidal multiple (cada onda afecta a un
nudo diferente en la direccion deseada)
# Funcion Senoidal-----
set TmaxAnalysis 10; # Tiempo de duracion maxima del analisis, el analisis durara 10
segundos
set dt 0.01; # paso del tiempo para el analisis

# Para el nudo 1
set GMSineDispAmpl1 0.1; # amplitud de la onda sinusoidal (esto indica el
movimiento del apoyo, no el movimiento del nudo libre)
set TPeriodSine1 0.35; # periodo de la onda sinusoidal
set DurationSine1 3; #duracion de la onda sinusoidal
# Para el nudo 2
set GMSineDispAmpl2 0.15; # amplitud de la onda sinusoidal (esto indica el
movimiento del apoyo, no el movimiento del nudo libre)
set TPeriodSine2 0.36; # periodo de la onda sinusoidal
set DurationSine2 3.1; # duracion de la onda sinusoidal

# estos parametros son solamente para excitacion sismica Multiple
# Se crean los los patrones para la excitacion multiple
pattern MultipleSupport 9 {
set DispSeries3 "Sine 0 $DurationSine1 $TPeriodSine1 -factor $GMSineDispAmpl1" ;
# serie para el nudo 1
set DispSeries4 "Sine 0 $DurationSine2 $TPeriodSine2 -factor $GMSineDispAmpl2" ;
# serie para el nudo 2
    groundMotion 10 Plain -disp $DispSeries3
    imposedMotion 1 1 10 ; # se aplica la onda senoidal al nudo 1 en direccion 1=X
    groundMotion 11 Plain -disp $DispSeries4
    imposedMotion 2 1 11 ; # se aplica la onda senoidal al nudo 2 en direccion 1=X
}

```

```

# Se define el numero de pasos para el analisis
set Nsteps [expr int($TmaxAnalysis/$dt)];
set ok [analyze $Nsteps $dt];
# Proceso iterativo en caso de no convergencia
if {$ok != 0} {
    # change some analysis parameters to achieve convergence
    # performance is slower inside this loop
    # Time-controlled analysis
    set ok 0;
    set controlTime [getTime];
    while {$controlTime < $TmaxAnalysis && $ok == 0} {
        set controlTime [getTime]
        set ok [analyze 1 $dt]
        if {$ok != 0} {
            puts "Intentado con Newton with Initial Tangent .."
            test NormDispIncr $Tol 1000 0
            algorithm Newton -initial
            set ok [analyze 1 $dt]
            test EnergyIncr 1.e-8 1000 0
            algorithm ModifiedNewton
        }
        if {$ok != 0} {
            puts "Intentado con Broyden .."
            algorithm Broyden 8
            set ok [analyze 1 $dt]
            algorithm ModifiedNewton
        }
        if {$ok != 0} {
            puts "Intentado con NewtonWithLineSearch .."
            algorithm NewtonLineSearch .8
            set ok [analyze 1 $dt]
            algorithm ModifiedNewton
        }
    }
}; # end if ok !0
puts "Análisis Ground Motion finalizado. Tiempo del análisis: [getTime]"

```

Análisis de Resultados

Una vez concluida la explicación del ejemplo y ejecutado el interpretador para cada uno de las combinaciones posibles, a continuación se muestran gráficas de la historia de desplazamiento para el nudo 3 considerando que el modelo fue construido con elementos inelásticos de secciones sin fibras.

