

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA MECÁNICA

**ANÁLISIS DEL MECANISMO BIELA-MANIVELA MEDIANTE EL
MÉTODO DE BOND GRAPH**

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO
MECÁNICO**

DIEGO ALEJANDRO MORÁN ESPINEL

diegomorane@hotmail.es

DIRECTOR: WILLIAM VENEGAS, Ing.

william.venegas@epn.edu.ec

CO-DIRECTOR: MARIO GRANJA, Ing.

mario.granja@epn.edu.ec

Quito, Septiembre del 2023

DECLARACIÓN

Yo Diego Alejandro Morán Espinel, declaro que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

La Escuela Politécnica Nacional, puede hacer uso de los derechos correspondientes a este trabajo, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Diego Alejandro Morán Espinel

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Diego Alejandro Morán Espinel bajo nuestra supervisión.

Dr. William Venegas

DIRECTOR DE PROYECTO

Ing. MARIO GRANJA

CODIRECTOR DE PROYECTO

AGRADECIMIENTO

A mi familia, por su confianza y apoyo, pues en todo momento han estado junto a mí, dándome ánimo.

A mis amigos, por acompañarme todo momento, porque a pesar de todo siempre han estado ahí acompañándome, dándome una mano.

Al Dr. William Venegas e Ing. Mario Granja por su apoyo y guía para el desarrollo del presente proyecto.

Diego Morán E.

DEDICATORIA

A mis padres, para ellos es la satisfacción de la culminación de mis estudios, de ellos es la euforia, de que su hijo ha logrado sus metas.

Diego Morán E.

Contenido

ESCUELA POLITÉCNICA NACIONAL.....	i
DECLARACIÓN.....	ii
CERTIFICACIÓN.....	iii
AGRADECIMIENTO.....	iv
DEDICATORIA.....	v
Contenido.....	vi
Índice de Figuras.....	ix
Índice de Tablas.....	xii
Resumen.....	xiii
Presentación.....	xiv
Capítulo 1.....	1
Introducción.....	1
1.1 Justificación.....	2
1.2 Objetivos.....	2
1.2.1 Objetivo General.....	2
1.2.2 Objetivos Específicos.....	3
1.3 Generalidades.....	3
Introducción a los mecanismos y a la cinemática.....	3
1.3.1 Cinética y cinemática.....	3
1.3.2. Máquinas y Mecanismos.....	4
1.3.3. Análisis y síntesis.....	4
1.3.4. Terminología de mecanismos.....	5
1.3.5. Tipos de mecanismos: Mecanismos planos, espaciales.....	6

1.3.6. Movilidad	6
1.4. Método de BOND GRAPH	7
1.4.1. Introducción	7
1.4.2. Elementos de Bond Graph	10
Capítulo 2.	25
Mecanismo de biela manivela	25
2.2 Metodología de la construcción de un Bond Graph	25
2.2.1 Dibujo del Bond Graph	28
2.2.2 Planteamiento del sistema de ecuaciones	41
2.2.3 Resolución de la ecuación de estado	51
Capítulo 3.	53
Comparación del Método de Bond Graph con otros métodos de resolución	53
3.1 Métodos alternativos de resolución	53
3.2 Comparación de Métodos	67
Capítulo 4.	69
Desarrollo del algoritmo en Software Computacional	69
4.1 Lenguaje de programación en PYTHON	69
4.2 Desarrollo del algoritmo	70
4.3 Simulación del modelo	74
4.4 Algoritmo genérico para resolución mecanismos	74
Capítulo 5.	81
Análisis y Resultados	81
5.1 Resultados	81
5.2 Análisis y Comparación de resultados	84
Capítulo 6.	94

Conclusiones y Recomendaciones.....	94
6.1 Conclusiones	94
6.2 Recomendaciones	95
Referencias Bibliográficas	96
Anexos	98
Anexo 1	98
Simplificación de Modelos de Bond Graph [20-sim referencia manual]	98
Anexo 2.....	100
Instalación de Python y librerías matplotlib, numpy, y otras.....	100

Índice de Figuras

Figura 1 Bond Graph correspondiente a un sistema	8
Figura 2 Circuito magnético dentro de un motor de torque de una válvula electrohidráulica.	9
Figura 3 Tetraedro de estado [7]	12
Figura 4 Elementos resistores 1-puerto a) resistor eléctrico, b) amortiguador mecánico, c) flujo a través de un tubo [7]	15
Figura 5 Representación elemento resistor 1-puerto [7]	15
Figura 6 Representación elemento capacitor 1-puerto [7]	16
Figura 7 Elementos capacitores 1-puerto a) condensador eléctrico, b) resorte mecánico, c) fluido almacenado, d) resorte torsional [7]	16
Figura 8 Representación elemento inductor 1-puerto [7]	17
Figura 9 Elementos inductivos 1-puerto a) inductor eléctrico, b) masa mecánico, c) inercia rotacional, d) fluido inercial [7]	18
Figura 10 Tetraedro de estado mostrando las variables generalizadas que están en la relación constitutiva para cada uno de los elementos 1- puerto [7]	18
Figura 11 Representación de una fuente de esfuerzo	19
Figura 12 Representación de una fuente de flujo	20
Figura 13 Representación de un elemento transformador	20
Figura 14 Representación esquemática de elementos transformadores en diferentes dominios a) transformador eléctrico, b) transformador mecánico (palanca), c) transformador hidráulico (pistón)	21
Figura 15 Representación de un elemento girador	22
Figura 16 Representación de un nodo de unión tipo 0	23
Figura 17 Representación de un nodo de unión tipo 1	24
Figura 18 Representación de un mecanismo biela manivela.	25

Figura 19 Representación Física de un mecanismo Biela manivela.	32
Figura 20 Representación de nodos tipo 1 correspondiente a las velocidades.	33
Figura 21 Insertando elementos tipo fuerza.	33
Figura 22 Insertando elementos capacitores y resistores.	34
Figura 23 Insertando elementos Inercia.	34
Figura 24 Insertando elementos transformadores.	35
Figura 25 Asignación de dirección de Elementos.	35
Figura 26 Eliminación de nodos tipo 1.	36
Figura 27 Elementos a Simplificar del Bond Graph.	36
Figura 28 Simplificación del Bond Graph.	36
Figura 29 Análisis de causalidad.	37
Figura 30 Relación entre elementos Físicos y Bond Graph.	37
Figura 31 Variables de un Elemento [7 pg 391]	39
Figura 32 Análisis de velocidades en ejes separados	39
Figura 33 Modelo de bond Graph para un elemento con movimiento libre [7 pg 400] .	40
Figura 34 Modelo de Bond Graph para un mecanismo biela manivela .	41
Figura 35 Modelo de Bond Graph para un mecanismo biela manivela con resistencia.	41
Figura 36 Numeración de Bonds.	42
Figura 37 Numeración de Bonds.	47
Figura 38 Simplificación de Bonds.	48
Figura 39 Simplificación y Numeración de Bonds.	48
Figura 40 Contrucción de triángulos.	54
Figura 41 Cinemática de la corredera.	57
Figura 42 Esquema biela-manivela.	59

Figura 43 Cinemática de la Manivela.	65
Figura 44 Cinemática de la Corredera.	65
Figura 45 Cinemática de la Manivela.	66
Figura 46 Cinemática de la Corredera.	66
Figura 47 Movimiento descrito por Bond Graph.	73
Figura 48 Diagrama de Flujo del código de Python.	74
Figura 49 Bond Graph genérico.	75
Figura 50 Numeración del Bond Graph.	76
Figura 51 Movimiento descrito por Bond Graph.	80
Figura 52 Análisis comparativo Desplazamiento de la corredera.	81
Figura 53 Análisis comparativo Velocidad de la corredera.	82
Figura 54 Análisis comparativo Desplazamiento de la corredera.	83
Figura 55 Análisis comparativo Velocidad de la corredera.	83
Figura 56 Análisis comparativo Velocidad de la corredera.	84

Índice de Tablas

Tabla 1 Variables generalizadas [4]	13
Tabla 2 Variables de potencia y Energía para sistemas mecánicos traslacionales [8]	26
Tabla 3 Variables de potencia y Energía para sistemas mecánicos Rotacional [8]..	26
Tabla 4 Relaciones constitutivas para elementos básicos [8]	27
Tabla 5 Representación de Elementos Mecánicos Lineales en Bond Graph	30
Tabla 6 Representación de Elementos Mecánicos Rotacional en Bond Graph.....	31
Tabla 7 Sistema de ecuaciones para el mecanismo	45
Tabla 8 Sistema de ecuaciones para el mecanismo	49
Tabla 9 Sistema de ecuaciones para el mecanismo	76

Resumen

El presente trabajo de titulación está compuesto de seis capítulos que contienen el desarrollo del proyecto del análisis del mecanismo biela manivela mediante el método de Bond Graph, cuyo objetivo es desarrollar una metodología para el desarrollo de mecanismos.

En el capítulo 1 se presenta una breve introducción de donde apareció la técnica de Bond Graph, seguido de definiciones correspondientes a mecanismos indispensables que se coaligan en este método, además se describe elementos y variables que aparecen en la metodología de Bond Graph.

En el capítulo 2 se describe brevemente un mecanismo biela manivela, luego de éste, se analiza la metodología para la resolución de problemas mecánicos con este método, y al final se analiza el mecanismo de biela manivela.

En el capítulo 3 se presentan métodos tradicionales de resolución de mecanismos, que luego se comparan con los el método de Bond Graph.

En el capítulo 4 se desarrolla un algoritmo en lenguaje de programación Python para la resolución de las ecuaciones diferenciales obtenidas en el capítulo 3 resultantes del modelo de Bond Graph, seguido a esto se crea una metodología genérica para resolver mecanismos que posean un grado de libertad.

En el capítulo 5 se realiza una comparación entre los datos obtenidos del modelo propuesto y el método analítico por triángulos, dando validez al método.

El capítulo 6 presenta las conclusiones y recomendaciones realizadas para el presente trabajo.

Presentación

En este trabajo se presenta un Análisis del Mecanismo Biela Manivela mediante el método de Bond Graph. Este método en los últimos años ha tenido un gran desarrollo por ser relativamente nuevo, una de las características del método es que permiten analizar mecanismos.

El modelamiento por BOND GRAPH (Gráficos de Enlaces), se realiza analizando la energía que entra y sale en un determinado elemento o sistema, que a su vez define Variables de Potencia y Variables de Energía y crea una ecuación o sistema de ecuaciones diferenciales a resolver para determinar la ecuación de estado del sistema.

El mecanismo de Biela Manivela es uno de los más utilizados dentro de las aplicaciones mecánicas, se puede describir como un mecanismo plano, con un grado de libertad que intercambia la energía rotacional de un elemento llamado manivela a otro llamado corredera, mediante un elemento llamado biela. Mediante el método de Bond Graph, se realiza un modelo de un mecanismo Biela Manivela, para lo cual se comienza describiendo paso a paso la formación del Bond Graph y el posterior planteamiento y resolución del sistema de ecuaciones.

A partir del planteamiento del Bond Graph de un mecanismo Biela Manivela, se extrapola este conocimiento y se crea un algoritmo genérico que posea las mismas características del mecanismo descrito.

Capítulo 1.

Introducción

La idea de BOND GRAPH fue desarrollada por el profesor Henry Paynter, y fue formalmente introducido como parte del pensum de estudio en el instituto tecnológico de Massachusetts en el año de 1959 luego de lo cual sus estudiantes: el profesor Karnopp, el profesor Margolis y el profesor Rosenberg elaboraron el concepto dentro de la metodología para el modelamiento de sistemas físicos como hoy se conoce. [5]

A lo largo de los años siguientes a la formulación del método, solo sus estudiantes tenían conocimiento de este y solo eran tratados como temas de clase, para los años ochenta y noventa el método fue incluido en proyectos de investigación de grandes empresas que pusieron a prueba al método, siendo uno de los primeros usuarios la NASA, a finales de los años ochenta y durante todos los años siguientes se ha inundado el mundo con conferencias que dieron a conocer este método [5].

En los últimos años el método se ha convertido una pieza importante en el análisis de sistemas de control orientado principalmente a la parte automotriz logrando un gran desarrollo en el tema de controladores y actuadores.

El análisis de mecanismos constituye un paso fundamental en el trabajo de ingeniería mecánica, y su aplicación en el diseño de máquinas, se ha convertido en el pilar del desarrollo de la industria.

Durante el proceso de diseño de máquinas, es muy importante analizar su comportamiento dinámico, esto permite predecir el comportamiento espacial, principalmente su movimiento y las fuerzas que son resultado de este.

A lo largo de la historia el ser humano ha venido desarrollando una gran cantidad de métodos que permiten analizar los movimientos de las máquinas que se han ido creando, así también métodos novedosos se han creado, y han remplazado a métodos anteriores, en esta proyecto se analiza un método relativamente nuevo que entre muchas otras cosas puede ser utilizado para analizar mecanismos.

Entre los métodos novedosos para el modelamiento de sistemas dinámicos sobresale el denominado Bond Graph (Gráfico de enlaces), que partiendo de fundamentos físicos sencillos y una metodología de planteamientos del sistema de ecuaciones diferenciales de primer orden, y basados en una estructura gráfica permite sistematizar la escritura de los modelos y su solución.

Una de las principales características del método de Bond Graph es la posibilidad de analizar un sistema por partes o submodelos. Los que pueden ser analizados independientemente o ensamblados para formar un sistema global y dar como resultado un modelo final.

1.1 Justificación

En el presente proyecto de “Análisis del mecanismo biela-manivela mediante el método de Bond Graph”, busca analizar al método poco común de modelación en un mecanismo común a fin de comparar este método con otros métodos tradicionales, este método fue desarrollado hace poco tiempo, pero ha tenido un gran desarrollo en el estudio de sistemas de control, no así en el área del estudio de mecanismos.

El método de Bond Graph es un método relativamente nuevo desarrollado en los años 60' que simplifica la interpretación del mundo físico para crear ecuaciones diferenciales de primer orden de fácil interpretación y un procesamiento rápido.

El mecanismo de biela manivela es uno de los más estudiados y al realizar una comparación entre los métodos tradicionales y el de Bond Graph permitirá tener una mayor comprensión de las ventajas y desventajas al usar esta metodología.

1.2 Objetivos

1.2.1 Objetivo General

Analizar el mecanismo biela-manivela mediante el método de BOND GRAPH.

1.2.2 Objetivos Específicos

- Desarrollar una metodología de los GRAPHOS para el desarrollo de mecanismos.
- Desarrollar un modelo de BOND GRAPH Para el mecanismo de biela manivela.
- Comparar el método de BOND GRAPH con otros métodos tradicionales.
- Desarrollar un algoritmo genérico en PYTHON para resolución de mecanismos.

1.3 Generalidades

Introducción a los mecanismos y a la cinemática

1.3.1 Cinética y cinemática

Hablar de cinética y cinemática es hablar del movimiento de objetos, los dos términos tratan el comportamiento espacial de estos objetos a lo largo del tiempo. Estos términos en la vida práctica no se pueden separar pero por cuestiones didácticas se las estudia por separado.

La cinemática estudia la geometría del movimiento de los cuerpos determinando la posición, desplazamiento, rotación, rapidez, velocidad y aceleración de un mecanismo sin la consideración de las fuerzas que generan estos movimientos. [1] [2]

A diferencia de la cinemática, la cinética estudia las fuerzas que generan este movimiento. [1]

Por lo general para el diseño de un sistema mecánico se requiere conocer ambos temas, conocer uno para calcular el otro, esto principalmente se debe a la segunda ley de Newton $F=m.a$, de donde se conoce las fuerzas aplicadas para llegar a determinar las aceleraciones o viceversa.

1.3.2. Máquinas y Mecanismos

La palabra máquina se ha utilizado a lo largo del tiempo para referirse a un dispositivo que transforma, modifica y dirige fuerzas, acompañadas de sus respectivas interacciones de movimiento, para cumplir un determinado objetivo [2].

Según Reuleaux F. “una máquina es una combinación de cuerpos resistentes de tal manera que, por medio de ellos, las fuerzas mecánicas de la naturaleza se puedan encausar para realizar un trabajo acompañado de movimientos determinados” [3 Pag5].

A un mecanismo se lo define como una parte mecánica de una máquina, cuya función es transmitir movimiento y/o fuerza de un lugar llamado fuente hacia otro lugar llamado salida, por lo que se puede decir que el mecanismo de una máquina son las partes rígidas que están configuradas y conectadas de modo que generen un movimiento [2].

De igual modo Reuleaux F. define a un mecanismo como una “combinación de cuerpos resistentes conectados por medio de articulaciones móviles para formar una cadena cinemática cerrada con un eslabón fijo, y cuyo propósito es transformar el movimiento [3 Pag5].

1.3.3. Análisis y síntesis

La palabra síntesis corresponde al término diseño, en otras palabras, cuando se habla de síntesis se hace mención al diseño de un sistema mecánico, esto consiste en idear un patrón o método para lograr un determinado propósito a través de establecer tamaños, formas, materiales y disposición de las piezas para que la máquina obtenida realice lo deseado. [3]

La palabra análisis denota la técnica que emplea el diseñador para examinar en forma crítica un diseño existente o propuesto con el objetivo de ver si es adecuado para el trabajo que va a realizar, en otras palabras el análisis garantizará que el mecanismo se mueva de tal modo que satisfaga la necesidad requerida. [3]

El análisis como tal no tiene ningún objetivo en el diseño, lo que el diseñador busca es llegar a la síntesis de estos mecanismos mediante su análisis, y es así, que el análisis se utiliza como herramienta para lograr obtener una maquina o mecanismo adecuado, pudiendo llegar a la definición de perfecto para la tarea que va a desempeñar.

1.3.4. Terminología de mecanismos

Un mecanismo está conformado por eslabones, los eslabones son elementos que se les consideran rígidos para el análisis a pesar de no serlo, estos están conectados entre sí para transmitir movimiento y fuerza, a elementos como resortes que no son rígidos por lo general no se les toma en cuenta para el análisis de su movimiento. [2]

La unión, también llamada articulación, entre eslabones se realiza mediante conexiones que permiten el movimiento relativo entre ellos, las uniones que se analizan en esta tesis corresponden a una unión de revoluta conocida también como de tornillo o de bisagra, que permiten rotaciones puras entre dos eslabones y la unión de corredera conocida también como de pistón o prismática, que permite el desplazamiento lineal entre los eslabones conectados.

A la unión de estos eslabones se los llama cadenas cinemáticas pudiendo ser cadenas cinemáticas abiertas o cerradas, en las cadenas cinemáticas abiertas al menos un eslabón está conectado solo con otro, mientras que en las cadenas cinemáticas cerradas todos los eslabones están conectados con al menos dos eslabones.

Un punto de interés es un punto específico del cual se requiere hacer el estudio cinemático poniendo principal interés en determinar posiciones velocidades y aceleraciones de este punto. [2]

1.3.5. Tipos de mecanismos: Mecanismos planos, espaciales

Los mecanismos en función de sus movimientos que producen los dividen en planos, esféricos y espaciales, para distinguirlos se basan en las características de los movimientos de los eslabones.

Mecanismo plano

Un mecanismo plano es aquel en el que todos sus componentes realizan movimientos planos en planos paralelos. El lugar geométrico de cualquier punto elegido de un mecanismo plano se representa con su verdadero tamaño y forma, es decir, en un solo dibujo. La transformación del movimiento de cualquier mecanismo se llama acople. En los mecanismos planos coinciden el plano del dibujo con el del movimiento, y están dibujados todos los miembros en un mismo plano, aunque están en planos paralelos. La transformación del movimiento de cualquier mecanismo de esta índole se llama coplanar. El eslabonamiento plano de cuatro barras, la leva de placa y su seguidor, y el mecanismo de corredera-manivela son ejemplos muy conocidos de mecanismos planos [2 pg7].

Mecanismo espacial

Mecanismo espacial no tiene restricción alguna en los movimientos relativos de las partículas. Un mecanismo espacial puede poseer movimientos de rotación y traslación en cualquiera de sus ejes. Cualquier eslabonamiento que comprenda un par de tornillo, por ejemplo, es un mecanismo espacial, porque el movimiento relativo dentro del par de tornillo es helicoidal. Estos se obtienen como una consecuencia de la geometría espacial en las orientaciones particulares de los ejes de sus pares.

1.3.6. Movilidad

En el análisis de mecanismos es importante encontrar el número de grados de libertad del eslabonamiento. El grado de libertad se puede definir como el número de entradas independientes requeridas para posesionar con exactitud todos los eslabones de un mecanismo respecto al suelo.

La movilidad de un mecanismo también se llama número de grado de libertad, generalmente las máquinas comerciales tienen un grado de libertad, el cual se define con GDL [2 pg8].