- Elementos tipo 1 (elementos inelásticas con secciones sin fibras) y análisis tipo 1 (análisis dinámico Eq_Uniforme.tcl)

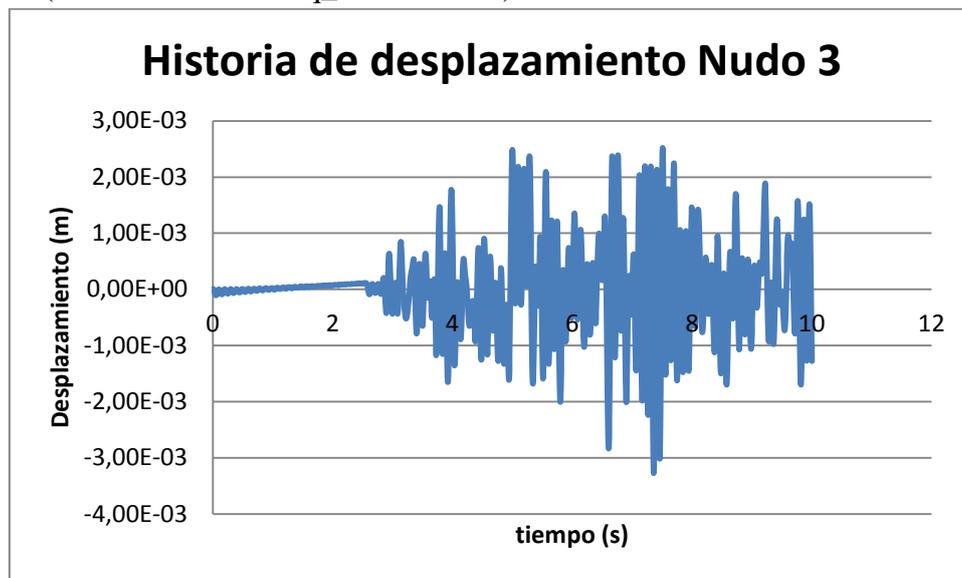


FIGURA 76 – Historia de desplazamiento en dirección X nudo 3, análisis 1

- Elementos tipo 1 (elementos inelásticas con secciones sin fibras) y análisis tipo 2 (análisis dinámico Eq_Bidireccional.tcl)

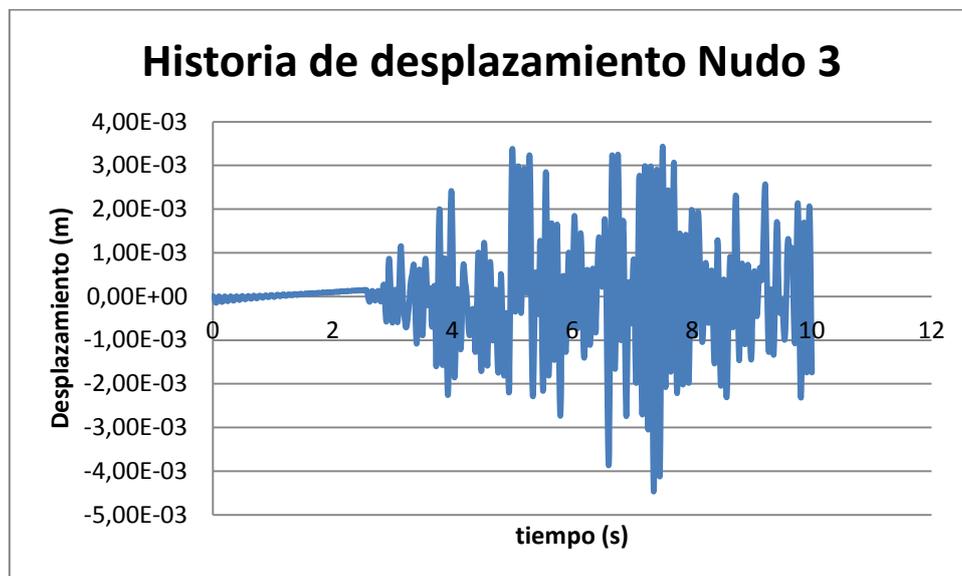


FIGURA 77 – Historia de desplazamiento en dirección X nudo 3, análisis 2

- Elementos tipo 1 (elementos inelásticas con secciones sin fibras) y análisis tipo 3 (análisis dinámico Eq_Múltiple.tcl)

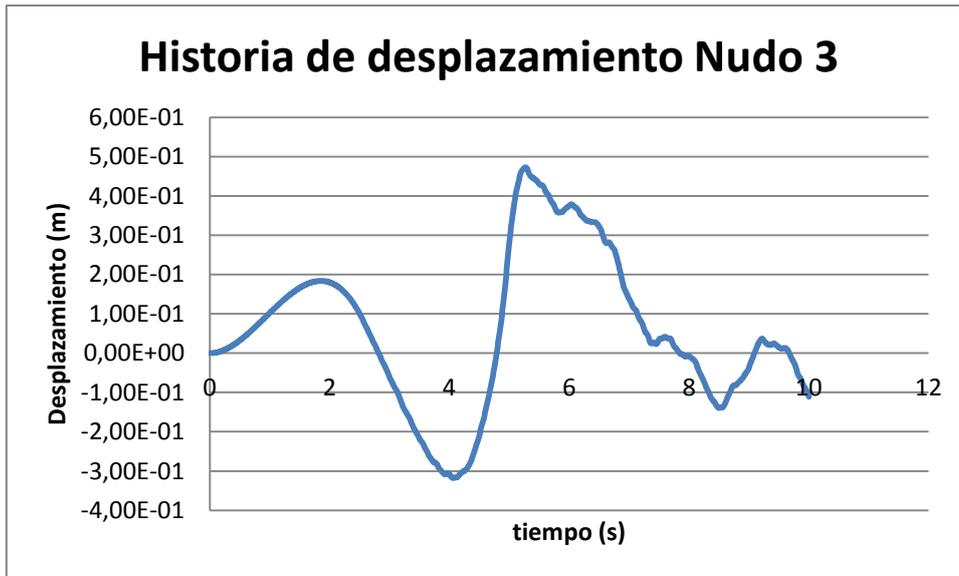


FIGURA 78 – Historia de desplazamiento en dirección X nudo 3, análisis 3