Ecuación de Gruebler

En los mecanismos planos conectados con uniones comunes, los grados de libertad se pueden calcular mediante la Ecuación de Gruebler.

$$\text{Grados de libertad} = 3(n - 1) - 2j_p - j_h \quad (1)$$

Donde:

n = número total de eslabones en el mecanismo

j_p = número total de uniones principales (uniones de pernos o correderas)

j_h = número total de uniones de orden superior (uniones de leva o engranes)

1.4. Método de BOND GRAPH

1.4.1. Introducción

El modelamiento por el método de BOND GRAPH es completamente diferente a la representación de sistemas tradicionales, parte del concepto de la conservación de la energía; la energía no se crea ni desaparece, solo se transforma en otro tipo de energía. En un enfoque sistémico, el modelamiento por BOND GRAPH se realiza analizando la potencia que entra y sale en un determinado elemento o sistema.

La representación de modelos físicos a lo largo de tiempo se lo ha hecho tratando de representar lo más cercano a un modelo real, el modelado por BOND GRAPH permite con gran facilidad elaborar modelos de sistemas físicos multidisciplinarios muy correspondientes a su esquema físico, acompañado de una fácil interpretación.

El modelamiento por el método de BOND GRAPH se centra en un enfoque intuitivo y físico, donde un sistema dinámico está compuesto por subsistemas, componentes o elementos básicos que interactúan intercambiando energía, esto parte del entendimiento de que en un sistema físico la energía puede ser almacenada, disipada o intercambiada. [5]

La palabra BOND GRAPH o en español “Gráfico de Enlaces”, viene de la representación esquemática que se realiza un sistema físico, donde un elemento o incluso sistemas son representados como nodos y el flujo de energía entre estos es representado por enlaces (BONDS), como se puede ver en la Figura 1 [5][6].

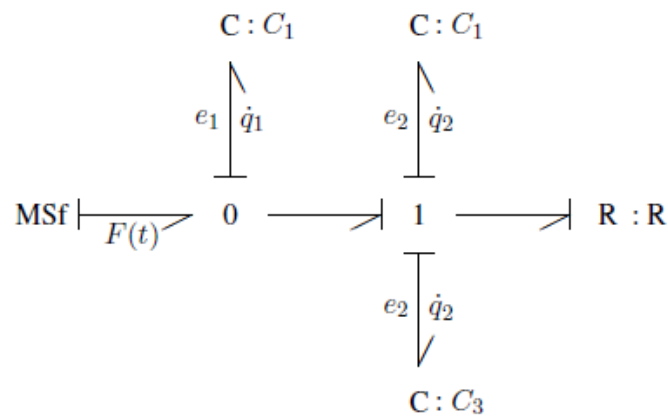


Figura 1 Bond Graph correspondiente a un sistema

En general la transferencia de energía entre subsistemas es posible por medio de enlaces ingenieriles como por ejemplo: ejes o barras mecánicas, cables eléctricos, conductos hidráulicos, mangueras etc. Estos subsistemas, componentes o elementos son representados por vértices en el BOND GRAPH, y sus interacciones energéticas mediante Power Bonds (enlaces de potencia), como se señaló, el Bond Graph refleja con gran similitud con la estructura física de un sistema, en otras palabras, como en realidad son conectados los elementos, esto se puede observar en la Figura 2 que representa el circuito magnético dentro de un motor de torque de una válvula electrohidráulica. [5]

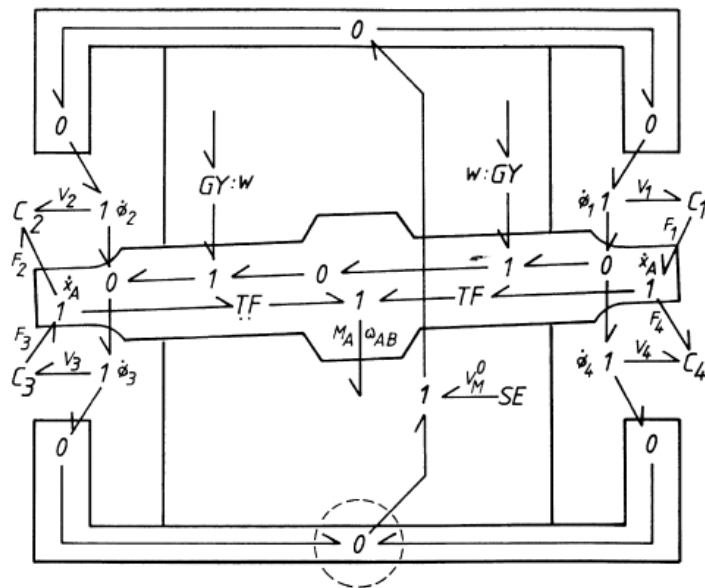


Figura 2 Circuito magnético dentro de un motor de torque de una válvula electrohidráulica.

Una de las características al utilizar Bond Graph es la posibilidad de trabajar con un enfoque multidisciplinario, en otras palabras se puede analizar como los sistemas interactúan analizando los efectos de sus diferentes dominios energéticos, como es el caso de un motor eléctrico donde la energía eléctrica es transformada para obtener un torque, o como en el caso de una turbina donde la energía hidráulica se transforma en eléctrica. [4]

Otra de las características del modelado con Bond Graph es la jerarquización y el recursivo modelamiento que se puede realizar, en otras palabras, el Bond Graph soporta una recursiva descomposición en sistemas y subsistemas, que son representados por Bond Graphs hasta que son identificados submodelos que representan procesos físicos básicos, y además pueden ser descritos por una ecuación, a estos procesos básicos de subsistemas se los puede denotar con palabras y símbolos alfanuméricos (encerados en una elipse).

Si un Bond Graph está compuesto de subsistemas denotados por palabras el Bond Graph toma el nombre de "Word Bond Graph" o en español Bond Graph de palabras, esto es el caso de que se tome un enfoque de mayor a menor, en el caso contrario, en un enfoque de menor a mayor los nodos de un Bond Graph de palabras son representados. [4]

Los nodos de un Bond Graph que representan subsistemas o elementos, intercambian energía con otros, esto lo hacen mediante puntos de conexión llamados “Power Ports” o puertos de energía, un puerto de energía se conecta con otro mediante un Power Bond lo que esquematiza la transferencia de energía de un nodo a otro o explicado de otra manera representa el intercambio energético ente subsistemas o elementos de un determinado sistema. Un nodo puede tener un sinnúmero de puertos a lo cual se le llama “Multiport” o multipuerto, estos nodos se conectan unos con otros y de esta manera forman el Bond Graph del sistema. [4]

1.4.2. Elementos de Bond Graph

Desde su concepción del método de Bond Graph ha sido muy bien estructurado, teniendo una representación estándar de elementos y variables, las que se describe a continuación:

1.4.2.1. Variables generalizadas.

El modelado de sistemas y componentes con el método de Bond Graph, se basa en flujos de Potencia y Energía.

El método de Bond Graph define cuatro variables generalizadas que se pueden separarse en dos tipos: Variables de Potencia y Variables de Energía.

Variables de Potencia, hacen referencia a las dos variables: de esfuerzo y flujo, se puede decir que cuando la potencia viaja a través de un puerto estas dos variables son las que intervienen, por lo que son llamadas variables de Potencia, aludiendo a que el producto de estas dos son igual a la potencia.

Esfuerzo, $e(t)$

Flujo, $f(t)$

$$Power = e(t) * f(t) \quad (2)$$

Las variables de energía son las que están relacionadas directamente con cada una de las variables de potencia, estas son el momento y el desplazamiento.

Momento $p(t)$

Desplazamiento $q(t)$

$$p(t) = \int e(t).dt \quad (3)$$

También conocida como Variable de momento generalizado

$$q(t) = \int f(t).dt \quad (4)$$

También conocida como Variable de desplazamiento generalizado

Las variables de energía pueden ser representadas de dos formas:

La potencia simple es la relación de cambio de energía

$$E = \int (\text{Potencia})dt = \int e(t).f(t)dt = \int e(q).dq(t) = \int f(p).dp(t) \quad (5)$$

Escribiéndose como:

$$E(q) = \int e(q).dq \quad (6)$$

O también:

$$E(p) = \int f(p).dp \quad (7)$$

Estas variables toman el nombre de variables generalizadas, ya que mediante estas se pueden representar en todos los dominios físicos , como por ejemplo para el caso de la velocidad lineal, la velocidad angular, el caudal o la intensidad de corriente, todas estas pueden ser representadas por flujo $f(t)$. [5]

Durante el manejo del Bond Graph se puede utilizar cualquiera de las dos representaciones, la forma clásica, donde por ejemplo la velocidad es representada por $v(t)$ o el caso de flujo $f(t)$. La mayor cantidad de usuarios de Bond Graph prefieren utilizar la representación clásica dejando la representación con variables generalizadas únicamente para cuando se realiza la interacción entre dominios. [5]

Una representación gráfica de la relación que existe entre las cuatro variables generalizadas (e, f, p, q), se logra a través del tetraedro de estado, las aristas del tetraedro están asociadas a cada una de las variables generalizadas, que para cada dominio tienen un significado específico. [7]

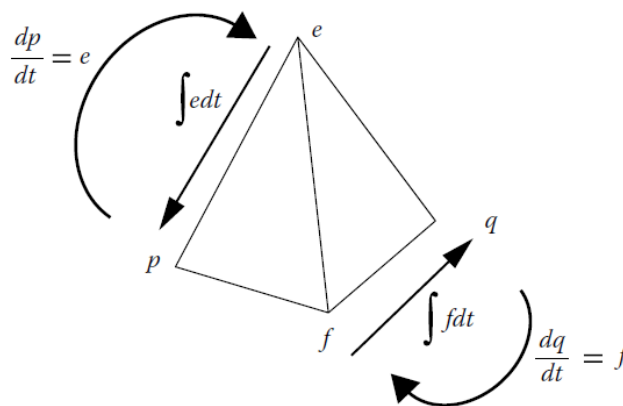


Figura 3 Tetraedro de estado [7]

En los diferentes dominios las variables generalizadas toman diferentes nombres, como es el caso del dominio de la mecánica traslacional, donde la variable de esfuerzo es la fuerza, la variable de flujo es la velocidad de movimiento. La variable de momentum es el momento lineal (masa * velocidad) y la variable de desplazamiento, es la distancia recorrida. Así como esta, se indican los nombres que toman las variables generalizadas en cada uno de los dominios posibles. [7]

Tabla 1 Variables generalizadas [4]

Energy domain	Effort e	Flow f	Generalised momentum p	Generalised displacement q
Translational mechanics	Force F [N]	Velocity v [m/s]	Momentum p [Ns]	Displacement x [m]
Rotational mechanics	Angular moment M [Nm]	Angular velocity ω [rad/s]	Angular momentum p_ω [Nms]	Angle θ [rad]
Electro-magnetic domain	Voltage u [V]	Current i [A]	Linkage flux λ [Vs]	Charge q [As]
	Magnetomotive force V [A]	Magnetic flux rate $\dot{\phi}$ [Wb/s]	–	Magnetic flux ϕ [Wb]
Hydraulic domain	Total pressure p [N/m ²]	Volume flow Q [m ³ /s]	Pressure momentum p_p [N/m ² s]	Volume V_c [m ³]
Thermodynamic	Temperature T [K]	Entropy flow \dot{S} [J/K/s]	–	Entropy S [J/K]
Chemical domain	Chemical potential μ [J/mole]	Molar flow \dot{N} [mole/s]	–	Molar mass N [mole]

Fuente: Bond Graph Methodology, Borutzky Wolfgang, pg 27.

1.4.2.2. Elementos Básicos del Bond Graph

Para poder representar la modelación mediante el método de Bond Graph se requieren varios elementos, para el estudio de estos elementos es común dividirlos en subcategorías, en función a la cantidad de puertos que intervienen en dichos componentes, como componentes 1-puerto, componente 2-puertos, componentes 3-puertos o más puertos.[7]

1.4.2.2.1 componentes 1-puerto (1-port)

Al hablar de elementos 1-puerto se hace mención a los elementos que es direccionado a través de un único puerto de poder (power port), y este puerto a un único puerto de variables de esfuerzo y flujo. [8]

Existen tres componentes 1-puerto principales, estos son categorizados por la naturaleza de su comportamiento, la forma de entender esto es considerar que tipo de energía /potencia manejan cada uno de estos elementos. Si consideramos la energía que fluye como resultado del trabajo hecho en un sistema mecánico, se tiene tres posibles opciones. Energía Potencial, Energía cinética y Energía disipada. De igual manera en un sistema eléctrico se tiene energía eléctrica, energía magnética y la energía disipativa. [7]

En todos los sistemas existen tres tipos de energías que intervienen, estos pueden ser asociados de la siguiente manera.

1.4.2.2.1.1 Resistor 1-puerto: Dispositivos disipadores de energía

Este tipo de elementos son disipadores de potencia, la energía que este dispositivo recibe es perdida en forma de calor generalmente o en algún otro tipo de energía, estos elementos no pueden almacenar energía, hay tres comunes elementos resistores: un resistor eléctrico, un amortiguador mecánico, y la caída de presión debido a rugosidad en las paredes de una tubería. [7]

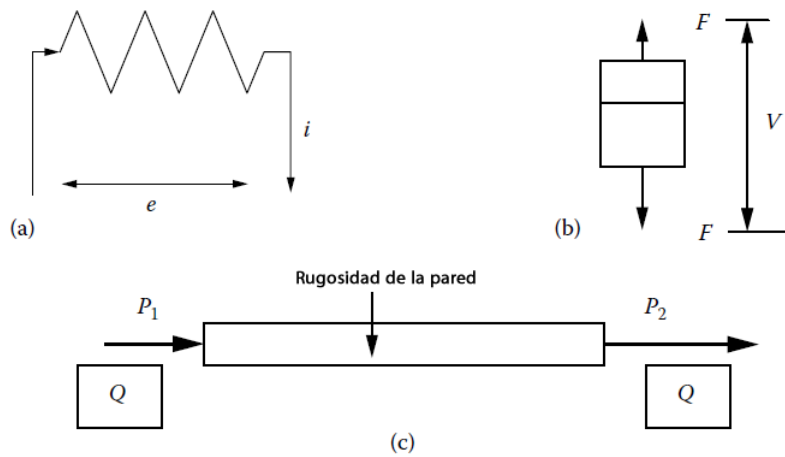


Figura 4 Elementos resistores 1-puerto a) resistor eléctrico, b) amortiguador mecánico, c) flujo a través de un tubo [7]

Todos estos elementos se comportan de la misma manera, estos pueden ser representados por un elemento resistor 1-puerto, como se muestra a continuación. [7]

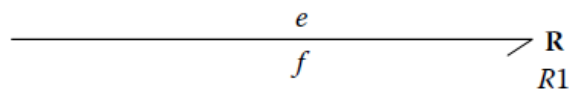


Figura 5 Representación elemento resistor 1-puerto [7]

La relación constitutiva general para el elemento de resistencia es

$$e = Rf \tag{8}$$

Para el caso de otros dominios se utiliza las variables correspondientes como se puede observar a continuación:

$$Fuerza = B * (velocidad) \tag{9}$$

En el dominio de la mecánica lineal, la fuerza es igual al producto del coeficiente de amortiguación, representado por B, con la velocidad.

1.4.2.2.1.2 Capacitor 1-puerto: Dispositivos almacenadores de energía

Estos elementos son característicos por almacenar y liberar energía, pero no disiparla. La relación constitutiva para el capacitor 1-puerto se relaciona con la variable de desplazamiento y puede ser escrita como: [7]

$$e = q/C \quad (10)$$

En el diagrama de Bond Graph se representa de la siguiente manera: [7]

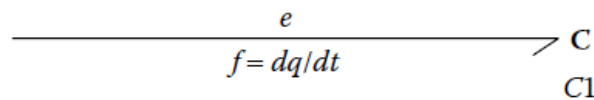


Figura 6 Representación elemento capacitor 1-puerto [7]

Para el caso de los elementos capacitivos se escribe el flujo como la derivada del desplazamiento, a diferencia de los elementos resistivos que relacionan directamente las variables de flujo y esfuerzo y no requieren de esta notación. [7]

Algunos ejemplos de estos elementos son: el condensador eléctrico en el dominio eléctrico, el resorte mecánico en el dominio mecánico, un tanque en el dominio hidráulico entre otros. [7]

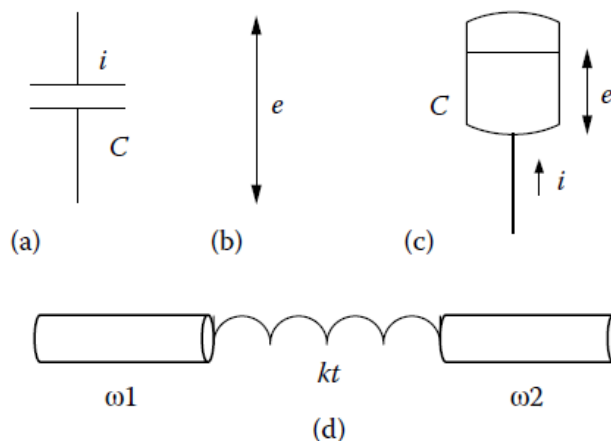


Figura 7 Elementos capacitores 1-puerto a) condensador eléctrico, b) resorte mecánico, c) fluido almacenado, d) resorte torsional [7]

1.4.2.2.1.3 Inductor/Inercia 1-puerto: Dispositivos almacenadores de energía

Un inductor es otro dispositivo almacenador de energía igual que el capacitor, se considera tratar por separado a los inductores por tener su relación constitutiva diferente. La relación constitutiva para un elemento capacitor es entre las variables de esfuerzo y desplazamiento, la relación constitutiva para el caso de un elemento inductor es entre las variables momentum y la variable de flujo [7].

$$p = If \quad (11)$$

donde:

p es la variable de momentum,

f es la variable de flujo, y

I es la inercia

La representación de Bond Graph para un elemento inductor es:

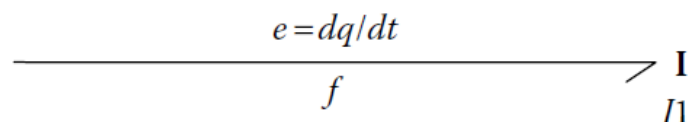


Figura 8 Representación elemento inductor 1-puerto [7]

Para el caso de los elementos inductores, el esfuerzo se escribe como la derivada de momentum, la energía almacenada en estos dispositivos puede ser escrita por la integral de la potencia instantánea, lo que lleva a escribir que la energía almacenada en un elemento inductivo como [7]:

$$E = \frac{1}{2I} p^2 \quad (12)$$

Algunos elementos que se comportan como elementos inductivos son: la masa o inercia en el dominio mecánico, la inductancia eléctrica, la masa de un fluido que atraviesa un ducto.

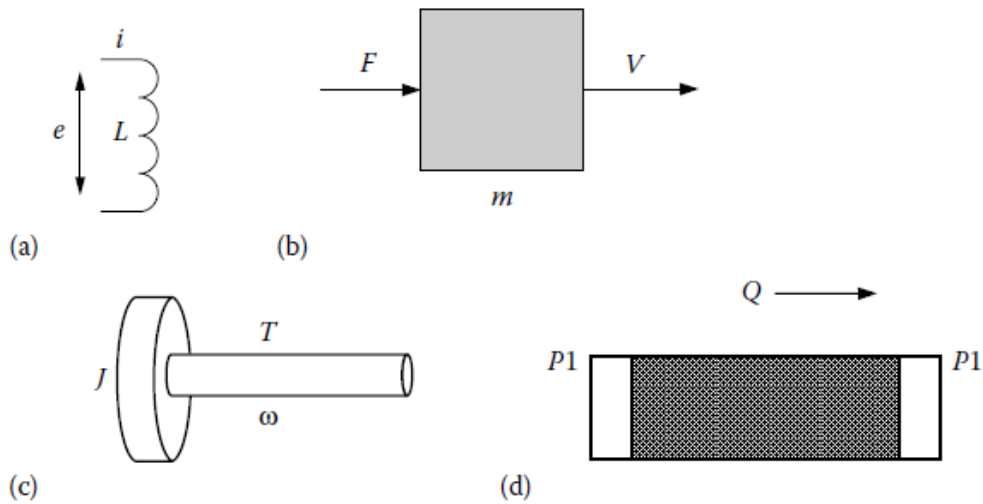


Figura 9 Elementos inductivos 1-puerto a) inductor eléctrico, b) masa mecánico, c) inercia rotacional, d) fluido inercial [7]

En el tetraedro de estado se puede observar las variables constitutivas para cada una de las relaciones constitutivas de cada uno de los elementos descrito [7].