- Elementos tipo 2 (elementos inelásticas con secciones con fibras) y análisis tipo 4 (análisis dinámico Sinusoidal_Uniforme.tcl)

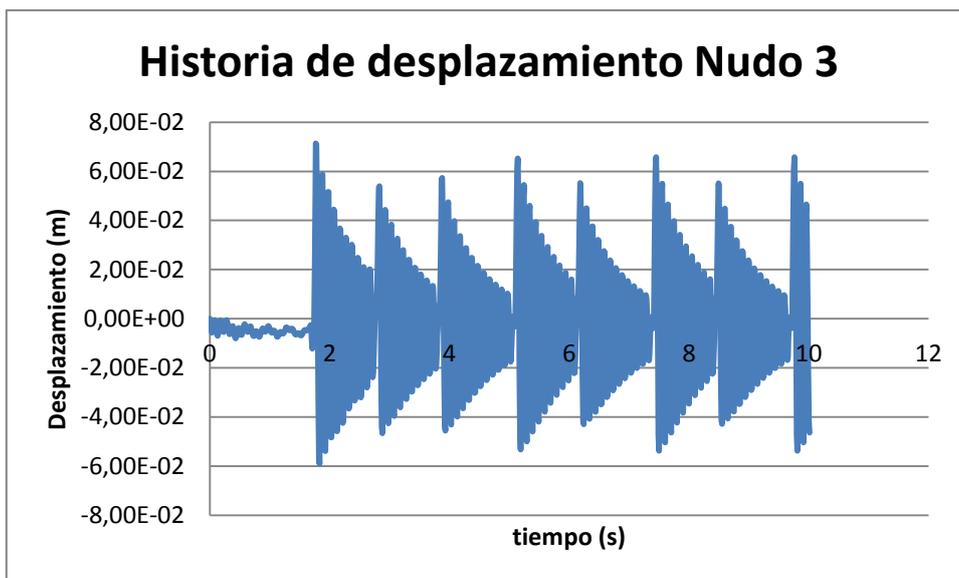


FIGURA 79 – Historia de desplazamiento en dirección X nudo 3, análisis 3

- Elementos tipo 2 (elementos inelásticas con secciones con fibras) y análisis tipo 4 (análisis dinámico Sinusoidal_Multiple.tcl)

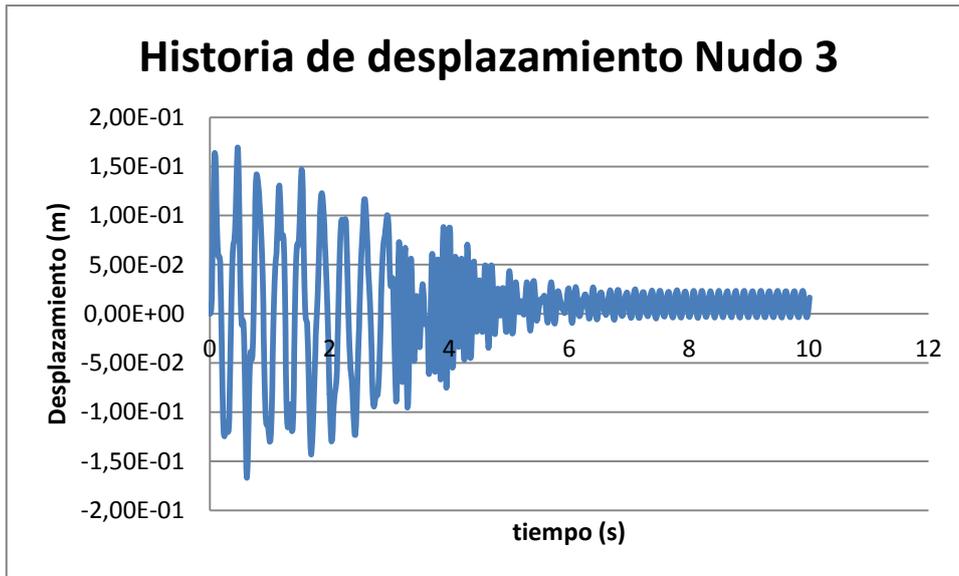


FIGURA 80 – Historia de desplazamiento en dirección X nudo 3, análisis 3

De igual forma se pueden obtener gráficas para el otro tipo de elementos, por ejemplo en la siguiente figura se muestra el similitud de las curvas de desplazamiento en el nudo libre 4 para los dos tipos de elementos cuando se ejecutó el análisis tipo 3.



FIGURA 81 – Comparación de desplazamientos entre elementos tipo 1 y tipo 2

Con el resto de archivos pueden obtenerse gráficas similares a las obtenidas en los ejemplos anteriores.

CAPÍTULO VIII

OPENSEES EN PARALELO

8. OPENSEES EN PARALELO

8.1.INTRODUCCIÓN

La programación en paralelo está incrementando de manera considerable para la simulación de terremotos reales sobre sistemas estructurales y geotécnicos. Con la llegada de procesadores multi-núcleos estos análisis se harán muy rápido y llegarán a ser el único medio disponible para estudiar de manera completa, real y en poco tiempo el comportamiento de las estructuras. En este capítulo se proveerá una descripción de las capacidades que tiene el intérprete de OpenSees en paralelo, por ejemplo como corren los computadores en paralelo, y los pasos de cómo crear los ejecutables de OpenSees en paralelo. Para poder entender este capítulo es necesario conocer como trabajan OpenSees y las computadoras en paralelo sobre la cual se intenta correr OpenSees.

8.2.COMPUTADORES EN PARALELO

Una computadora en paralelo es una máquina compuesta por más de un procesador que permite procesar la información entre ellos. Dicho de otro modo, una computadora en paralelo es una colección de procesadores que trabajan de manera conjunta para resolver problemas grandes en poco tiempo. Hay muchas maneras diferentes de formar máquinas que puedan trabajar en paralelo. Entre estos tenemos supercomputadoras con miles de

procesadores, redes locales con decenas o cientos de procesadores, e incluso computadores personales con un procesador multi-núcleo (pocos núcleos). De ahí que se pueda clasificar en:

- *Computadoras escalares.*- un solo procesador con “pipelining” es una técnica usada en microprocesadores avanzados a través de la cual el microprocesador comienza ejecutando una segunda instrucción antes que haya finalizado la primera, por ejemplo: pentium 4.
- *Mircoprocesadores de memoria compartida.*- procesadores que operan independientemente pero todos tienen acceso a la misma memoria. Los cambios que un procesador haga pueden ser vistos por el resto. El acceso a la memoria puede ser uniforme “UMA” o no uniforme “NUMA”.

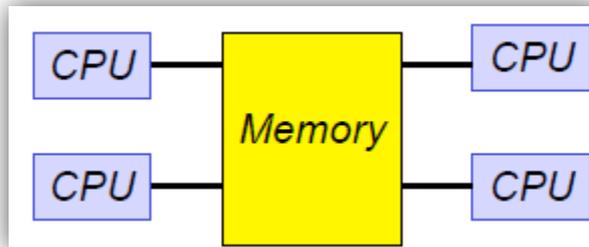


FIGURA 82 – Mircoprocesadores de memoria compartida

- De memoria distribuida:
 - o *Memoria distribuida MPPs (Massively Parallel System).*- Los procesadores operan independientemente y cada uno tiene su propia memoria. La datos deben ser compartidos a través de la comunicación de red “network”.

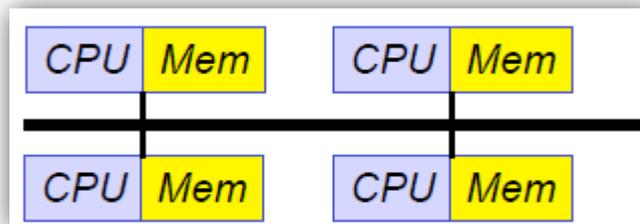


FIGURA 83 – Memoria distribuida MPPs

- o *Memoria distribuida SMPs – Hybrid Systems.*- Con los nuevos sistemas multi – núcleos la memoria se distribuye a los procesadores multi – núcleos.