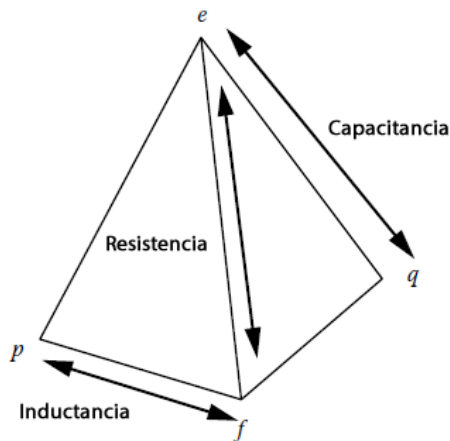


Figura 10 Tetraedro de estado mostrando las variables generalizadas que están en la relación constitutiva para cada uno de los elementos 1- puerto [7]

1.4.2.2.2 Otros elementos 1 puerto

Además de los elementos principales existen otros dos elementos, a los que se les llama elementos activos, estos elementos tienen que ver con la fuente que suministra ya sea esfuerzo o flujo a cualquier sistema, por lo que se pueden representar con la media flecha en dirección opuesta a la fuente de la misma, o entrando (suministrando) esfuerzo o flujo a un sistema [7].

Fuentes de esfuerzo

Estos elementos suministran potencia al sistema, la componente de esfuerzo es constante mientras que el flujo es determinado por la carga del sistema. Algunos ejemplos de este tipo de elementos son: baterías o fuentes de voltaje, fuerza gravitacional, flujo de calor o cualquier esfuerzo externo que actúe sobre un sistema. [7]

La fuente de esfuerzo es representada en un modelo de Bond Graph como Se



Figura 11 Representación de una fuente de esfuerzo

Fuentes de Flujo

Al igual que las fuentes de esfuerzo suministran potencia al sistema estas lo hacen mediante una fuente de flujo constante, y al igual que en la fuente de esfuerzo, la componente de esfuerzo es determinada por la carga del sistema. Algunos ejemplos para este tipo de elementos son: fuentes de corriente constante, bombas de fluido, velocidad o rotación constante. [7]

La fuente de flujo se representa en Bond Graph como Sf

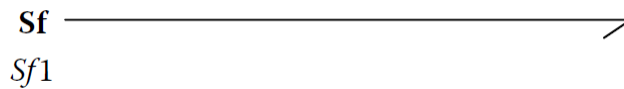


Figura 12 Representación de una fuente de flujo

1.4.2.2.3 Componentes 2-puertos (2-port)

Un elemento de 2-puertos es un elemento que recibe potencia de una parte de un sistema a través de un puerto y lo suministra a otra parte del sistema por otro puerto. De esta manera se pueden señalar dos tipos de elementos de 2-puertos: elementos transformadores y elementos giradores. [7]

1.4.2.2.3.1 Elementos transformadores (Transformer Element)

Es un elemento que sirve para relacionar el esfuerzo de entrada con el esfuerzo de salida mediante un factor, de igual manera con el flujo pero en este caso es lo contrario

$$\begin{aligned} e_2(t) &= N e_1(t) \\ f_1(t) &= N f_2(t) \end{aligned} \quad (13)$$

El elemento transformador se representa entre dos medias flechas por las letras **TR** que indican transformador, acompañadas del factor de transformación. [7]

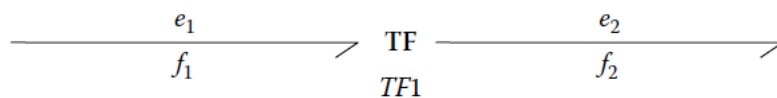


Figura 13 Representación de un elemento transformador

Algunos ejemplos de estos elementos transformadores son: transformador eléctrico, palanca mecánica, cremallera piñón, tren de engranajes, cilindro-pistón hidráulico entre otros. [7]

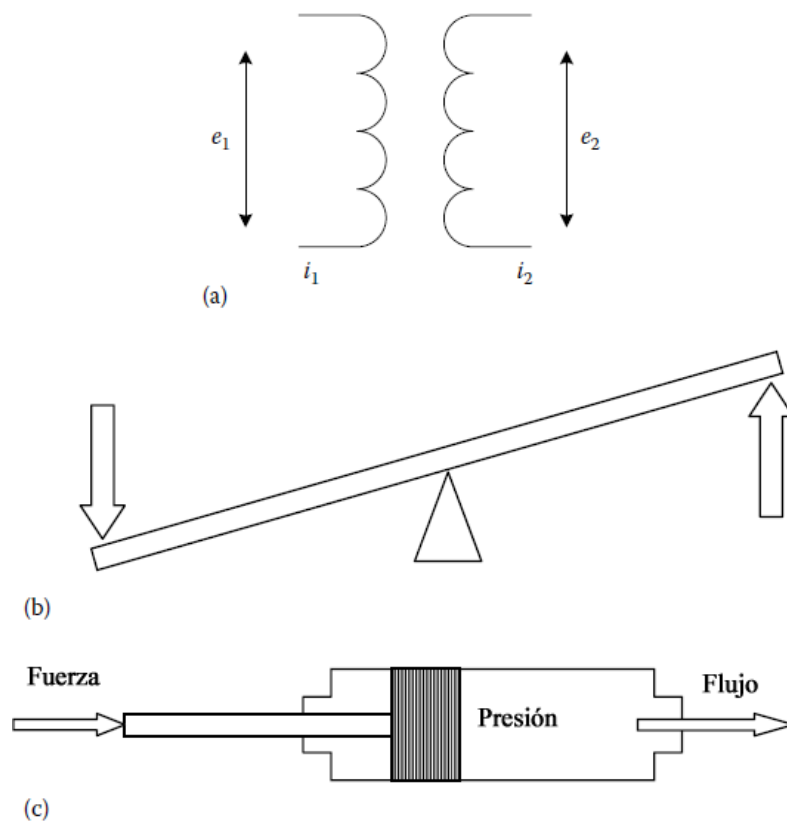


Figura 14 Representación esquemática de elementos transformadores en diferentes dominios a) transformador eléctrico, b) transformador mecánico (palanca), c) transformador hidráulico (pistón)

Para encontrar la relación de transformación se parte de un conocimiento básico del comportamiento del sistema, por ejemplo: para el caso de la palanca, la relación de transformación se obtiene al dividir las longitudes de los brazos. [7]

1.4.2.2.3.2 Elementos giradores (Gyrator Element)

Este elemento girador relaciona al esfuerzo de entrada con el flujo de salida y viceversa mediante un factor, esta relación satisface necesariamente el requerimiento en la que la potencia de entrada es igual a la potencia de salida, se representa mediante las siglas **GY**. [7]

$$\begin{aligned} e_2(t) &= Mf_1(t) \\ e_1(t) &= Mf_2(t) \end{aligned} \quad (14)$$

La representación del elemento girador se grafica como:

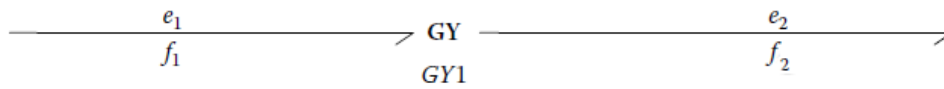


Figura 15 Representación de un elemento girador

Algunos ejemplos de elementos giradores son: bomba centrífuga, turbina, motores DC, giroscopio entre otros.

Al igual que en el elemento transformador la relación de giro se obtiene por entendimiento básico del comportamiento del sistema, como es el caso de un motor DC donde este factor depende de múltiples variables. [7]

1.4.2.2.4 Componentes 3-puertos o más (3-port or higher-port)

Este tipo de elementos es usado para denotar posiciones donde múltiples elementos se encuentran unos con otros, existen dos tipos principales de uniones: uniones tipo 1 y uniones tipo 0. [7]

1.4.2.2.4.1 Nodos de uniones tipo 0, uniones de esfuerzo común

También se les llama unión paralela o unión de flujo, en este tipo de uniones, los esfuerzos son iguales en cada uno de los nodos que llega y salen. Esto es similar a lo que pasa en un circuito eléctrico en paralelo donde el voltaje permanece constante mientras la corriente varía. [7]

La representación en el Bond Graph se la realiza mediante una unión tipo 0 como se muestra en la figura (16). [7]

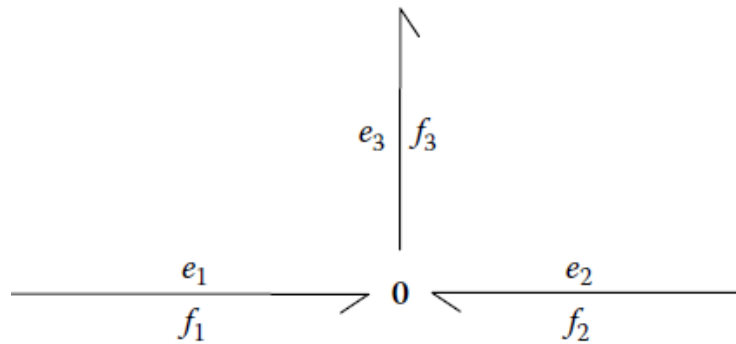


Figura 16 Representación de un nodo de unión tipo 0

Las propiedades que tienen las uniones de este tipo son: todos los esfuerzos son iguales y la suma de los flujos es igual a cero, por lo que se pueden concluir las siguientes ecuaciones. [7]

$$\begin{aligned} e_1(t) &= e_2(t) = e_3(t) \\ f_1(t) + f_2(t) &= f_3(t) \end{aligned} \quad (15)$$

En las uniones la dirección de la media flecha es muy importante, esta señala el signo que posee el flujo al igual que indica la dirección del flujo de potencia. [7]

1.4.2.2.4.2 Nodos de uniones tipo 1, uniones de flujo común

También llamadas uniones en serie o uniones de esfuerzo. En este caso la variable de flujo se mantiene constante, mientras que la de esfuerzo es diferente. Haciendo una comparación con un circuito eléctrico en serie se tiene que la corriente permanece constante mientras que el voltaje varía. [7]

La representación en el Bond Graph se la realiza mediante una unión tipo 1 como se muestra en la figura (17). [7]

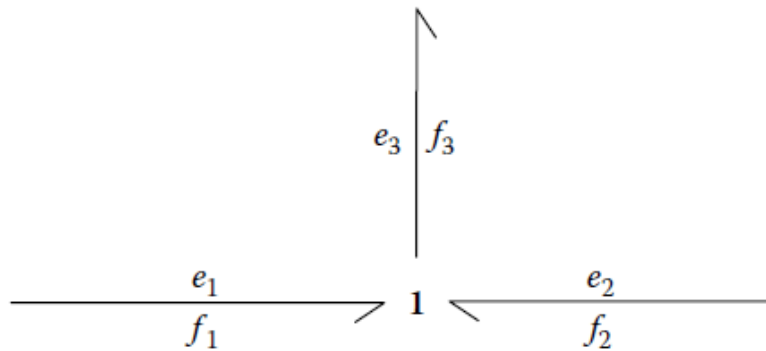


Figura 17 Representación de un nodo de unión tipo 1

Al igual que en las uniones de tipo 0, en las uniones de tipo uno se tiene que los flujos son iguales y la suma de los esfuerzos es igual a cero en los nodos.

$$\begin{aligned} f_1(t) &= f_2(t) = f_3(t) \\ e_1(t) + e_2(t) &= e_3(t) \end{aligned} \quad (16)$$

De la misma forma, la media flecha indica el signo en la ecuación y la dirección del flujo de potencia que tiene cada elemento

1.4.2.2.5 Componentes modulados

En la vida real es muy difícil encontrar elementos que generen flujos o esfuerzos constantes, la mayoría dependen de muchos factores, para lo que en un diagrama de Bond Graph se ha creado este tipo de elementos, tienen las mismas características de que se han señalado pero con la diferencia de que la señal que estos generan no es constante, dependiendo generalmente del tiempo. [7]

Para diferenciar estos elementos con señal constante se les antepone la letra M de modulated o modulado en español, como por ejemplo: una fuente de esfuerzo representada por S_e , para el caso que [1] [2] sea modulada se representa por MS_e . [7]

Capítulo 2.

Mecanismo de biela manivela

El mecanismo conocido como biela manivela hace referencia al conjunto entre un mecanismo de biela y otra llamada manivela, este mecanismo se utiliza principalmente en la transformación de movimiento lineal de vaivén en giratorio o viceversa.

El mecanismo de biela manivela es uno de los más utilizados dentro de las aplicaciones mecánicas. Es un elemento que sirve para la transformación de movimiento de uno lineal a otro rotacional o viceversa. Consta de un elemento llamado biela y de otro llamado manivela conectado como se muestra en la figura.

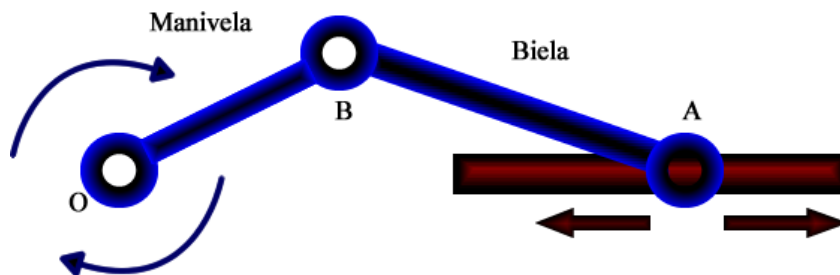


Figura 18 Representación de un mecanismo biela manivela.

El mecanismo tiene un eje fijo en el que rota la manivela representado por O, un punto de unión B, y la biela con un punto deslizante en el punto A sobre la guía.

Este mecanismo es un mecanismo plano que tiene un grado de libertad ya que posee el extremo "A" restringido uno de sus movimientos.

2.2 Metodología de la construcción de un Bond Graph

Como partida para la diagramación de un sistema se enlista las diferentes variables y elementos presentes en los sistemas mecánicos que intervienen dentro del campo de la mecánica traslacional y la rotacional.

Tabla 2 Variables de potencia y Energía para sistemas mecánicos traslacionales [8]

Variables Generalizadas	Mecánica Traslacional	Unidades SI
Esfuerzo, e	Fuerza, F	Newton (N)
Flujo, f	Velocidad, V	Metros por Segundo (m/s)
Momentum, p	Momentum, P	N-s
Desplazamiento, q	Desplazamiento, X	m
Potencia, P	$F(t) V(t)$	Vatios (N-m/s = W)
Energía, E	$\int^x F dx, \int^P V dP$	Julios (N-m = J)

Tabla 3 Variables de potencia y Energía para sistemas mecánicos Rotacional [8]

Variables Generalizadas	Mecánica Traslacional	Unidades SI
Esfuerzo, e	Torque, τ	Newton-metros (N-m)
Flujo, f	Velocidad angular, ω	Radianes por segundo (rad/s)
Momentum, p	Momentum angular, p_τ	N-m-s
Desplazamiento, q	Angulo, θ	rad
Potencia, P	$\tau(t)\omega(t)$	Vatios (N-m/s = W)
Energía, E	$\int^\theta \tau d\theta, \int^{p_r} \omega dp_r$	Julios (N-m = J)

De igual manera para los elementos básicos:

Tabla 4 Relaciones constitutivas para elementos básicos [8]

Elemento	Variables General	Mec. Traslacional	Mec. Rotacional
Se	e	f	T
Sf	f	v	ω
R	$e = Rf ; f = e/R$	$F = Bv ; v = F/B$	$T = B\omega ; \omega = T/B$
C	$e = \frac{q}{C} ; q = eC$	$F = kx ; x = F/k$	$T = k\theta ; \theta = T/k$
I	$p = If ; f = p/I$	$p = mv ; v = p/m$	$p = J\omega ; \omega = p/J$

Dónde:

B, es el coeficiente de amortiguamiento lineal o rotacional, expresado en N-s/m para el caso de mecánica traslacional y N-m-s para mecánica rotacional.

k, es el coeficiente de rigidez del resorte lineal o rotacional, expresado en N/m para el caso de mecánica traslacional y N-m/rad para mecánica rotacional. La relación $k=1/C$, donde C es el coeficiente de conformidad.

m, es la masa expresada en Kg.

J, es la inercia del momento polar expresado en N-m-s².

Para el caso de las variables representadas con la misma letra, en mecánica rotacional y traslacional se acostumbra a diferenciarlas mediante un sufijo que haga referencia al Bond asociado.

Para analizar el mecanismo de biela manivela se parte del planteamiento del Bond Graph correspondiente al este sistema, para lo cual se parte hablando de cómo se logra construir el modelo para luego analizarlo.

La metodología para la construcción de un Bond Graph parte desde la esquematización del sistema físico real, la cantidad de elementos que se deseen incluir en el análisis determinara la complejidad del Bond Graph.

La metodología de Bond Graph se divide en tres pasos principales, cada uno pudiendo realizarse indistintamente y utilizando métodos diferentes; la metodología de Bond Graph es tan amplia que existen un sin número de alternativas para el planteamiento y resolución de los problemas.

Los tres pasos fundamentales para resolver un sistema mediante el método de Bond Graph son:

1. Dibujo del Bond Graph
2. Planteamiento de ecuaciones y obtención de la ecuación de estado
3. Resolución de la ecuación de estado

2.2.1 Dibujo del Bond Graph

En la actualidad no existe un procedimiento estándar para dibujar un Bond Graph, muchos autores de libros proponen sus métodos de tal manera de llegar a plantear un Bond Graph que represente de mejor forma la realidad.

La diagramación de modelos de Bond Graph está enfocado de dos formas, la primera es un enfoque general donde se analizan los sistemas de una forma simple colocando los Bonds y nodos necesarios para satisfacer las necesidades de elementos de dicho sistema, una segunda forma, es utilizar algoritmos de Bonds creados principalmente por Rosenberg y Karnoop para determinar otro conjunto de variables que no son tomadas en cuenta o requieren de un análisis posterior en el método general. [5]

En el caso del enfoque general, los programas de computación dedicados a la resolución por el método de Bond Graph parten su estudio desde la esquematización del modelo físico, esto se debe a que una de las mejores formas de dibujar el Bond Graph inicia desde aquí. Partiendo de la premisa de que a cada elemento físico le

corresponde un Bond Graph y que cada subsistema, sistema o elemento se puede unir a otro, se inicia dibujando el Bond Graph de cada uno de los de los elementos y luego se los une para formar así el Bond Graph completo.

Otra forma de dibujar un Bond Graph desde un enfoque general parte de identificar de donde se encuentra la energía y ver hacia donde fluye, esto es más evidente dentro de un dominio térmico, tomando en cuenta que el calor que genera, se pierde, se transmite y se almacena. De la misma manera en la mecánica traslacional y rotacional se analiza este comportamiento el movimiento en el elemento que se ve detenido por fricción, amortiguación, etc. Este método es más rápido pero tiene el inconveniente de que se requiere un dominio profundo del tipo de interacciones entre los elementos, los nodos tipo 0 y 1 no son fáciles de identificar y en este caso no existe un procedimiento establecido para completar el Bond Graph.[4]

2.2.1.1 Método formal para dibujar Bond Graph

Se describe a continuación el método habitualmente utilizado [7 pg72][4 pg66]:

1. Para cada velocidad (velocidad angular en el caso de rotación), se establece un nodo tipo 1, algunos de estos nodos representaran velocidades absolutas y otras relativas.
2. Insertar los elementos de un puerto tipo fuerza (torque en el caso de rotacional) entre un par de nodos tipo 1 usando un nodo tipo 0. Atar los elementos capacitivos y resistivos por power bonds y conectarlos entre dos nodos tipo 1 usando un nodo tipo 0, por ultimo añadir los elementos inercia a los respectivos nodos tipo 1.
3. Añadir los elementos transformadores y giradores entre las parejas de fuerzas que lo requieran, donde exista un cambio en el flujo y esfuerzo.
4. Asignar la dirección de los power bonds.
5. Eliminar cualquier nodo tipo 1 que no tengan secuencia y los que cuya velocidad sea cero (tierra), además de todos los Bonds Conectados a estos.

6. Simplificar el Bond Graph utilizando las reglas de simplificación.

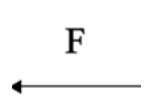
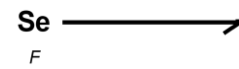
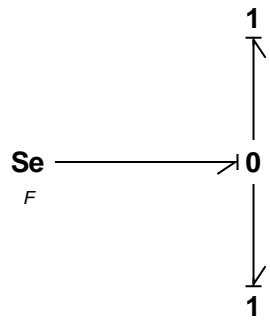
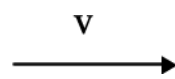
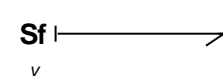
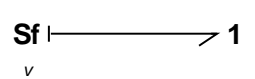
7. Analizar la causalidad.