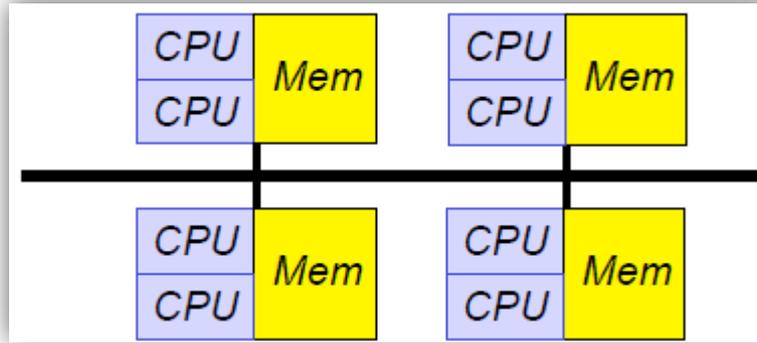


FIGURA 84 – Memoria distribuida SMPs – Hybrid Systems

- *Sistemas Cluster.*- La “network” enciende de manera muy rápida las máquinas para minimizar el tiempo de ejecución de los procesos.

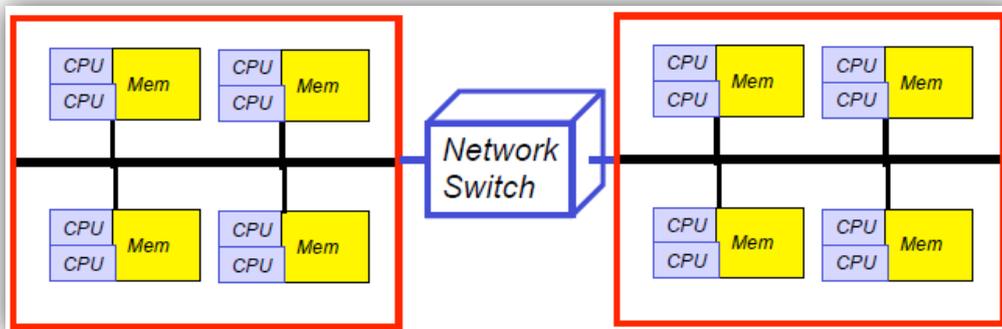


FIGURA 85 – Sistemas Cluster

8.3.MODELOS DE PROGRAMACIÓN EN PARALELO

Los modelos de programación en paralelo proveen un concepto abstracto de la estructura y de la forma que operación del sistema. Actualmente hay dos modelos dominantes para programar en paralelo:

- 1.- **Shared Memory “Memoria Compartida”:** La ejecución del programa es vista como una colección de procesos *continuos o enlazados* “threads” donde cada uno comparte una dirección virtual con los otros procesos y acceso a las variables compartidas. El acceso a los datos compartidos debe ser sincronizado para evitar que ocurran fallas entre los procesos (por ejemplo cuando dos o más procesos acceden por la misma ruta a la memoria).
- 2.- **Message Passing:** La ejecución del programa es vista como una colección de procesos *independientes*. Cada proceso se ejecuta por su lado, tiene su propia memoria, un identificador único y es capaz de comunicarse con el resto de procesos.

8.4.INTERPRETES DE OPENSEES EN PARALELO

Como ya se mencionó al inicio de este trabajo OpenSees es una plataforma que se usa para simular de manera realista sistemas estructurales y geotécnicos analizando el comportamiento no-lineal de los elementos. Al inicio la plataforma fue escrita en C++ y brindaba la posibilidad de crear aplicaciones con elementos finitos que podían ser ejecutadas tanto secuencialmente como sobre computadoras en paralelo.

Así, para trabajar sobre computadores en paralelo se han creado dos tipos de intérpretes, dependiendo de la magnitud del modelo:

- **OpenSeesSP:** intérprete se usa para analizar modelos de gran tamaño, además es creado para usuarios con poco conocimiento sobre la computación en paralelo y para quienes tienen archivos de entrada que son muy grandes o también cuando el tiempo que toma en correr el análisis secuencial es muy largo. No hay comandos especiales para el procesamiento en paralelo, sin embargo hay opciones adicionales en el momento de la elección de las respuestas. De ahí que el nombre sea “Single Parallel Interpreter”
- **OpenSeesMP:** intérprete para análisis de modelos de tamaño pequeño o moderado, por ejemplo estudios paramétricos, o para correr códigos más complicados los cuales se especifican subroutines y ecuaciones de solución en paralelo. Cuando se ejecuta sobre máquinas en paralelo, cada procesador corre con el mismo intérprete con los mismos códigos de ingreso. El usuario tiene el control tanto en el nivel de las líneas de comando como en el nivel de código para especificar trabajo que cada procesador va a realizar. De ahí que el nombre sea “Multiple Parallel Interpreter's”

En el presente trabajo solamente se estudiará OpenSeesSP.