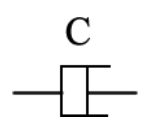
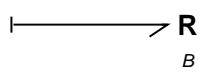
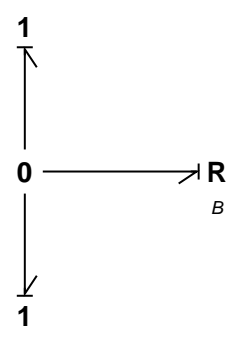
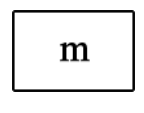
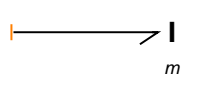
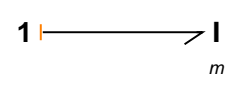
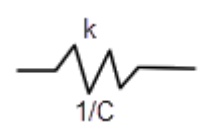
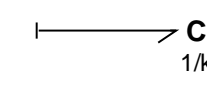
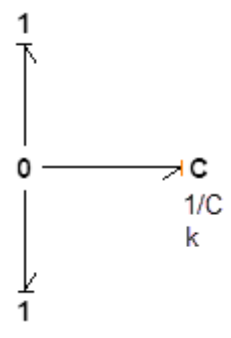
El análisis de causalidad no es un paso obligatorio ya que sin este, también se podría resolver el Bond Graph.

Una forma fácil de dibujar un Bond Graph parte de la obtención de elementos estándares que permiten identificarlos rápidamente y graficar directamente el Bond Graph. Estos elementos estándar se unen entre si formando el Bond Graph del sistema, luego de lo cual este se simplifica con las reglas de simplificación descritas en el Anexo 1, dando como resultado un Bond Graph apropiado el sistema, con pocos elementos.

Se presentan los elementos estándares para sistemas mecánicos traslacionales y rotacionales.

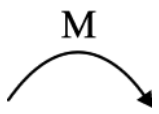
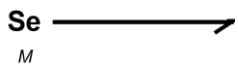
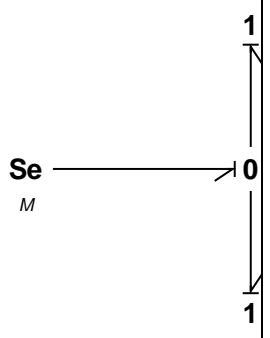
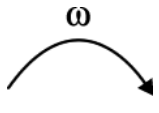
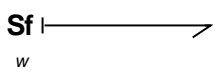
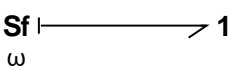
Tabla 5 Representación de Elementos Mecánicos Lineales en Bond Graph

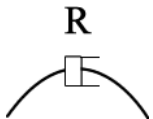
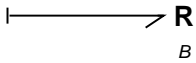
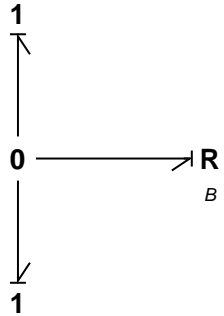
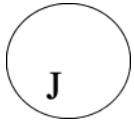
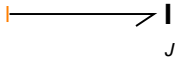
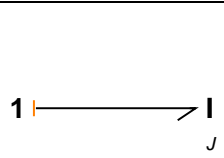

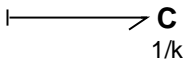
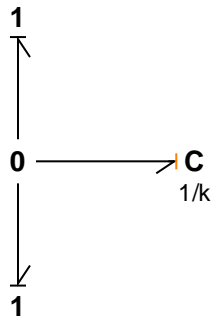
Elemento	Representación gráfica	Bond	Ecuaciones Características	Bond Graph
Fuente de esfuerzo Se			$e = F$ $\dot{f} = m = F$	
Fuente de Flujo Sf			$e = \dot{v} = m$	

Resistor			$e = B v$	
Inductor			$p = m v$ $e = \dot{p}$	
Capacitor			$e = k x$ $e = x/C$	

Del mismo modo los elementos estándares para sistemas mecánicos rotacionales son:

Tabla 6 Representación de Elementos Mecánicos Rotacional en Bond Graph

Elemento	Representación gráfica	Bond	Ecuaciones Características	Bond Graph
Fuente de esfuerzo Se			$e = M$ $\dot{f} m = M$	
Fuente de Flujo Sf			$e = \dot{\omega} J$	

Resistor			$e = B \omega$	
Inductor			$p = \theta = m \omega$ $e = \dot{p}$	
Capacitor			$e = k \theta$	

Tomando en cuenta cada uno de los pasos y la representación estándar de cada elemento para dibujar el diagrama correspondiente al mecanismo de biela manivela se tendría lo siguiente:

Dado el diagrama físico

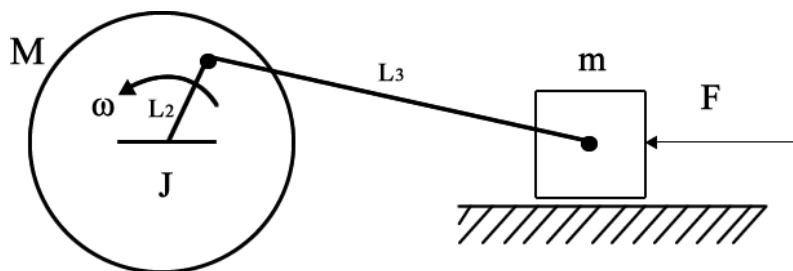


Figura 19 Representación Física de un mecanismo Biela manivela.

Dónde:

M es el momento angular de la manivela

ω es la velocidad angular de la manivela

- L_2 es el radio de giro, distancia del brazo de la manivela.
- J es la inercia de la manivela
- L_3 es la longitud del brazo de la biela
- m es la masa de la corredera
- v es la velocidad de la corredera
- F es la fuerza que actúa sobre la corredera

Paso 1:

El mecanismo biela manivela tiene dos velocidades fácilmente distinguibles la velocidad angular de la manivela y la velocidad lineal de la corredera, estas se las representan mediante nodos tipo 1, el nodo de la izquierda representa la velocidad angular ω de la manivela, mientras que el de la derecha corresponde a la velocidad v de la corredera.



Figura 20 Representación de nodos tipo 1 correspondiente a las velocidades.

Paso 2:

Insertando los elementos de tipo fuerza y torque unidos a cada una de las velocidades correspondientes respetando la inserción de elementos correspondientes, de esta manera se tiene.

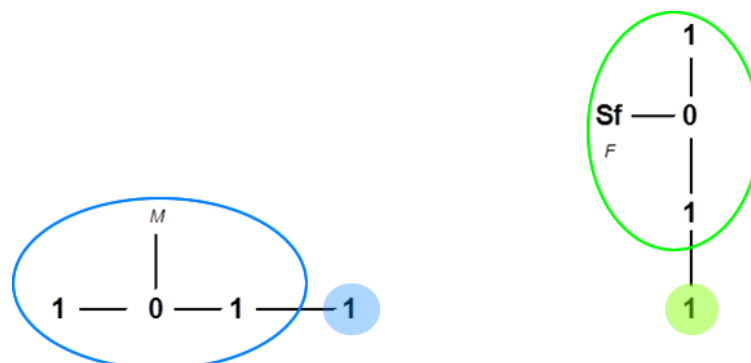


Figura 21 Insertando elementos tipo fuerza.

Atar los elementos capacitivos y resistivos de igual manera que en el caso de las fuentes, respetando su diagrama, en concreto como no hay elementos capacitivos solo la fricción en la corredera como elemento resistivo.

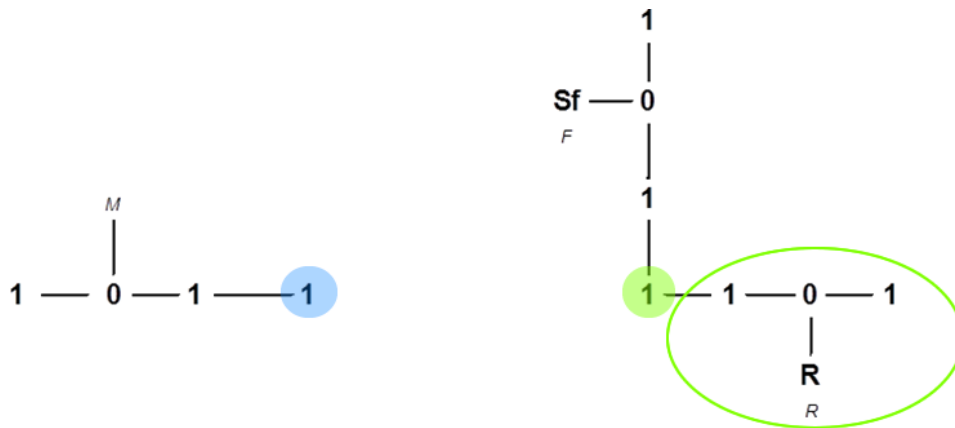


Figura 22 Insertando elementos capacitores y resistores.

Insertando los elementos de tipo inercia colocando los respectivos valores de m y J de las inercias de cada elemento.

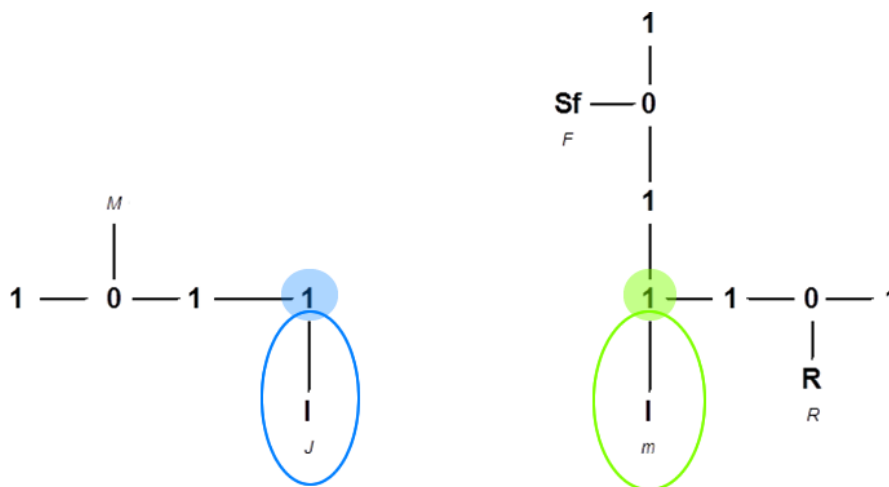


Figura 23 Insertando elementos Inercia.

Paso 3

Añadir los elementos transformadores y giradores, para este caso el elemento utilizado es un transformador ya que se puede obtener una relación directa que depende de θ , entre la velocidad angular ω de la manivela, y la velocidad lineal de la corredera v , lo que se demuestra más adelante.

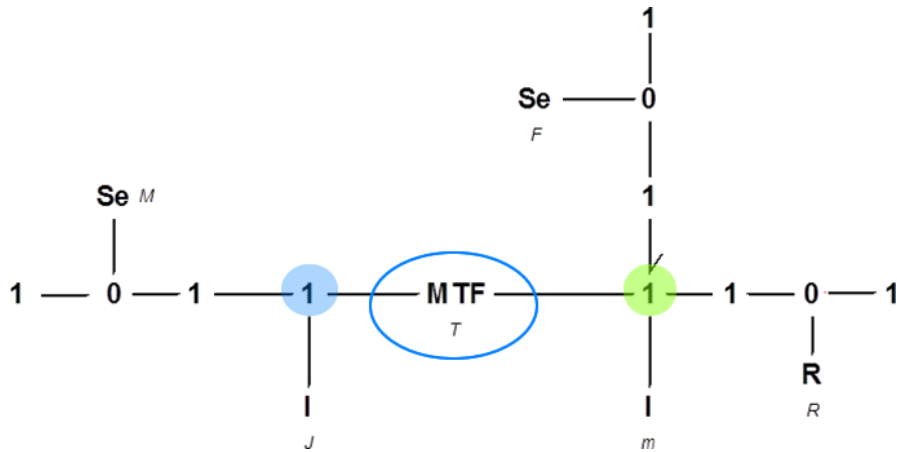


Figura 24 Insertando elementos transformadores.

Paso 4

Analizando la transmisión de movimiento la dirección de los power bonds, sabiendo que los elementos fuente Se entregan energía al sistema, mientras que los elementos inercia y resistencia sacan energía, se coloca las medias flechas siguiendo el flujo de esta energía, el Bond Graph queda:

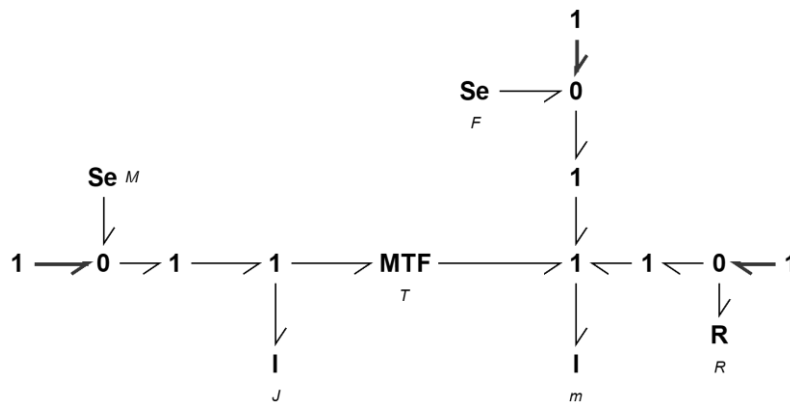


Figura 25 Asignación de dirección de Elementos.

Paso 5

Eliminando los nodos tipo 1 que no tienen secuencia, o que deja el Bond Graph reducido a:

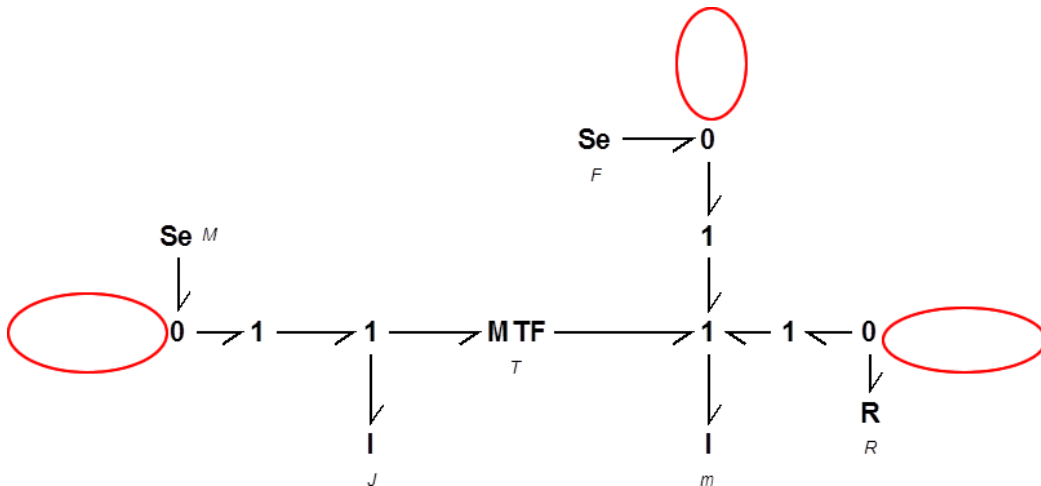


Figura 26 Eliminación de nodos tipo 1.

Paso 6

Simplificando el Bond Graph mediante las reglas de simplificación descritas en el Anexo 1, lo que elimina los nodos tipo 0 que tienen una entrada y una salida, resultando:

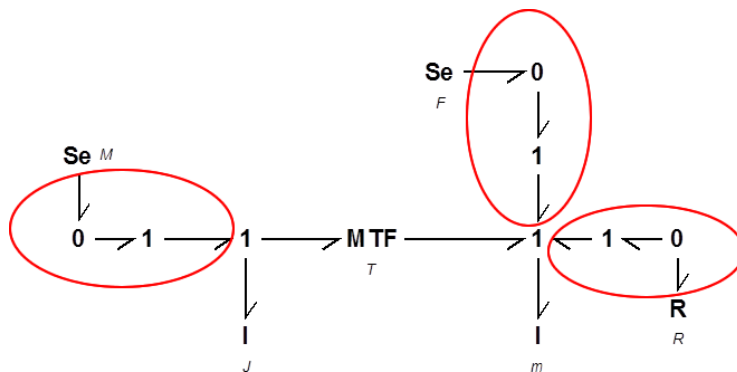


Figura 27 Elementos a Simplificar del Bond Graph.

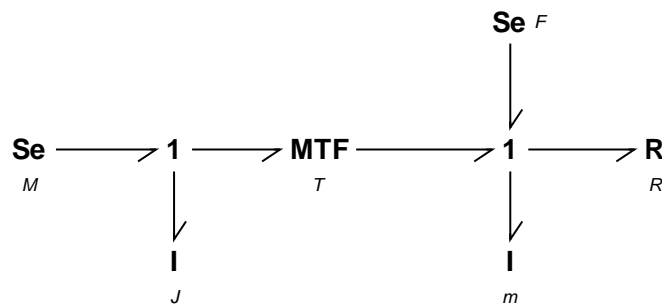


Figura 28 Simplificación del Bond Graph.

Paso 7

Analizando la causalidad resulta, partiendo de la fuente $Se: M$ se tiene causalidad de esfuerzo, la fuente $Se: F$ de la misma manera tiene causalidad esfuerzo, la inercia $I: J$ asumimos causalidad esfuerzo y para la inercia $I: m$ asumimos causalidad flujo, la resistencia colocamos la causalidad flujo, resultando:

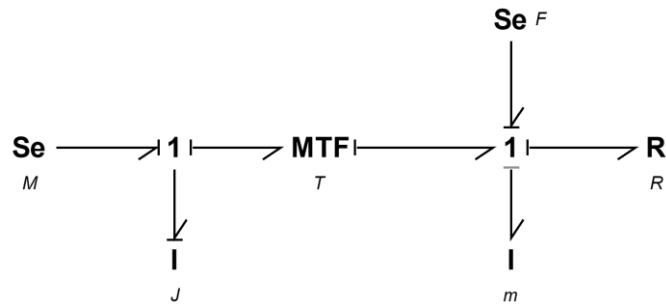


Figura 29 Análisis de causalidad.

De esta manera queda determinado el Bond Graph correspondiente al mecanismo biela manivela, cada uno de los elementos son fácilmente distinguibles en el modelo físico, y gráficamente la relación se observa como:

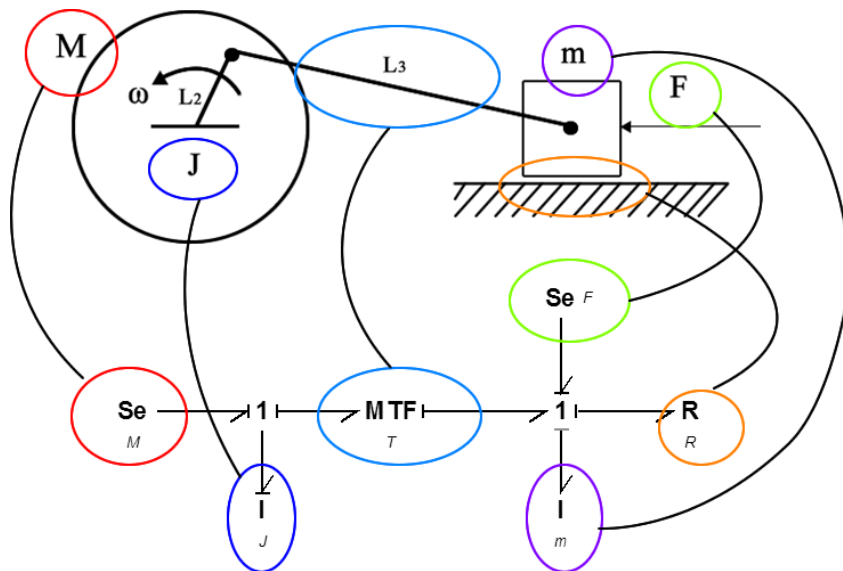


Figura 30 Relación entre elementos Físicos y Bond Graph.

En un modelo analizado por Bond Graph, se incluye todos los efectos posibles que afectan al sistema, permitiendo analizar el modelo variando cada uno de estos elementos, en el caso de que no existiera algún elemento se le considera cero y de esta manera no influye en el sistema.

2.2.1.2 Algoritmo específico para dibujar un Bond Graph

A diferencia del método anteriormente descrito en un algoritmo de Bonds, donde se permite escribir directamente el sistema añadiendo pequeños elementos modificatorios que se deseen, sin preocuparse de seguir o cumplir la lista de pasos descritos.

Entre los muchos algoritmos existentes para la resolución de mecanismos resalta el planteado por Rosenberg, el cual se aplica para mecanismos de barras donde los extremos de las mismas se encuentran conjugados con otros elementos.