8.5.OPENSEES SP

Como ya se mencionó antes este intérprete es creado para análisis de modelos muy grandes, teniendo en cuenta dos objetivos en mente: el uno minimizar los cambios en el archivo de entrada (en lo posible el mismo tcl de OpenSees.exe); y el segundo minimizar los conocimientos requeridos para procesos en paralelo.

8.5.1. ¿Qué ocurre cuando se ejecuta OpenSeesSP?

Cuando se ejecuta un modelo sobre una computadora en paralelo, lo primero se ejecuta el intérprete sobre un solo procesador, “P0”, quien empieza a interpretar y procesar los

comandos del archivo de entrada de igual forma como están ingresados en el allí. Mientras esto ocurre el resto de procesadores (P1, P2, Pn-1) se encuentran estáticos esperando recibir las instrucciones. El modelo es construido en base a comandos que van creando paso a paso (los nudos, los elementos, las cargas), todo se construye en el dominio, es decir en “P0”. Como se muestra en la FIGURA 86.

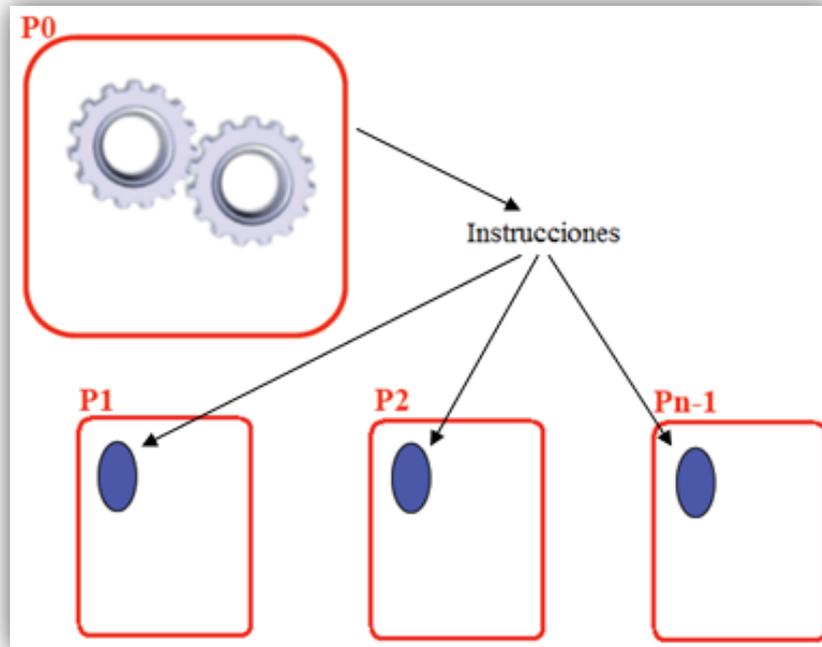


FIGURA 86 – Interprete OpenSeesSP antes del análisis

Una vez que llega el momento de ejecutar el análisis el modelo es fraccionado, es decir los elementos son divididos y distribuidos entre todas las “n-1” máquinas. Después de esto la solución del sistema de ecuaciones puede ser hecha en paralelo, como se muestra en la FIGURA 87.

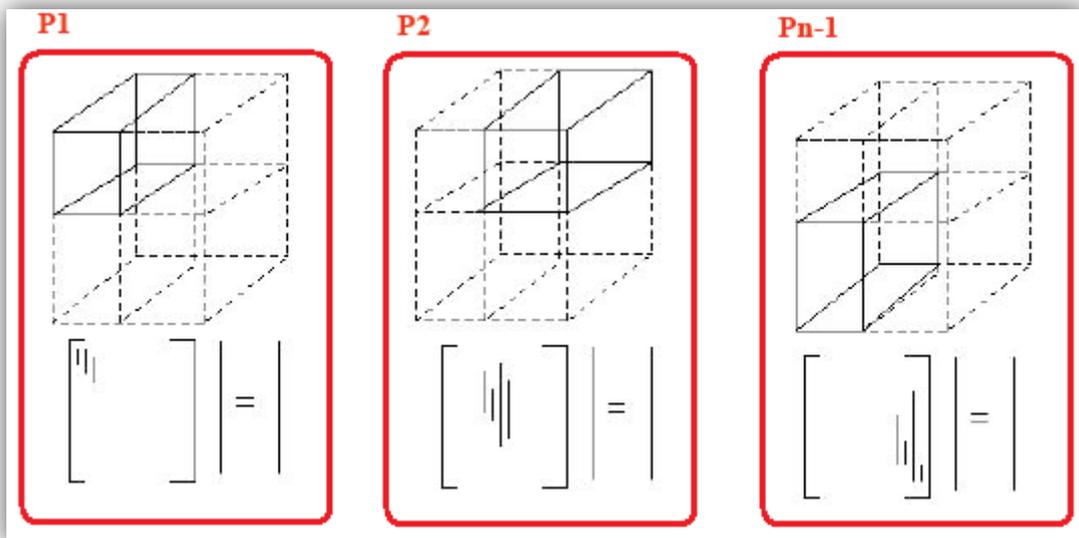


FIGURA 87 – Interprete OpenSeesSP después del análisis

8.6.COMANDOS MODIFICADOS

Para que la solución del sistema de ecuaciones lineales se encuentre, depende de la ecuación de solución que se escogió, para OpenseesSP se debe modificar el comando “System” para conseguir resolver las ecuaciones en paralelo:

- System mumps
- System diagonal

Adicionalmente los archivos de salida generados por los recorders son diferentes. El orden en que se presentan los resultados en las filas no es necesariamente similar a los que se presentan cuando se ejecuta un análisis secuencial (OpenSees.exe). Por ello, si recordando la sintáxis del comando recorder:

```
recorder Node <-file $fileName> <-xml $fileName> <-time> <-node ($node1  
$node2 ...)> <-nodeRange $startNode $endNode> <-region  
$RegionTag> <-node all> -dof ($dof1 $dof2 ...) $respType
```

Cuando se ejecuta un análisis secuencial (OpenSees.exe) en el recorder se define o usa la opción “-file” y se guarda la información de más de un elemento o nudo tal como el usuario lo indique, por el contrario cuando se ejecuta un análisis en paralelo se debe cambiar la opción a “-xml” en lugar de “-file”, ya que permite etiquetar cada columna de resultados del archivo de datos (-file y -xml no pueden especificarse simultáneamente).

8.7.PASOS PARA LA EJECUCIÓN EN PARALELO