El algoritmo planteado por Rosenberg para mecanismos de barras, fue en principio desarrollado para analizar el comportamiento de un elemento libre en el espacio, por lo que también puede ser utilizado en otros mecanismos que dispongan de dos puntos de interés, el algoritmo analiza las velocidades existentes, en los dos puntos de interés (en los extremos de la barra), y en centro de masa del elemento, que se la relaciona mediante elementos transformadores modulados como se describe a continuación:

Teniendo un elemento con puntos de interés denominados 1 y 2, se puede representar sus elementos como se describe en la figura siguiente:

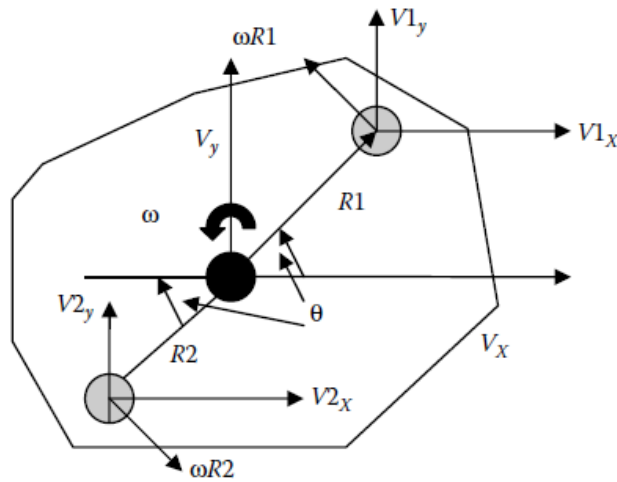


Figura 31 Variables de un Elemento [7 pg 391]

Si se analiza por separado cada uno de los ejes, las velocidades de cada uno de los puntos de interés provocan el movimiento de la masa como se describe:

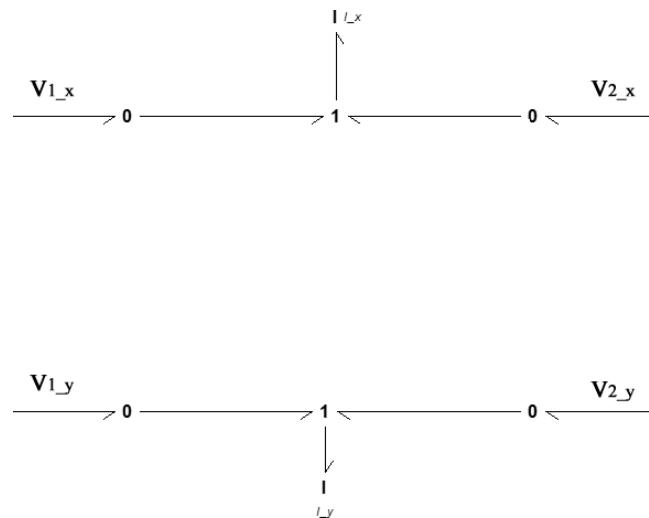


Figura 32 Análisis de velocidades en ejes separados

Además de las inercias en los ejes x y y la inercia del centro de gravedad también se ve afectada producto de estas velocidades, por lo que podemos graficar este comportamiento mediante elementos transformadores que salgan de las velocidades e influyan sobre la inercia rotacional J , esto en el Bond Graph se representa en la figura 33.

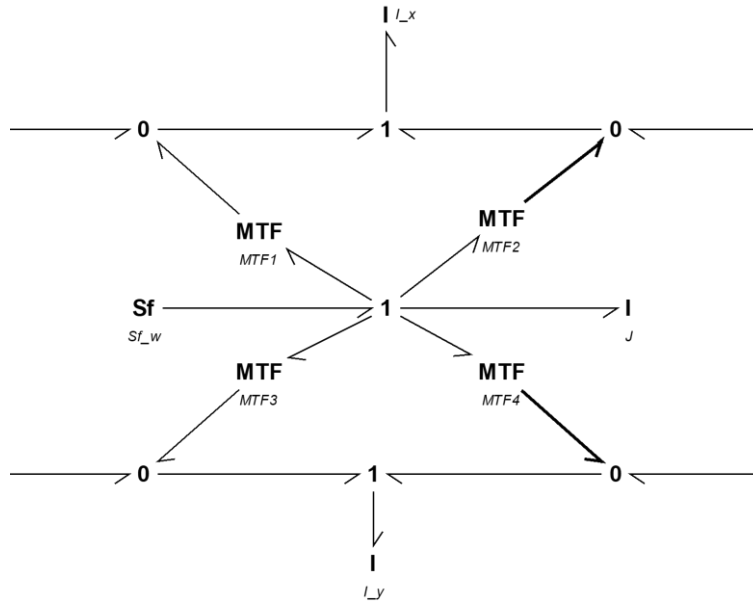


Figura 33 Modelo de bond Graph para un elemento con movimiento libre [7 pg 400] .

Una fuente de flujo añadida a la inercia es el producto de un bond adicional sobre la inercia J , conjuntamente con las velocidades de los puntos 1 y 2 se puede representar las influencias del medio sobre este elemento. En el Bond Graph descrito los dos puntos del elemento de preferencia están alineados con el centro de gravedad, que permite reducir notablemente la complejidad de las ecuaciones, [7 pg. 391]

En este modelo se distingue tres partes, una parte central del nodo tipo 1, que representa la velocidad angular del elemento, y en los extremos superior e inferior dos nodos tipo 1, que representan las velocidades correspondientes a la dirección x y y respectivamente, hacia los extremos derecho e izquierdo se encuentran los Bonds encargados de transmitir el flujo fuera o hacia el elemento.

Para el caso de un mecanismo biela manivela se conjuga dos elementos de este tipo mediante nodos tipo 1, uniendo sus extremos uno con otro, fijando uno de sus extremos y restringiendo el movimiento en la dirección y en el otro extremo.

El Bond Graph resultante es:

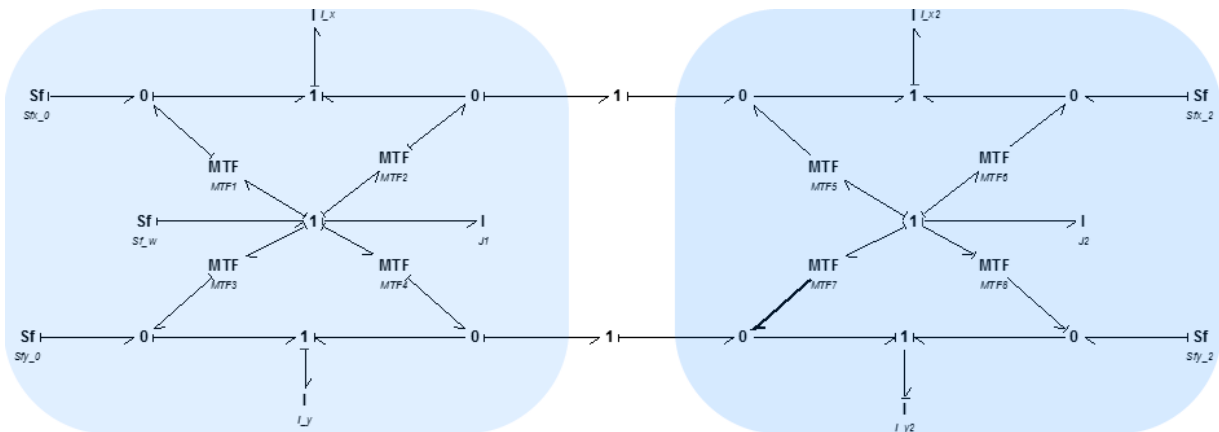


Figura 34 Modelo de Bond Graph para un mecanismo biela manivela .

Añadiendo un elemento resistencia en la corredera se obtiene:

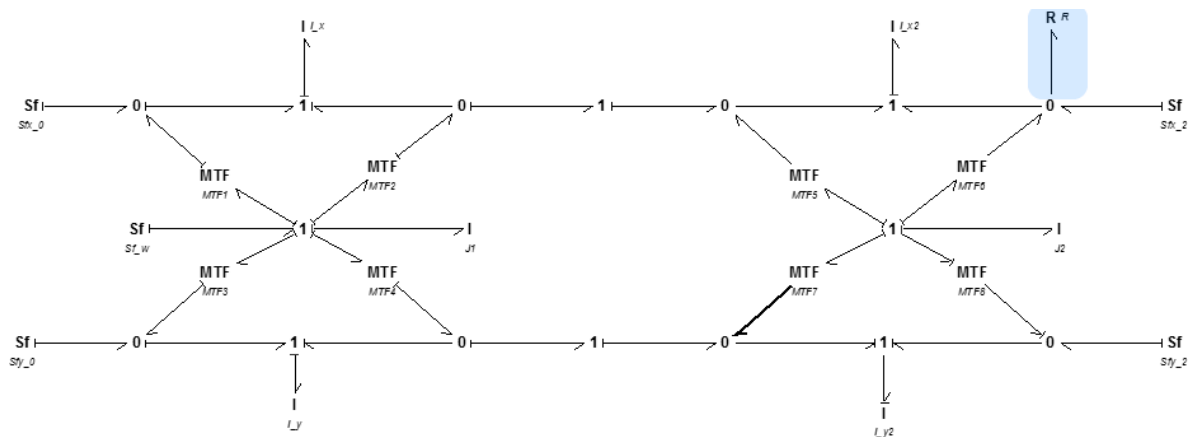


Figura 35 Modelo de Bond Graph para un mecanismo biela manivela con resistencia.

2.2.2 Planteamiento del sistema de ecuaciones

Se logra la obtención del sistema de ecuaciones del Bond Graph al escribir todas las ecuaciones de las variables asociadas a cada uno de los Bonds presentes, para esto se tiene que tomar en cuenta la relación constitutiva correspondiente para los elementos básicos que como ya se indicó en el capítulo 1 [7 pg147]:

$$\text{Resistencia } e = R f \quad (17)$$

$$\text{Capacitancia } e = \frac{q}{c} = k \cdot q \quad (18)$$

$$\text{Inductancia } p = I f \quad (19)$$

También se utiliza las relaciones entre las variables descritas en el capítulo 1

$$q = \int f dt \quad (20)$$

$$p = \int e dt \quad (21)$$

Para comenzar a escribir las ecuaciones la mejor manera es numerar los Bonds de tal manera que se los pueda identificar correctamente, para el caso de la figura 28 la numeración quedaría:

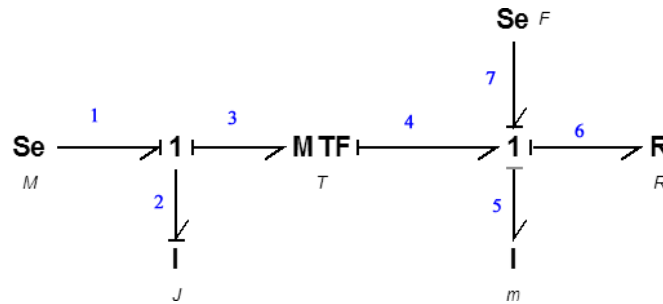


Figura 36 Numeración de Bonds.

Como norma general las variables de esfuerzo y flujo se denotan con su correspondiente e y f acompañado de un subíndice asociado al número de Bond que se etiqueta.

Para el Bond Graph descrito en la figura 33, la obtención de ecuaciones resulta de escribir las ecuaciones constitutivas de cada elemento y las relaciones presentes en los nodos tipo 0 y 1.

Bond 1

Estando f_1 conectado con un nodo tipo 1 los flujos de todos los Bonds conectados a este tienen el mismo valor

$$f_1 = f_2 = f_3 = \omega \quad (22)$$

$$e_1 = M \quad (23)$$

Bond 2

Partiendo de las ecuaciones características del Bond inercia se puede escribir al e_2 como función del flujo f_2 .

$$f_2 = \frac{p_2}{J} = \omega \quad (24)$$

$$e_2 = \dot{p}_2 = J\dot{f}_2 = J\dot{\omega} \quad (25)$$

Bond 3

Para determinar el esfuerzo e_3 se recurre a la definición de nodo tipo1 donde la suma de todos los nodos es igual a cero.

$$f_3 = \omega_3 \quad (26)$$

$$e_3 = e_1 - e_2 = M - J\dot{\omega} \quad (27)$$

Bond 4

El flujo f_4 por ser parte de un nodo tipo 1 comparte el valor de su flujo con los demás elementos vinculados a este. Mientras que el esfuerzo e_4 se relaciona mediante el esfuerzo e_3 por una transformación denominada T que se estudia más adelante.

$$f_4 = f_5 = f_6 = f_7 = v \quad (28)$$

$$e_4 = \frac{1}{T} e_3 \quad (29)$$

Bond 5

Al igual que el Bond 2, la ecuación característica del elemento inercia permite escribir la ecuación de esfuerzo en función de flujo.

$$f_5 = \frac{p_5}{m} \quad (30)$$

$$e_5 = \dot{p}_5 = m \dot{f}_5 \quad (31)$$

Bond 6

El esfuerzo e_6 se escribe como la diferencia de los elementos que entran y salen en el nodo tipo 1

$$f_6 = v \quad (32)$$

$$e_6 = e_4 - e_5 - e_7 \Rightarrow B v = \frac{M - j\omega}{T} - m\dot{v} - F \quad (33)$$

Bond 7

Se escriben los correspondientes valores manifestados como datos,

$$f_7 = v \quad (34)$$

$$e_7 = F \quad (35)$$

Todas estas ecuaciones del sistema escritas en una tabla resultan

Tabla 7 Sistema de ecuaciones para el mecanismo

Grafo	f	e
1	$f_1 = f_2 = f_3 = \omega$	$e_1 = M$
2	$f_2 = \frac{p_2}{J} = \omega$	$e_2 = \dot{p}_2 = J\dot{f}_2 = J\dot{\omega}$
3	$f_3 = \omega_3$	$e_3 = e_1 - e_2 = M - J\dot{\omega}$
4	$f_4 = f_5 = f_6 = f_7 = v$	$e_4 = \frac{1}{T}e_3$
5	$f_5 = \frac{p_5}{m}$	$e_5 = \dot{p}_5 = m\dot{f}_5 = m\dot{v}$
6	$f_6 = v$	$e_6 = e_4 - e_4 - e_7 = Bv$ $= \frac{M - j\dot{\omega}}{T} - m\dot{v} - F$
7	$f_7 = v$	$e_7 = F$

El método de Bond Graph advierte que se tendrán como resultado un par de ecuaciones diferenciales de primer orden, estas son las correspondientes a la ecuación de esfuerzo del Bond 7 sin reemplazar el esfuerzo del Bond 2, acompañada con la ecuación de flujo de la ecuación 2 quedando el siguiente sistema:

$$\omega = \frac{p_2}{J}$$

$$v = T \omega \tag{36}$$

$$m\dot{v} = \frac{M - \dot{p}_2}{T} - B v - F$$

Otra forma de representar este sistema de ecuaciones, es mediante la derivación de una de las ecuaciones y su correspondiente reemplazo en la otra ecuación, lo que da como resultado:

$$m\dot{v} + B v = \frac{M - J\dot{\omega}}{T} - F \quad (37)$$

Partiendo de esta ecuación al reemplazar la velocidad como la derivada del desplazamiento se obtiene:

$$m\ddot{x} + B \dot{x} = \frac{M - j\dot{\omega}}{T} - F \quad (38)$$

Esta ecuación tiene la forma de la ecuación de una masa vibrante $m\ddot{x} + B \dot{x} + kx = G(t)$, donde B es el factor de amortiguación, el valor de kx no aparece debido a que no se dispone de un elemento capacitor y $G(t)$ son las fuerzas externas aplicadas al mecanismo, dejando la posibilidad de resolver el sistema de ecuaciones de primer grado o la ecuación de segundo grado.

Para determinar el valor que toma T , el factor de transformación, se analiza la relación que existe entre la velocidad angular ω de la manivela y la velocidad v de la corredera, para lo cual se toma las ecuaciones presentes en el capítulo 3 que describen la relación entre los diferentes elementos del mecanismo.

Tomando como referencia a las ecuaciones (58) a la (63). El coeficiente T de transformación resulta.

$$v_4 = \pm \omega L_2 \left(\frac{(L_2/2L_3)\sin 2\theta}{\sqrt{1 - (L_2^2/L_3^2)\sin^2 \theta}} + \sin \theta \right) \quad (39)$$

La relación $v/\omega = T$

$$\frac{v_4}{\omega} = T = \pm L_2 \left(\frac{(L_2/2L_3)\sin 2\theta}{\sqrt{1 - (L_2^2/L_3^2)\sin^2 \theta}} + \sin \theta \right) \quad (40)$$

Para el caso del algoritmo se parte por simplificar la mayor cantidad de Bonds, esto permitirá reducir la cantidad de ecuaciones, además de reducir la cantidad de variables que intervendrían de tomar todo el Bond Graph.

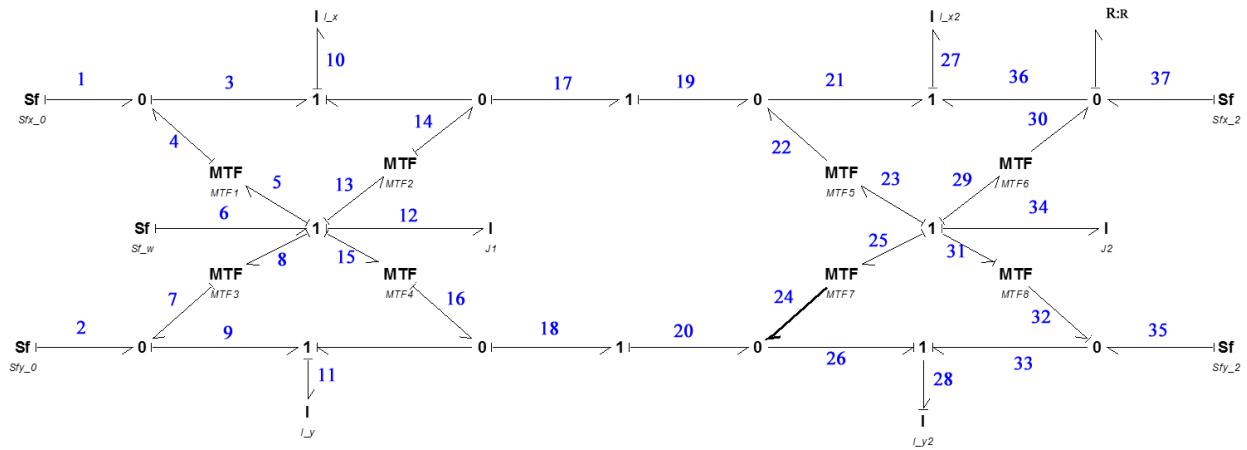


Figura 37 Numeración de Bonds.

Primero se eliminan los Bonds que tienen velocidad cero, estos corresponden a los que se encuentran en el extremo de la manivela Bond 1 y 2, acompañando a esta simplificación también se simplifican los Bonds de inercia del elemento manivela, ya que se puede asumir que el centro de masa del elemento se encuentra sobre el centro de rotación, simplificando de esta manera los Bonds 10 y 11, a la par de estas dos simplificaciones todos los Bonds involucrados en la trasmisión de potencia se simplifican, esto es los Bonds 3, 4, 5 y 7, 8, 9.

Otra simplificación que se puede realizar, es entre los nodos que tienen una entrada y una salida, correspondientes a los Bonds 14 (luego de la primera simplificación), 17 y 19, como también a los 16, 18, 20.

Una última simplificación se realiza en el elemento corredera donde su desplazamiento en Y es cero, por lo que el Bond 35 desaparece, y de la misma manera que en la manivela, la inercia de eje Y desaparece, por lo tanto, el Bond 28, acompañado del Bond 26, 31, 32, 32 serían simplificados por el mismo motivo que se explicó en la manivela.

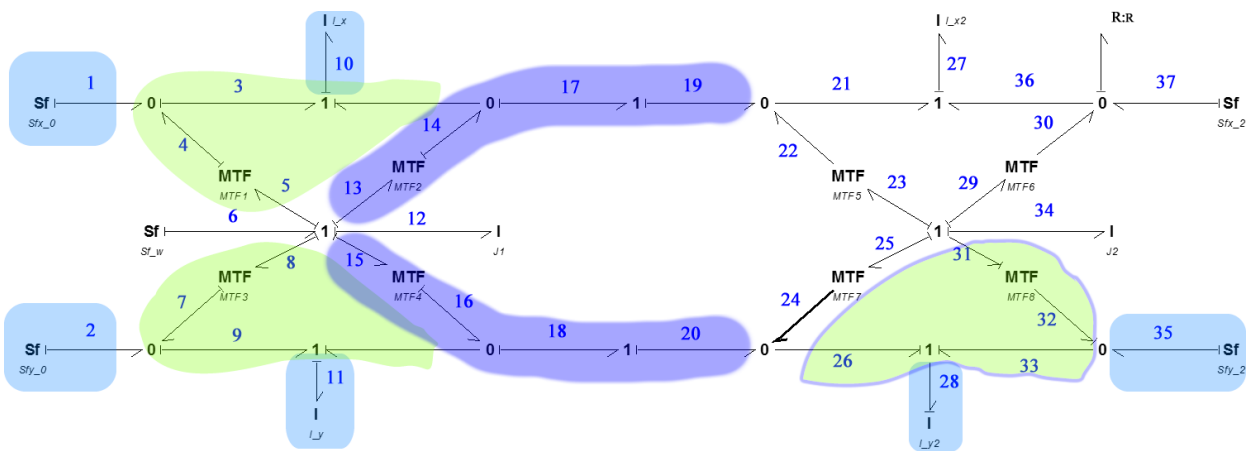


Figura 38 Simplificación de Bonds.

Renumerando los Bonds simplificados nos queda:

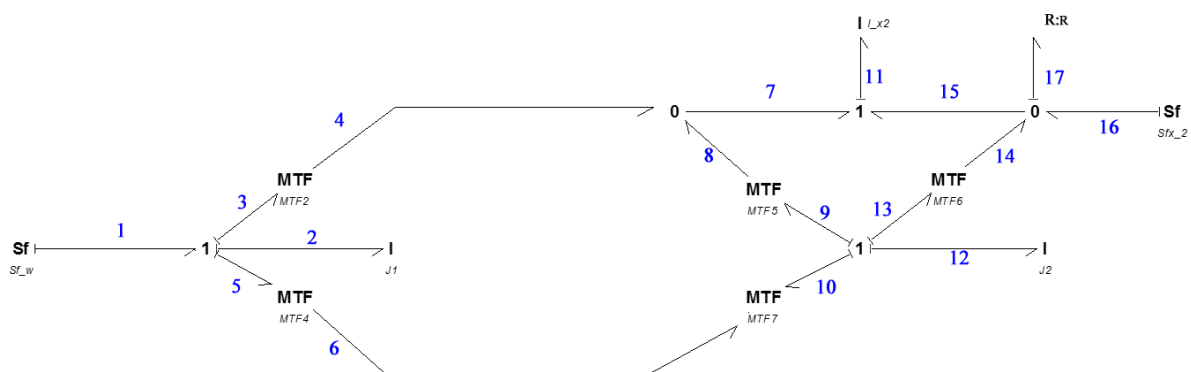


Figura 39 Simplificación y Numeración de Bonds.

El sistema de ecuaciones se obtiene de la misma manera que en el caso anterior. La tabla de ecuaciones resulta.

Tabla 8 Sistema de ecuaciones para el mecanismo

Grafo	f	e
1	$f_1 = f_2 = f_3 = f_5 = \omega$	$e_1 = M$
2	$f_2 = \frac{p_2}{J} = \omega$	$e_2 = \dot{p}_2 = J\dot{f}_2$
3	$f_3 = \omega_3$	$e_3 = e_1 - e_2 - e_5$
4	$f_4 = T_1 f_3$	$e_4 = \frac{1}{T_1} e_3$
5	$f_5 = \frac{1}{T_2} f_6$	$e_5 = T_2 e_6$
6	$f_6 = f_4 = f_8$	$e_4 = e_7 = e_8$
7	f_7	e_7
8	$f_8 = T_3 f_9$	$e_8 = \frac{1}{T_3} e_9$
9	$f_9 = f_{11} = f_{13} = f_{14}$	$e_{13} = e_9 + e_{11} + e_{14}$
10	$f_{10} = \frac{1}{T_4} f_6$	$e_{10} = T_4 e_6$
11	$p_{11} = m f_{11}$	$e_{11} = \dot{p}_{11} = m\dot{f}_{11}$
12	$p_{12} = m f_{12}$	$e_{11} = \dot{p}_{12} = m\dot{f}_{12}$
13	f_{13}	e_{13}
14	$f_{14} = T_5 f_{13}$	$e_{14} = \frac{1}{T_5} e_{13}$
15	$f_{17} = f_{14} + f_{15} + f_{16}$	$e_{14} = e_{15} = e_{16} = e_{17}$
16	f_{16}	$e_{16} = F$
17	$e_5 = T_2 e_6$	$e_{17} = r f_{17}$

Los transformadores modula relaciona la velocidad lineal con la velocidad angular que se puede obtener fácilmente del cálculo de sus velocidades angulares dando como resultado:

$$v1_x = v_x - \omega R1 \text{ sen}(\theta) \quad (41)$$

$$v1_y = v_y - \omega R1 \text{ cos}(\theta)$$

$$v1_x = v_x - \omega R2 \text{ sen}(\theta) \quad (42)$$

$$v1_y = v_y - \omega R2 \text{ cos}(\theta)$$

Resultando los transformadores

$$MTR2 \quad T1 = R1 \text{ sen}(\theta) \quad (43)$$

$$MTR4 \quad T2 = R1 \text{ cos}(\theta)$$

$$MTR5 \quad T3 = R2 \text{ sen}(\theta) \quad (44)$$

$$MTR7 \quad T4 = R2 \text{ cos}(\theta)$$

$$MTR6 \quad T5 = R2 \text{ sen}(\theta)$$

Por lo que el sistema de ecuaciones resultante después de sustituir las ecuaciones de esfuerzo quedaría formado por las siguientes ecuaciones:

$$\frac{1}{T_1} (M - J\dot{f}_2 - T_2 e_6) = \frac{1}{T_3} (T_5 e_{14} - m\dot{f}_{11} - m\dot{f}_{12}) \quad (45)$$

$$e_{10} = T_4 e_6 \quad (46)$$

$$\frac{1}{T_5} e_{13} = F = e_{17} = r f_{17} \quad (47)$$

$$e_7 = e_{11} + e_{15} \quad (48)$$

2.2.3 Resolución de la ecuación de estado

La resolución de ecuaciones de este estilo se pueden resolver de diferentes maneras, entre ellos se puede señalar a los métodos numéricos como los más apropiados ya que una solución exacta, en muchos de los casos causa gran inversión de tiempo.

Entre los métodos numéricos más utilizados está el método de Euler y el método de Runge-Kutta de cuarto orden.

Método de Euler, es una solución simple a las ecuaciones obtenidas por el método de Bond Graph, que consiste en obtener la pendiente de la ecuación diferencial en un tiempo instantáneo. [7 pg212]

$$(pendiente) = f(y_1, t) \quad (49)$$

Cuando la pendiente necesaria en la ecuación es calculada en el tiempo instantáneo y esta es multiplicada por el salto de tiempo, se obtiene el valor de y en el nuevo tiempo.

$$y_{i+1} = f(y_i, t_i)\Delta t + y_i \quad (50)$$

Este método genera un gran error ya que se basa en la pendiente de la curva, y cuando se analiza un salto de tiempo no adecuado, se pierde rápidamente precisión en el método. Para evitar esto se toma valores de Δt muy pequeños, pudiendo disminuir significativamente el error.

Método de Runge-Kutta, su forma discreta explícita es de multipasos, viene de la integración y toma una forma rígida "Stiff" que provoca una inestabilidad numérica a menos que el Δt sea muy pequeño.

Regla del trapecio o Runge Kutta de 2do orden

$$k_1 = f(y_i, t_i) \quad (51)$$

$$k_2 = f(y_i + k_1 \Delta t, t_i + \Delta t/2) \quad (52)$$

$$y_{i+1} = y_i + (k_1 + k_2)/2 \quad (53)$$

La solución de las ecuaciones diferenciales de 1er orden se la realiza mediante el software Python que se presenta en el capítulo 4.

Capítulo 3.

Comparación del Método de Bond Graph con otros métodos de resolución

A la hora de resolver mecanismos, el método de Bond Graph no es el más elegido, existen otros métodos que permiten obtener los mismos resultados, con mayores y menores ventajas.

3.1 Métodos alternativos de resolución

Entre los métodos de resolución cinemática y dinámica de mecanismos existen los métodos gráficos y los métodos analíticos. Los métodos gráficos se limitan al análisis cinemático y no permiten comparar valores con los otros métodos, ya que este análisis se realiza únicamente para posiciones fijas determinadas de sus elementos, para un análisis en función del tiempo requiere de graficar un sin número de veces dependiendo de la posición específica que se requiera analizar. Otros métodos para el análisis cinemático son: Newton-Raphson, coordenadas naturales y Raven.

Por otro lado, para el análisis dinámico de mecanismos, los más usados son los métodos de Newton-Euler y Lagrange. Método de Newton-Euler, cuyo análisis se basa en ecuaciones vectoriales basadas en el equilibrio dinámico de fuerzas y momentos. Mientras que la mecánica analítica Lagrangiana se basa en ecuaciones escalares basadas en funciones de estado relacionadas con la energía cinética y potencial en sistemas conservativos y energía de disipación o función de Rayleigh en sistemas no conservativos.

Para analizar mecanismos el método de Bond Graph ha quedado en segundo plano, por su simplicidad se usa el método del triángulo y también la mecánica Lagrangiana, con el fin de analizar por separado posición, velocidad, aceleración y fuerzas mediante la inserción de elementos que permitan calcular lo requerido.

3.1.1 Método analítico del triángulo

Usando el método del triángulo, el método implica la inserción de líneas de referencia dentro del mecanismo y el análisis de los triángulos restantes, se usan las leyes de los triángulos y rectángulos, para determinar las longitudes de los lados y la magnitud de los ángulos interiores.

El método analítico permite modificar y recalcular rápidamente la solución. Las ecuaciones descritas en el libro de Myska que controlan el movimiento para el caso del mecanismo de biela manivela analizado son: [2]

La construcción de una línea auxiliar produce dos triángulos como se ve en la figura 40.

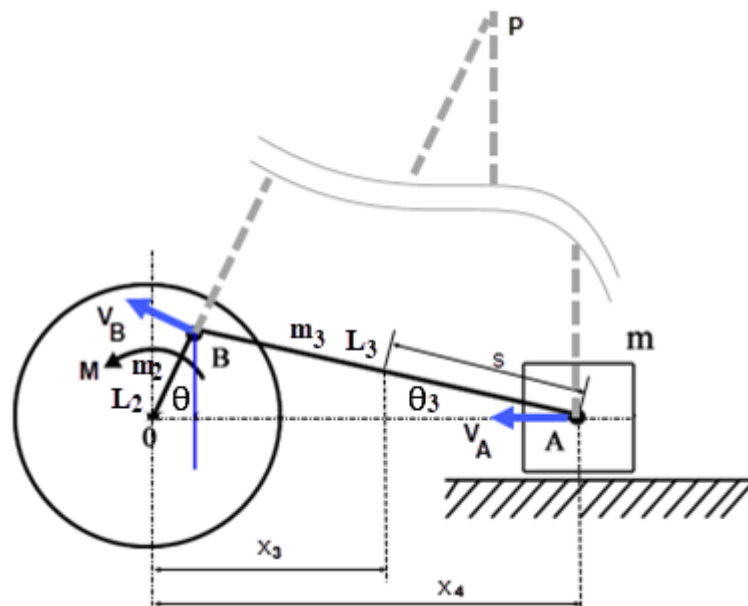


Figura 40 Contrucción de triángulos.

Las ecuaciones obtenidas para los diferentes casos son:

Se aplica el teorema del seno al $\Delta=OBA$.

$$\frac{\text{sen}(\theta_3)}{L_2} = \frac{\text{sen}(\theta)}{L_3}, \quad \theta_3 = \text{sen}^{-1} \left[\frac{L_2 \text{sen}(\theta)}{L_3} \right] \quad (54)$$

La distancia OA es representada por x_4 , se tiene.

$$x_4 = L_2 \cos(\theta) + L_3 \cos(\theta_3) \quad (55)$$

Desarrollando la raíz cuadrada en serie para $L_2 \leq L_3$, obtenemos la expresión anterior.

$$L_3 \sin \theta_3 = L_2 \sin \theta \quad (56)$$

$$x_4 = L_2 \cos \theta + L_3 \cos \theta_3 = L_2 \cos \theta + \sqrt{L_3^2 - L_2^2 \sin^2 \theta} \quad (57)$$

En el mecanismo de manivela corredera está definido únicamente por las dimensiones L2, L3. Conociendo θ , $\dot{\theta} = \omega$. Si derivamos x_4 .

$$v_4 = -\dot{\theta}L_2 \sin \theta - \frac{\dot{\theta}L_2^2 \sin \theta \cos \theta}{\sqrt{L_3^2 - L_2^2 \sin^2 \theta}} \quad (58)$$

$$v_4 = -\omega L_2 \left(\frac{(L_2/2L_3)\sin 2\theta}{\sqrt{1 - (L_2^2/L_3^2) \sin^2 \theta}} + \sin \theta \right) \quad (59)$$

$$a_4 = -\ddot{\theta}L_2 \left(\frac{(L_2/2L_3)\sin 2\theta}{\sqrt{1 - (L_2^2/L_3^2) \sin^2 \theta}} + \sin \theta \right) - \dot{\theta}^2 L_2 \left(\frac{(L_2/L_3)\cos 2\theta}{\sqrt{1 - (L_2^2/L_3^2) \sin^2 \theta}} + \frac{(L_2/L_3)^2 \sin 2\theta}{2\sqrt{(1 - (L_2^2/L_3^2) \sin^2 \theta)^3}} + \cos \theta \right) \quad (60)$$

El resultado del cálculo de estas ecuaciones se obtiene en Python tomando como datos de entrada en la manivela $\theta = \text{sen}(t)$ para la posición angular, derivando la función en el tiempo se obtiene, $\omega = \text{cos}(t)$ para la velocidad angular y $\alpha = -\text{sen}(t)$.

En la figura 40 se construye el centro instantáneo de rotación de la biela (punto P). Se aplica el teorema del seno al $\triangle BPA$.

$$\frac{PB}{\sin \angle BAP} = \frac{PA}{\sin \angle PBA} \quad (61)$$

$$\frac{PB}{PA} = \frac{\sin \angle BAP}{\sin \angle PBA} = \frac{\sin(\theta_3 - \theta)}{\sin(90 - \theta_3)}$$

$$\frac{PB}{PA} = \frac{\sin \theta_3 \cos \theta + \cos \theta_3 \sin \theta}{\cos \theta_3} = \frac{\sin \theta_3}{\cos \theta_3} \cos \theta + \sin \theta \quad (62)$$

$$v_4 = \omega L_2 \frac{PB}{PA} = \omega L_2 \left(\frac{(L_2/L_3) \sin \theta \cos \theta}{\sqrt{1 - (L_2/L_3)^2 \sin^2 \theta}} + \sin \theta \right) \quad (63)$$

La ecuación 59 y 63 tienen el mismo comportamiento de velocidad para la corredera.

El siguiente código se puede ejecutar en Python.

```
#Importación de Librerías
from sympy import *
from pylab import *
import math

tsim=float(input('Ingrese el tiempo de simulación [s] : ')) # 10 seg
n=int(input('Ingrese número de discretizaciones : ')) # 100

#PRE PROCESO
#Datos geométricos
L2=1; #[m] Longitud manivela
L3=2; #[m] Longitud biela

#PROCESOS
dt=float(tsim/(n-1)); #[s] Paso temporal

#Definir vectores vacíos
ti = [0.0]*(n+3); #tiempo discreto
O = [0.0]*(n+3); #desplazamiento angular manivela discreto
w = [0.0]*(n+1); #velocidad angular discreta manivela
a = [0.0]*(n); #aceleración angular discreta manivela
x4 = [0.0]*(n+3); #desplazamiento lineal discreta corredera
v4 = [0.0]*(n+1); #velocidad lineal discreta corredera
vA = [0.0]*(n+1); #velocidad lineal discreta corredera
a4 = [0.0]*(n); #aceleración lineal discreta corredera
a4_n= [0.0]*(n); #aceleración lineal discreta corredera termino 1
a4_t= [0.0]*(n); #aceleración lineal discreta corredera termino 2

#Condiciones iniciales para la manivela
O[0]=sin(ti[0]);
w[0]=cos(ti[0]); #Flujo de entrada

for i in range(0,n+2,1):
    ti[i+1]=ti[i]+dt; #Nodos temporales ti: t1,t2,...,tn,tn+1
```

```
#Aproximación de primera derivada por diferencias finitas
```

```
for i in range(0,n+3,1):
```

```
    O[i]=sin(ti[i]);
```

```
    x4[i]=L2*cos(O[i])+(L3**2-(L2*sin(O[i]))**2)**(1/2);
```

```
for i in range(0,n+1,1):
```

```
    w[i]=(O[i+1]-O[i])/dt; #Forma discreta de Euler
```

```
    v4[i]=w[i]*L2*(0.5*(L2/L3)*sin(2*O[i])/(1-((L2/L3)*sin(O[i]))**2)**(1/2)+sin(O[i]));
```

```
for i in range(0,n,1):
```

```
    a4_t[i]=a[i]*L2*(0.5*(L2/L3)*sin(2*O[i])/(1-((L2/L3)*sin(O[i]))**2)**(1/2)+sin(O[i]));
```

```
    a4_n[i]=(w[i])**2*L2*((L2/L3)*cos(2*O[i])/(1-((L2/L3)*sin(O[i]))**2)**(1/2)
```

```
+0.5*(L2/L3)**2*sin(2*O[i])/(1-((L2/L3)*sin(O[i]))**2)**(3/2)+cos(O[i]));
```

```
    a4[i]=a4_t[i]+a4_n[i];
```

```
#POST PROCESOS
```

```
subplot(3,1,1)
```

```
plot(ti[0:n],x4[0:n])
```

```
ylabel('Desplazamiento [m]')
```

```
title('Movimiento lineal corredera')
```

```
grid(linestyle='dotted')
```

```
subplot(3,1,2)
```

```
plot(ti[0:n],v4[0:n])
```

```
ylabel('Velocidad [m/s]')
```

```
grid(linestyle='dotted')
```

```
subplot(3,1,3)
```

```
plot(ti[0:n],a4[0:n])
```

```
xlabel('Tiempo [s]')
```

```
ylabel('Aceleración [m/s^2]')
```

```
grid(linestyle='dotted')
```

```
show()
```

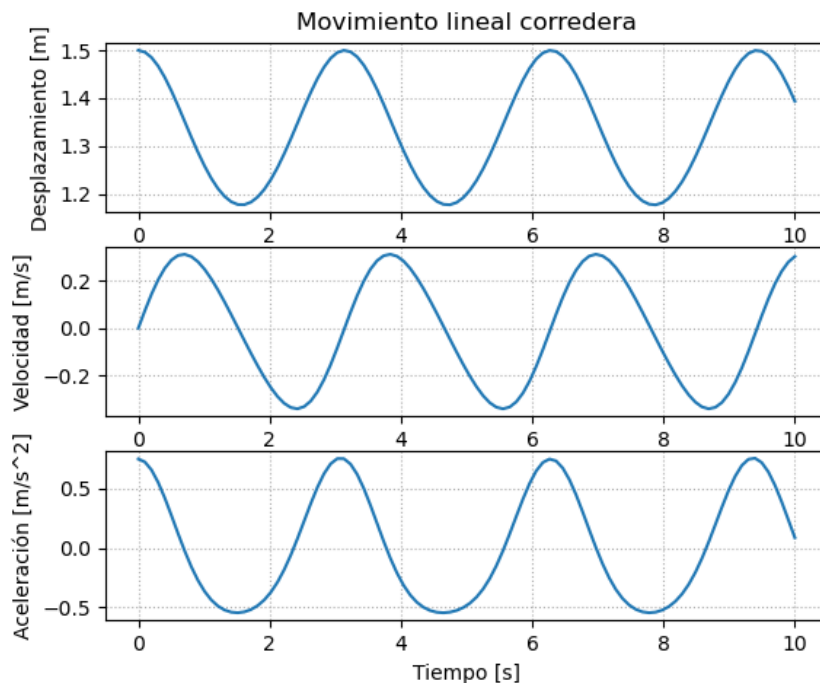


Figura 41 Cinemática de la corredera.

El comportamiento de la corredera depende de la entrada $\theta = \text{sen}(t)$ en la manivela, y se mantiene con un movimiento armónico.

3.1.2 Método analítico de Lagrange

Ahora cambiaremos la entrada, con el análisis de Lagrange. La ecuación (64) de Lagrange plantea un análisis dinámico multi-físico o sistema no conservativo.

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}} \right) - \left(\frac{\partial L}{\partial q} \right) = \frac{\partial \tau}{\partial \dot{q}} \quad (64)$$

Donde:

q: Coordenada generalizada

L: Lagrangiano

τ : Función de disipación de Rayleigh

El Lagrangiano se define: $L = T - V$, por la diferencia de la energía cinética T y la energía potencial V, y la función de Rayleigh es el efecto de la energía motriz y la energía de disipación $\tau = F \cdot \dot{q} - \frac{1}{2} \sum f r \cdot (\dot{q})^2$.