Hay algunos pasos que se deben seguir para obtener y construir la versión en paralelo del intérprete de OpenSees. A continuación se detallan a breves rasgos de estos pasos asumiendo que el usuario está trabajando sobre una máquina Unix.

1.- Obtener de Tcl/Tk

Se puede encontrar en: <http://www.tcl.tk/software/tcltk/>, al igual que instrucciones para su instalación.

2.- Obtener OpenSees

La última versión estable de OpenSees se puede obtener desde el sitio web: <http://opensees.berkeley.edu/OpenSees/developer/download.php>

3.- Creación del archivo del modelo

El paso más complicado es la creación del archivo que contiene las características del modelo que se va a analizar, así mismo es donde se asignan las tareas a cada proceso y se identifica las que se pueden ejecutarse al mismo tiempo. Una de las variables que necesariamente se debe establecer es el modo de programación “PROGRAMMING_MODE”. Este valor depende de sobre cual interprete se va a trabajar.

4.- Para OpenSees SP

PROGRAMMING_MODE=PARALLEL

5.- Para OpenSeesMP

PROGRAMMING_MODE=PARALLEL_INTERPRETERS

6.- Creación de los directorios y librerías

7.- Construcción del ejecutable .exe

Algo que hay que recordar para obtener buenos resultados cuando se trabaja en paralelo es el tamaño de las tareas asignadas, ya que dependen mucho de la capacidad que tenga la computadora, además que la solución obtenida en una máquina puede no ser la misma que se obtenga en otra igualmente en paralelo. Es por ello que la programación en paralelo si bien es cierto ahorra tiempo pero es más difícil programarla que si se la hace secuencialmente.

CONCLUSIONES:

- Se comprobó que la mayor fortaleza de OpenSees radica en su capacidad para simular el comportamiento real de los materiales, secciones y elementos que componen las estructuras, mediante la comparación de los resultados con otros software.
- Se demostró que OpenSees es actualmente la herramienta más precisa para la simulación en ingeniería sísmica, sin embargo la herramienta por sí misma no soluciona ningún problema.
- OpenSees es muy flexible y permite que un mismo problema se resuelva de distintas maneras. Por ello para lograr un mayor realismo en la simulación dependerá exclusivamente de las características que se de al modelo, así como de las opciones seleccionadas para el análisis.
- Es preciso seguir los pasos para la generación del código Tcl para evitar errores, ya que solamente los de sintaxis son encontrados por el interpretador Tcl, y nuestras equivocaciones en la definición de unidades, propiedades de materiales, conectividad de elementos, orientación de ejes locales, asignación de cargas, opciones de análisis, y otros que puedan producirse durante el procesamiento de resultados pueden pasar inadvertidos por el programa.