Para describir la ecuación de movimiento del mecanismo biela-manivela se emplea la ecuación de Lagrange como sistema conservativo. El sistema tiene tres puntos de flujo de velocidad θ , θ_3 , x_4 , correspondientes a la manivela, biela y corredera respectivamente, si un flujo debe ser conocido o entrada y otro relativo, de manera que el sistema presenta solo un grado de libertad. Entonces se toma como coordenada generalizada el ángulo θ , de modo que:

$$\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{\theta}} \right) - \left(\frac{\partial T}{\partial \theta} \right) = Q \quad (65)$$

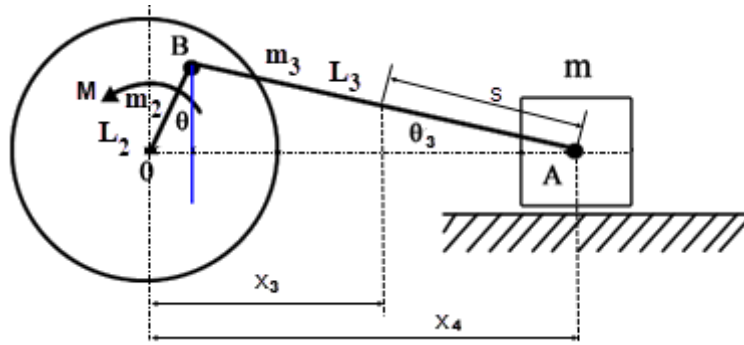


Figura 42 Esquema biela-manivela.

Donde T es la energía cinética del sistema y Q es su fuerza generalizada. [10]

$$T = T_{OB}^{manivela} + T_{BA}^{biela} + T_{OA}^{corredera} \quad (66)$$

Manivela: esta gira alrededor de un eje fijo, donde $J_2 = \frac{m_2 L_2^2}{3}$. Por tanto,

$$T_{OB} = \frac{1}{2} J_2 \omega^2 = \frac{1}{2} J_2 \dot{\theta}^2 \quad (67)$$

Biela: esta realiza un movimiento plano con velocidad v_3 del centro de masa de la biela y su velocidad angular $\omega_3 = \dot{\theta}_3$. Por tanto,

$$T_{BA} = \frac{1}{2} m_3 v_3^2 + \frac{1}{2} J_3 \omega_3^2 \quad (68)$$

$$v_3^2 = \dot{x}_3^2 + \dot{y}_3^2$$

$$x_3 = x_4 - s \cos \theta_3 \approx L_3 - s + L_2 \cos \theta$$

$$\dot{x}_3 = -L_2 \dot{\theta} \sin \theta$$

$$\frac{y_3}{L_2 \sin \theta} = \frac{s}{L_3}, \Rightarrow y_3 = \frac{s}{L_3} L_2 \sin \theta$$

$$\dot{y}_3 = \frac{s}{L_3} L_2 \dot{\theta} \cos \theta$$

$$v_3^2 = L_2^2 \left(\sin^2 \theta + \frac{s^2}{L_3^2} \cos^2 \theta \right) \dot{\theta}^2$$

$$\frac{s}{L_3} L_2 \sin \theta = s \cdot \sin \theta_3 \approx s \theta_3$$

$$\dot{\theta}_3 = \frac{L_2}{L_3} \dot{\theta} \cos \theta$$

De modo que,

$$T_{BA} = \frac{1}{2} m_3 L_2^2 \left(\sin^2 \theta + \frac{s^2}{L_3^2} \cos^2 \theta \right) \dot{\theta}^2 + \frac{1}{2} J_3 \frac{L_2^2}{L_3^2} \dot{\theta}^2 \cos^2 \theta \quad (69)$$

Corredera: la corredera se mueve por una recta. Por tanto,

$$T_{OA} = \frac{1}{2} m v_4^2 \quad (70)$$

Donde v_4 es la velocidad de la corredera, $v_4 = \dot{x}_4$

De este modo

$$x_4 = L_2 \cos \theta + L_3 \cos \theta_3 \approx L_2 \cos \theta + L_3$$

Y, por tanto,

$$v_4 = -L_2 \dot{\theta} \sin \theta$$

$$T_{OA} = \frac{1}{2} mL_2^2 \dot{\theta}^2 \sin^2 \theta \quad (71)$$

Como resultado se tiene la energía cinética del sistema

$$T = \frac{1}{2} J_2 \dot{\theta}^2 + \frac{1}{2} m_3 L_2^2 \left(\sin^2 \theta + \frac{s^2}{L_3^2} \cos^2 \theta \right) \dot{\theta}^2 + \frac{1}{2} J_3 \frac{L_2^2}{L_3^2} \dot{\theta}^2 \cos^2 \theta + \frac{1}{2} mL_2^2 \dot{\theta}^2 \sin^2 \theta \quad (72)$$

Resolviendo los términos de la ecuación de Lagrange

$$\begin{aligned} \frac{\partial T}{\partial \dot{\theta}} &= \left[(m + m_3) L_2^2 \sin^2 \theta + (J_3 + m_3 s^2) \frac{L_2^2}{L_3^2} \cos^2 \theta + J_2 \right] \dot{\theta} \\ \frac{d}{dt} \left(\frac{\partial T}{\partial \dot{\theta}} \right) &= \left[(m + m_3) L_2^2 \sin^2 \theta + (J_3 + m_3 s^2) \frac{L_2^2}{L_3^2} \cos^2 \theta + J_2 \right] \ddot{\theta} \\ &\quad + 2 \left[(m + m_3) L_2^2 - (J_3 + m_3 s^2) \frac{L_2^2}{L_3^2} \right] \dot{\theta}^2 \sin \theta \cos \theta \\ \frac{\partial T}{\partial \theta} &= \left[(m + m_3) L_2^2 - (J_3 + m_3 s^2) \frac{L_2^2}{L_3^2} \right] \dot{\theta}^2 \sin \theta \cos \theta \end{aligned}$$

La fuerza generalizada del sistema se obtiene calculando el trabajo δW realizado en el desplazamiento virtual $\delta \theta$ por la fuerza de presión que actúa la corredera y el momento de resistencia M:

$$\delta W = F \cdot \delta x_4 - M \cdot \delta \theta = F \cdot L_2 \cdot \delta \theta \cdot \sin \theta - M \cdot \delta \theta$$

Entonces,

$$Q = \frac{\delta W}{\delta \theta} = F \cdot L_2 \cdot \sin \theta - M$$

Sustituyendo en la ecuación de Lagrange, se tiene:

$$\begin{aligned} & \left[(m + m_3)L_2^2 \sin^2 \theta + (J_3 + m_3 s^2) \frac{L_2^2}{L_3^2} \cos^2 \theta + J_2 \right] \ddot{\theta} \\ & + \left[(m + m_3)L_2^2 - (J_3 + m_3 s^2) \frac{L_2^2}{L_3^2} \right] \dot{\theta}^2 \sin \theta \cos \theta = F \cdot L_2 \cdot \sin \theta - M \end{aligned} \quad (73)$$

Discretizando para Euler, se tiene:

$$\begin{aligned} & \begin{bmatrix} \frac{d\theta}{dt} \\ \frac{d\dot{\theta}}{dt} \end{bmatrix} = \begin{bmatrix} \dot{\theta} \\ \frac{F \cdot L_2 \cdot \sin \theta - M - \left[(m + m_3)L_2^2 - (J_3 + m_3 s^2) \frac{L_2^2}{L_3^2} \right] \dot{\theta}^2 \sin \theta \cos \theta}{\left[(m + m_3)L_2^2 \sin^2 \theta + (J_3 + m_3 s^2) \frac{L_2^2}{L_3^2} \cos^2 \theta + J_2 \right]} \end{bmatrix} \\ & \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix}_{i+1} = \begin{bmatrix} \dot{\theta} \\ \frac{F \cdot L_2 \cdot \sin \theta - M - \left[(m + m_3)L_2^2 - (J_3 + m_3 s^2) \frac{L_2^2}{L_3^2} \right] \dot{\theta}^2 \sin \theta \cos \theta}{\left[(m + m_3)L_2^2 \sin^2 \theta + (J_3 + m_3 s^2) \frac{L_2^2}{L_3^2} \cos^2 \theta + J_2 \right]} \end{bmatrix} dt + \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix}_i \end{aligned} \quad (74)$$

El siguiente código se puede ejecutar en Python.

#Importación de Librerías

from sympy import *

from pylab import *

tsim=float(input('Ingrese el tiempo de simulación [s] : '))

n=int(input('Ingrese número de discretizaciones : '))

#PRE PROCESO

#Datos geométricos

L2 = 1; #[m] Longitud manivela

L3 = 2; #[m] Longitud biela

s = L3/2; #[m] Centro de masa de la biela respecto a la corredera

#Datos parámetros pasivos

m = 1.0; #[kg] Masa corredera

k2 = 0.0; #[N/m] Rigidez manivela

k3 = 0.0; #[N/m] Rigidez biela

r = 0.0; #[Ns/m] fricción corredera

```

m2 = 1.0;      #[kg]  Masa manivela
m3 = 1.0;      #[kg]  Masa biela
J2 = m2*L2**2/3; #      Inercia manivela
J3 = m3*L3**2/3; #      Inercia biela

```

#Cargas externas

```

F=0.0;      #[N]  Fuerza externa aplicada en la corredera
M=0.0;      #[N.m] Momento en la manivela

```

#PROCESOS

```

dt=float(tsim/(n-1)); #[s] Paso temporal

```

#Definir vectores vacíos

```

ti = [0.0]*(n+3); #tiempo discreto
O = [0.0]*(n+3); #desplazamiento angular discreto manivela
w = [0.0]*(n+1); #velocidad angular discreta manivela
a = [0.0]*(n+1); #aceleración angular discreta manivela
x4 = [0.0]*(n+3); #desplazamiento lineal discreta corredera
v4 = [0.0]*(n+1); #velocidad lineal discreta corredera
a4 = [0.0]*(n); #aceleración lineal discreta corredera
a4_n = [0.0]*(n); #aceleración lineal discreta corredera
a4_t = [0.0]*(n); #aceleración lineal discreta corredera
N = [0.0]*(n+3); #Variable auxiliar numerador
D = [0.0]*(n+3); #Variable auxiliar denominador

```

#Condiciones iniciales para la manivela

```

O[0]=sin(ti[0]);
w[0]=cos(ti[0]); #Flujo de entrada

```

```

for i in range(0,n+2,1):
    ti[i+1]=ti[i]+dt; #Nodos temporales ti: t1,t2,...tn,tn+1

```

#Aproximación de primera derivada por diferencias finitas

```

for i in range(0,n,1):
    O[i+1]=w[i]*dt+sin(ti[i]); #O[i+1]=w[i]*dt+O[i], donde O[i]=sin(ti[i]) Forma discreta Euler
    N[i]=F*L2*sin(O[i])-M-((m+m3)*L2**2-(J3+m3*s**2)*(L2/L3)**2)*(w[i])**2*sin(O[i])*cos(O[i]);
#Numerador
    D[i]=((m+m3)*L2**2*(sin(O[i]))**2+(J3+m3*s**2)*(L2*cos(O[i])/L3)**2+J2); #Denominador
    w[i+1]=(N[i]/D[i])*dt+w[i]; #Forma discreta Euler

```

```

for i in range(0,n,1):
    x4[i]=L2*cos(O[i])+(L3**2-(L2*sin(O[i]))**2)**(1/2);
    v4[i]=w[i]*L2*(0.5*(L2/L3)*sin(2*O[i])/(1-((L2/L3)*sin(O[i]))**2)**(1/2)+sin(O[i]));
    a[i]=N[i]/D[i];
    a4_t[i]=a[i]*L2*(0.5*(L2/L3)*sin(2*O[i])/(1-((L2/L3)*sin(O[i]))**2)**(1/2)+sin(O[i]));
    a4_n[i]=(w[i])**2*L2*((L2/L3)*cos(2*O[i])/(1-
    ((L2/L3)*sin(O[i]))**2)**(1/2)+0.5*(L2/L3)**2*sin(2*O[i])/(1-((L2/L3)*sin(O[i]))**2)**(3/2)+cos(O[i]));
    a4[i]=a4_t[i]+a4_n[i];

```

#POST PROCESOS

```

figure(1)
subplot(3,1,1)
plot(ti[0:n],x4[0:n])
ylabel('Desplazamiento [m]')
title('Movimiento lineal corredera')
grid(linestyle='dotted')
subplot(3,1,2)
plot(ti[0:n],v4[0:n])

```

```
ylabel('Velocidad [m/s]')
grid(linestyle='dotted')
subplot(3,1,3)
plot(ti[0:n],a4[0:n])
xlabel('Tiempo [s]')
ylabel('Aceleración [m/s^2]')
grid(linestyle='dotted')
```

#SALIDA DE LA MANIVELA

```
figure(2)
subplot(3,1,1)
plot(ti[0:n],O[0:n])
ylabel('Desplazamiento [rad]')
title('Movimiento angular manivela')
grid(linestyle='dotted')
subplot(3,1,2)
plot(ti[0:n],w[0:n])
ylabel('Velocidad [rad/s]')
grid(linestyle='dotted')
subplot(3,1,3)
plot(ti[0:n],a[0:n])
xlabel('Tiempo [s]')
ylabel('Aceleración [rad/s^2]')
grid(linestyle='dotted')
show()
```

Si consideramos la forma discreta en la posición angular, tenemos la figura 43 y 44.
 $[\theta]_{i+1} = [\omega]_i dt + [\sin(t)]_i$

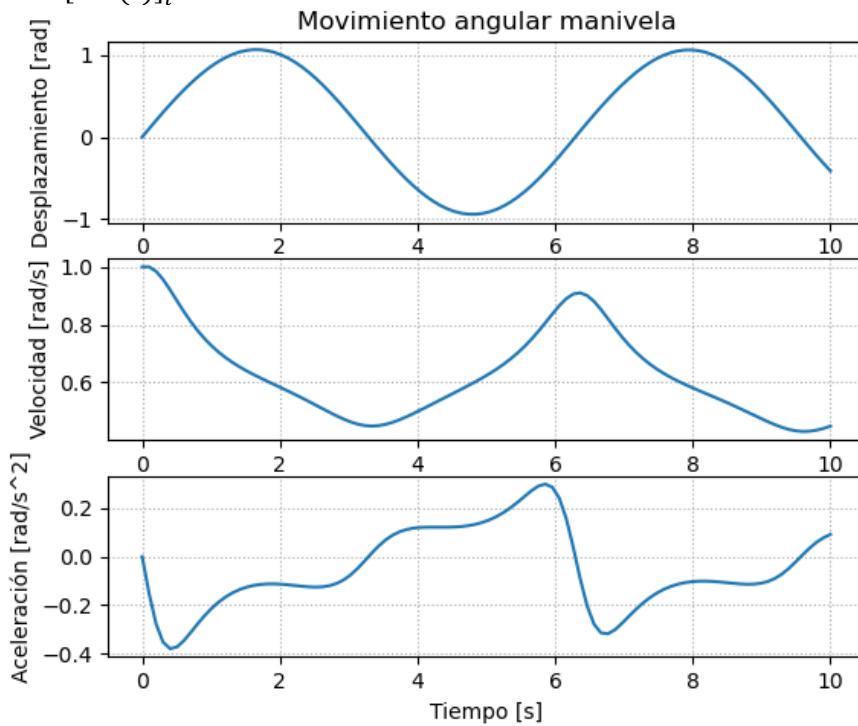


Figura 43 Cinemática de la Manivela.

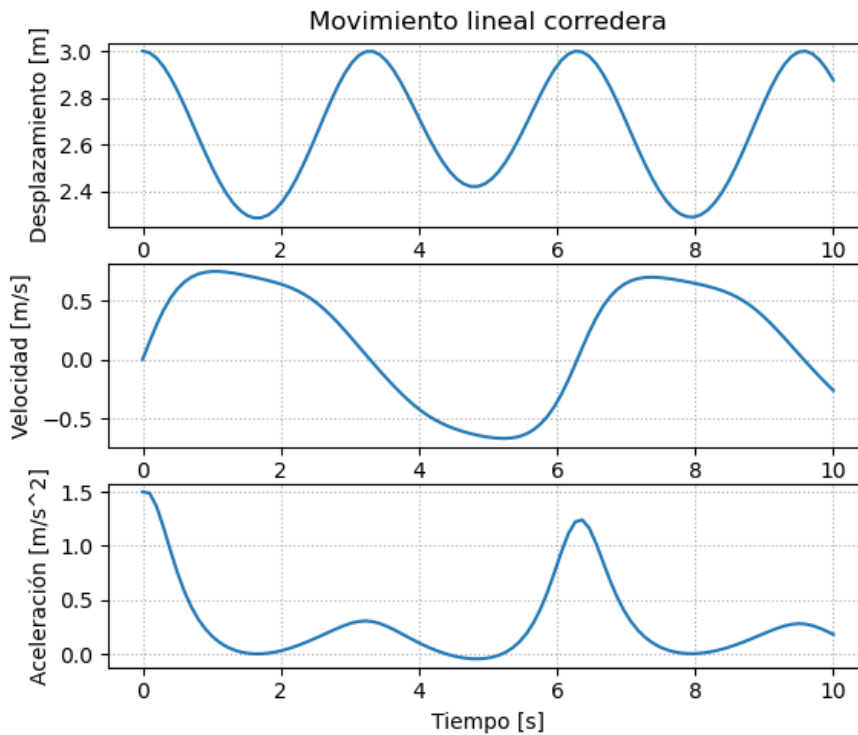


Figura 44 Cinemática de la Corredera.

Si modificamos la entrada en la forma discreta de la posición angular de la manivela, tenemos la figura 45 y 46. $[\theta]_{i+1} = [\omega]_i dt + [\theta]_i$

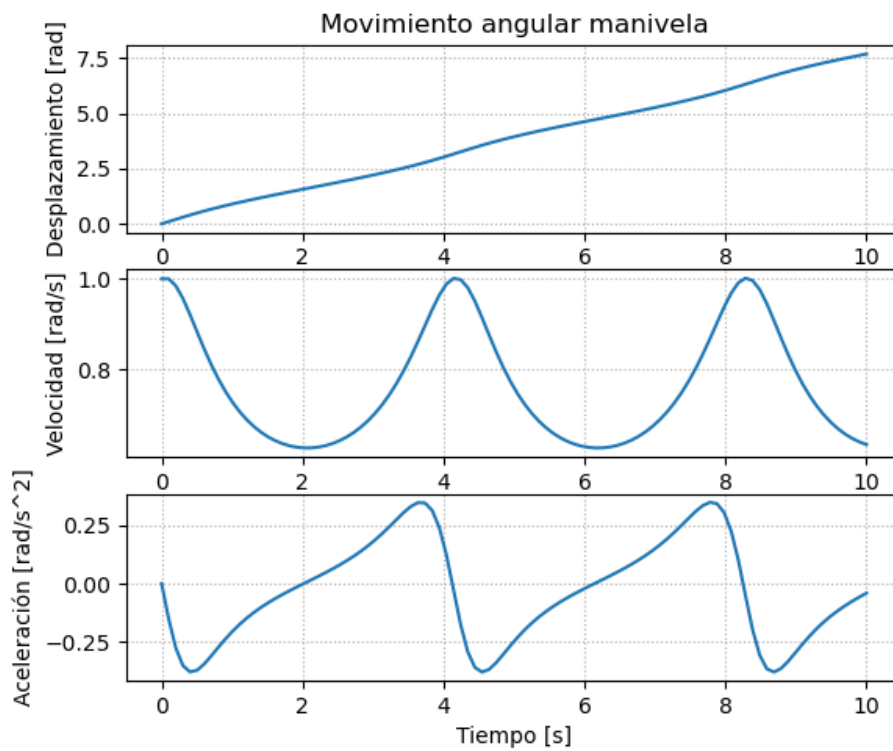


Figura 45 Cinemática de la Manivela.

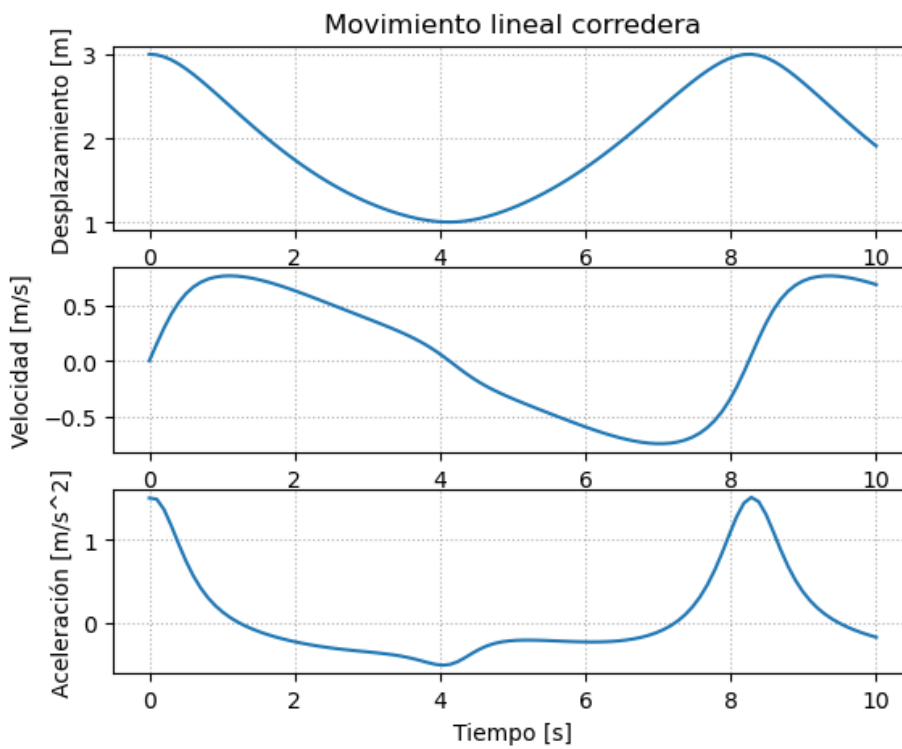


Figura 46 Cinemática de la Corredera.