- El análisis no lineal permite reproducir de forma cierta el comportamiento estructural del hormigón, ya que considera la no linealidad en las relaciones tensión – deformación de los elementos hormigón y acero, y también la imperfecta adherencia entre estos materiales. Constituyendo el método de los elementos finitos la herramienta más confiable para la representación de este comportamiento.

RECOMENDACIONES:

- El hecho mismo que OpenSees sea creado bajo una plataforma de código abierto hace fácil y casi inevitablemente se cometan errores durante la generación del código Tcl, por ello es recomendable seguir los pasos que se detallaron para la creación del modelo.
- Obtener las frecuencias de cada uno de los modos de vibración, frecuencias muy altas indican que el modelo está demasiado rígido o con poco masa, frecuencias muy bajas indican lo contrario. Esta comprobación es muy importante y básica.
- Comprobar el equilibrio, es decir la suma de las reacciones verticales en la base de un pórtico debe igualar a la suma de cargas gravitacionales aplicadas, si esto no se comprueba, existe un error en la asignación de cargas.
- Obtener la respuesta momento curvatura de las secciones con fibras, ya que es muy fácil cometer errores en la definición de secciones con fibras de distintos materiales. Estos errores son difíciles de advertir a partir de los resultados del análisis de toda la estructura, sin embargo, pueden ser fácilmente advertidos en un análisis momento curvatura de la sección, y porque además los valores de momento, curvatura de fluencia y la forma misma de la curva momento curvatura pueden ser fácilmente corroborados con otras herramientas o con cálculos manuales.
- Ejecutar un análisis pushover antes de un análisis no lineal de historia en el tiempo. El análisis pushover al igual que el análisis momento curvatura son muy prácticos a la hora de comprobar el modelo. Los valores de cortante basal de fluencia, el desplazamiento de fluencia y la forma misma de la curva pushover pueden ser corroborados fácilmente con otras herramientas o inclusive con cálculos manuales.
- Ir de sencillo a complejo, resulta una buena práctica comenzar con un modelo sencillo, fácil de comprobar, que resulte en resultados aproximados que sirva como referencia para la evaluación de resultados de modelos más complejos.
- Visitar constantemente la página web de OpenSees para obtener versiones actualizadas del programa y de su manual de usuario, ya que es una herramienta de continuo desarrollo.

BIBLIOGRAFÍA:

Álvarez Bencomo, I. C. (s.f.). Análisis del Comportamiento no Lineal del Hormigón en Vigas Isostáticas en Puentes. *Paper* , 2-4.

CÁCERES CHOMALI, J. P. (2001). MODELACIÓN, ANÁLISIS NO-LINEAL Y DISIPACIÓN DE ENERGÍA DE ESTRUCTURAS PLANAS SOMETIDAS A TERREMOTOS. *Tesis* .

Dobson, R. (2003). An Overview of P-Delta Analysis. 2.

Eads, L. (2010). *Pushover Analysis of 2-Story Moment Frame*. Recuperado el 30 de Julio de 2010, de http://opensees.berkeley.edu/wiki/index.php/Pushover_Analysis_of_2-Story_Moment_Frame

Galant, A. D. (Julio de 2005). Dynamic Analysis of RC Bridge Piers Using OpenSees. *Paper* , 1-3.

Habibullah, A., & Pyle, S. (1998). Practical Three Dimensional Nonlinear Static Pushover Analysis. *Structure Magazine* .

Mazzoni, S., & McKenna, F. (2010). *OpenSeesWiki*. Recuperado el 18 de Agosto de 2010, de http://opensees.berkeley.edu/wiki/index.php/Examples_Manual

Mazzoni, S., McKenna, F., Scott, M. H., & Fenves, G. L. (2007). *OpenSees Command Language Manual*. California, Esdos Unidos.

McKenna, F. (2008). OpenSeesSP. *OpenSees Parallel Workshop* .

McKenna, F. (2008). Parallel Computing. *OpenSees Parallel Workshop* .

McKenna, F., & Fenves, G. L. (2008). Using the OpenSees Interpreter on Parallel Computers. 2-5, 14-16.

Suarez, V. A., & Kowalsky, M. J. (2008). Manual ITHA-Bridge.

Suarez, V., Hurtado, J., & Ayala, A. (2008). *Laboratorio Virtual de Ingeniería Sísmica*. Recuperado el Enero, Febrero de 2010, de <http://eva.utpl.edu.ec/vlee/>