3.2 Comparación de Métodos

Entre el método analítico del triángulo y el método de Bond Graph existe un sinnúmero de diferencias que se detallan a continuación:

El método de Bond Graph genera ecuaciones diferenciales de primer orden que integradas pueden crear una de segundo orden, a diferencia de que en método del triángulo crea ecuaciones algebraicas.

El método de Bond Graph crea una ecuación o sistema de ecuaciones diferenciales a resolver para determinar las variables requeridas, el método analítico del triángulo crea una ecuación para cada variable, lo que incurre en un análisis y planteamiento de ecuaciones para cada caso.

El método analítico requiere de menos conocimientos de cálculo que el método de Bond Graph debido a las ecuaciones diferenciales que se deben resolver.

El método de Bond Graph es un método que analiza dinámicamente un sistema o mecanismo, mientras que el método analítico construye líneas en posiciones determinadas para escribir las ecuaciones dinámicas.

El método de Bond Graph requiere de un dominio cabal de la dinámica y en otros casos de toda la mecánica, el método analítico en algunos casos puede satisfacerse solo con conocimientos trigonométricos.

Para obtener factores de transformador y girador, el método de Bond Graph utiliza métodos analítico, en algunos casos hasta llegando a resolver el mecanismo antes de plantear el sistema de ecuaciones del Bond Graph.

La integración de más variables sobre el sistema se hace de manera simple en el Bond Graph añadiendo Bonds y replanteando el sistema de ecuaciones, en el método analítico esta integración se la realiza reformulando completamente el análisis por lo que se consideraría como un mecanismo completamente distinto.

Con la integración de nuevos elementos el Bond Graph no cambia, este se amplía y abarca a los nuevos elementos, en el caso del método analítico el sistema cambia drásticamente lo que hace comenzar todo nuevamente.

Capítulo 4.

Desarrollo del algoritmo en Software Computacional

4.1 Lenguaje de programación en PYTHON

Python es un lenguaje de programación de alto nivel, que permite trabajar de forma rápida e integral con mayor eficacia, fue creado por Guido van Rossum a finales de los ochenta, Python fue creado para resolver problemas reales y tiene una amplia gama de características que se encuentran en lenguajes de programación como C ++, Java, Modula-3. Las características de este lenguaje se pueden describir a continuación:

Python es un lenguaje muy simple de programación, por lo que es muy fácil iniciarse en este lenguaje. El pseudo-código natural de Python es una de sus grandes cualidades. No requiere de líneas de programación excesivas para crear un programa.

Debido a la naturaleza Open Source en Python; es modificable para que pueda funcionar en diversas plataformas (Linux, Windows, Macintosh, y muchos otros). Al ser Open Source es gratuito y existen gran cantidad de persona colaboradoras que aportan cada vez al crecimiento del programa, lo que conlleva aportes como librerías.

Al programar en Python no debemos preocuparnos por detalles de bajo nivel, como manejar la memoria empleada por el programa, una de las principales características del programa es que las variables se definen, siendo la principal diferencia con otros lenguajes de programación.

Python contiene un gran número de librerías, tipos de datos y funciones integradas en el propio lenguaje, que ayudan a realizar muchas tareas comunes sin necesidad de tener que programarlas desde cero. Estas librerías ahorran mucho tiempo de programación, permitiendo al programador dedicarse a desarrollar mejores programas.

Python tiene una sintaxis visual agradable, esto se traduce a arreglos de escritura establecido que son de carácter obligatorio ofreciendo mucho mayor estructura además de soporte para programas grandes. Por otro lado, Python ofrece mucho más chequeo de error que C, y siendo un lenguaje de muy alto nivel, tiene tipos de datos de alto nivel incorporados como arreglos de tamaño flexible y diccionarios.

Python viene con una gran colección de módulos estándar que se usan como base de tus programas y el resto se obtiene de librerías asociadas, instaladas dentro del programa.

Python permite escribir programas compactos y legibles. Los programas son típicamente más cortos que sus equivalentes, Principalmente utilizado como calculadora, los tipos de datos de alto nivel permiten expresar operaciones complejas con una sola instrucción además de que la agrupación de instrucciones se hace mediante sangrías en vez de llaves de apertura y cierre.

4.2 Desarrollo del algoritmo

Para el desarrollo del algoritmo se utilizan las librerías:

Pylab. Un Laboratorio de matemáticas y cálculo que permite arreglos y funciones especiales de matemáticas trigonometría y cálculo.

Matplotlib. Es una librería que permite graficar entre otras cosas, funciones.

Sympy. Una librería que permite trabajar a las ecuaciones matemáticas de manera simbólica.

El detalle de cómo se integran estas librerías a Python se encuentran detalladas en el Anexo 2

El procedimiento para la obtención del algoritmo es el siguiente:

El código de programación para la obtención resultante es:

```
#Importación de Librerías
from sympy import *
from pylab import *

tsim=float(input('Ingrese el tiempo de simulación [s] : '))
n=int(input('Ingrese número de discretizaciones : '))

#PRE PROCESO
#Datos geométricos
L2=1;    #[m] Longitud manivela
L3=2;    #[m] Longitud biela
#Datos parámetros pasivos
m = 1.0;    #[kg] Masa corredera
k = 0.0;    #[N/m] Rigidez corredera
r = 0.0;    #[Ns/m] fricción corredera
m2 = 1.0;   #[kg] Masa manivela
J = m2*L2**2/3; # Inercia manivela
g =9.81;    #[m/s^2] Gravedad

#Cargas externas
F=0;     #[N] Fuerza externa
M=0;     #[N.m] Momento

#PROCESOS
dt=float(tsim/(n-1)); #[s] Paso temporal

#Definir vectores vacios
ti = [0.0]*(n+3); #tiempo discreto
Om = [0.0]*(n+3); #desplazamiento angular discreto manivela
wm = [0.0]*(n+1); #velocidad angular discreta manivela
am = [0.0]*(n);   #aceleración angular discreta manivela
T = [0.0]*(n);   #Transformador discreto
x = [0.0]*(n+1); #desplazamiento discreto corredera
v = [0.0]*(n+1); #velocidad discreta corredera
v4 = [0.0]*(n+1); #velocidad discreta corredera
#Flujo de entrada
Om[0]=sin(ti[0]);
#Condiciones iniciales para la manivela
x[0]=0;
v[0]=0;

for i in range(0,n+2,1):
    ti[i+1]=ti[i]+dt; #Nodos temporales ti: t1,t2,...tn,tn+1

#Aproximación de primera derivada por diferencias finitas
for i in range(0,n+3,1):
    Om[i]=sin(ti[i]);

for i in range(0,n+1,1):
    wm[i]=(Om[i+1]-Om[i])/dt;

for i in range(0,n,1):
    am[i]=(wm[i+1]-wm[i])/dt;

for i in range(1,n,1):
    # Ecuaciones de estado
```

```

T[i]=L2*(0.5*(L2/L3)*sin(2*Om[i])/(1-((L2/L3)*sin(Om[i]))**2)**(1/2)+sin(Om[i]));
MT=[(T[i])*wm[i],((M-J*am[i])/(abs(T[i]))-r*v[i]-F)/m)];
# Forma discreta de Euler
x[i+1]= x[i]+MT[0]*dt; #x = x + (dt)*(T*w)
v[i+1]= v[i]+MT[1]*dt; #v = v + (dt)*((M-J*alf)/abs(T)-r*v-F)/m
v4[i+1]= (T[i])* wm[i];

```

#POST PROCESOS

```

subplot(3,1,1)
plot(ti[0:n],x[0:n])
ylabel('Desplazamiento [m]')
title('Movimiento corredera')
grid(linestyle='dotted')

```

```

subplot(3,1,2)
plot(ti[0:n],v4[0:n])
xlabel('Tiempo [s]')
ylabel('Velocidad [m/s]')
text(2,0,'Vía Transformador')
grid(linestyle='dotted')

```

```

subplot(3,1,3)
plot(ti[0:n],v[0:n])
xlabel('Tiempo [s]')
ylabel('Velocidad [m/s]')
text(2,0.5,'Vía Dinámica')
grid(linestyle='dotted')
show()

```

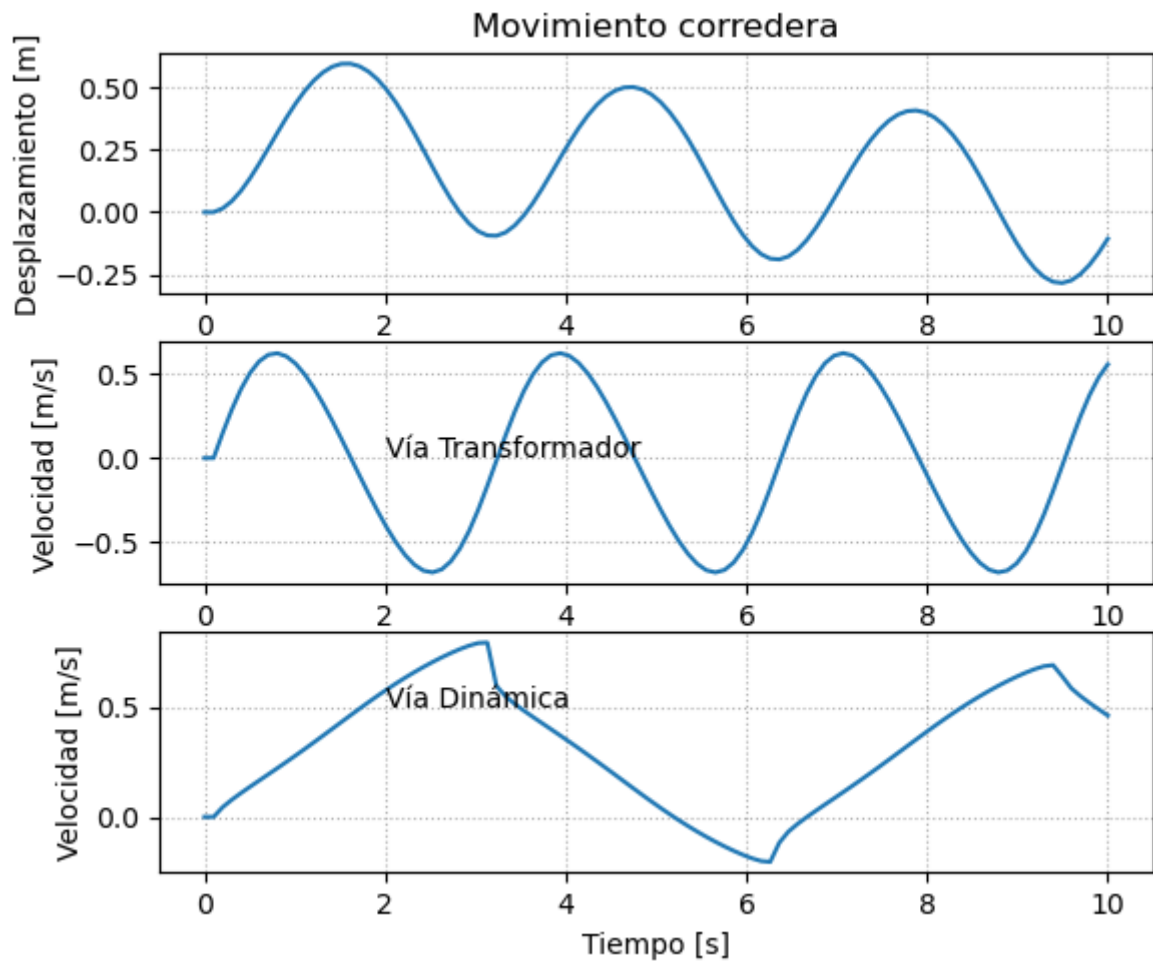


Figura 47 Movimiento descrito por Bond Graph.

Este código de programación representado en un diagrama de flujo se presenta a continuación:

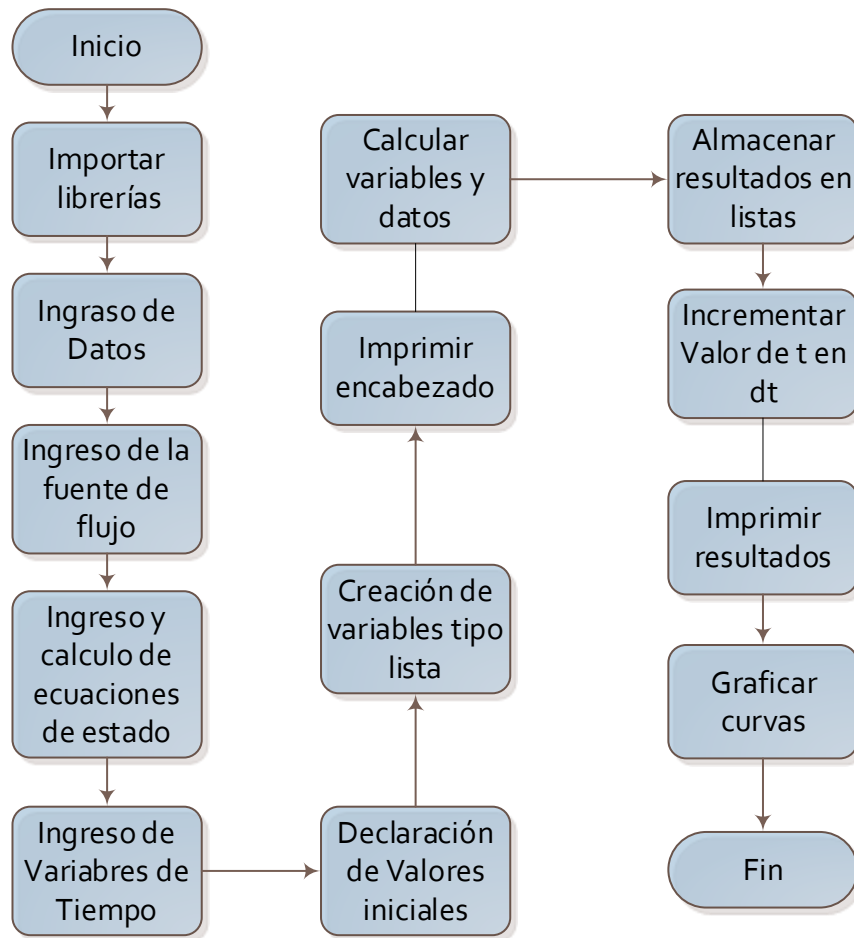


Figura 48 Diagrama de Flujo del código de Python.

4.3 Simulación del modelo

Mediante este código podemos obtener los valores de todas las variables que se observan en la figura 48.

4.4 Algoritmo genérico para resolución mecanismos

Si se parte del Bond Graph descrito en el gráfico de la figura 29 y se le satura de todos los elementos que puede llevar cada elemento tenemos un Bond Graph que se grafica a continuación

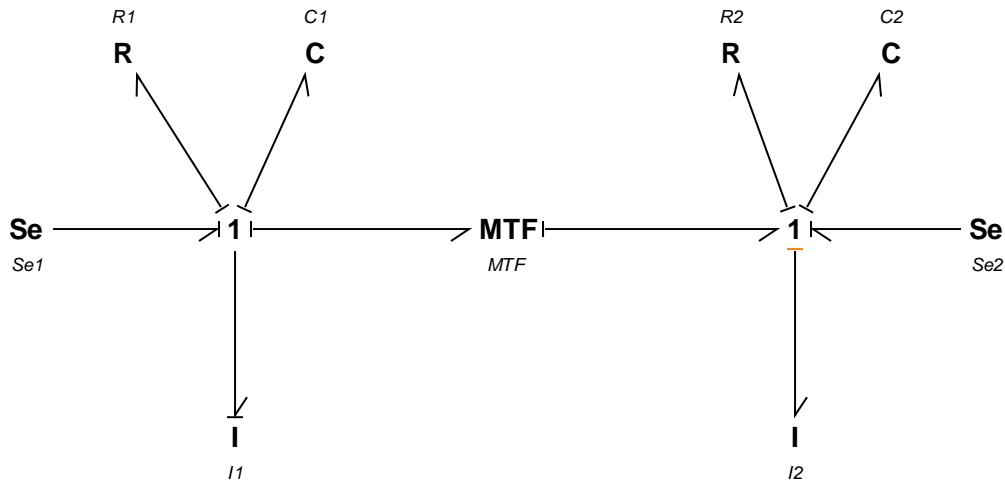


Figura 49 Bond Graph genérico.

Por definición de los elementos de un Bond Graph, este esquema no solo sería aplicable para el mecanismo de biela manivela, sino también a cualquier mecanismo o sistema mecánico o no mecánico que posea dos flujos (velocidades) relacionados directamente, en el caso de mecanismos restringido a mecanismos con un grado de libertad.

El Bond Graph genérico relaciona mecanismos con dos velocidades mediante el elemento transformador T, extrapolando este, se determinar la solución de cualquier mecanismo siempre y cuando se pueda encontrar la relación entre las velocidades, a las que se les puede llamar velocidades de entrada y salida de mecanismo. Como factor determinante el planteamiento de Bonds solo puede resolver mecanismos con un grado de libertad.

Las ecuaciones descritas para el mecanismo biela manivela son las mismas que para cualquier mecanismo, dejando expresado el flujo como f y el esfuerzo como e , permite adaptarse a las diferentes velocidades lineales y angulares además de fuerzas y torques producidos en el mecanismo.

La variable principal para utilizar el algoritmo de Bonds, es el factor de transformación T que tiene que ser calculado independientemente y luego introducido en el sistema de ecuaciones.

Si se numeran los Bonds se tiene:

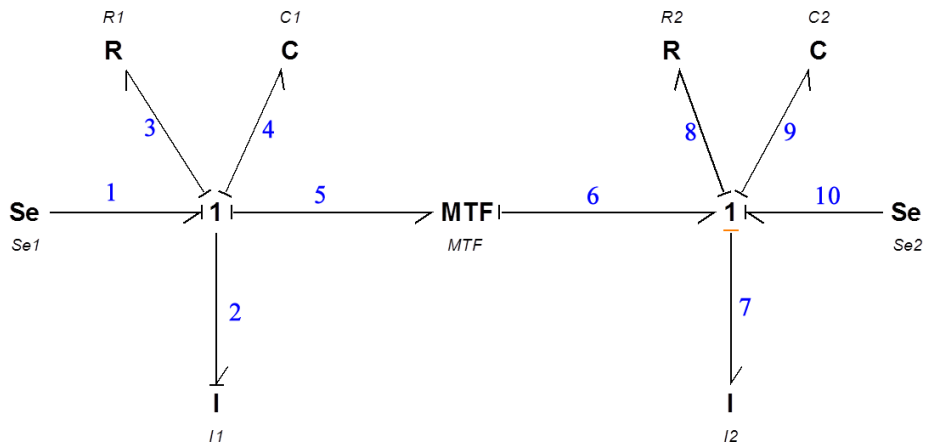


Figura 50 Numeración del Bond Graph.

Las ecuaciones que describen el algoritmo son:

Tabla 9 Sistema de ecuaciones para el mecanismo

Grafo	Flujo f	Esfuerzo e
1	$f_1 = f_2 = f_3 = f_4 = f_5$	$e_1 = Se_1$
2	$f_2 = \frac{p_2}{I}$	$e_2 = \dot{p}_2 = I\dot{f}_2$
3	f_3	$e_3 = B_1 f_3$
4	$f_4 = \dot{q}_4 = \frac{p_2}{I}$	$e_4 = k_1 q_4 = \frac{q_4}{C_1}$
5	f_5	$e_5 = e_1 - e_2 - e_3 - e_4$
6	$f_6 = T f_5$	$e_6 = e_{10} - e_7 - e_8 - e_9 = \frac{1}{T} e_5$

7	$f_7 = \frac{p_7}{I_2}$	$e_7 = \dot{p}_7 = I\dot{f}_7$
8	f_8	$e_8 = B_2 f_8$
9	$f_9 = \dot{q}_9 = \frac{p_7}{I_2}$	$e_9 = k_2 q_9 = \frac{q_9}{C_2}$
10	$f_{10} = f_6 = f_7 = f_8 = f_9$	$e_{10} = S e_2$

Las ecuaciones resultantes se obtienen de reemplazar los valores en las ecuaciones de esfuerzo del elemento 5 y 6.

$$e_{10} - e_7 - e_8 - e_9 = \frac{1}{T}(e_1 - e_2 - e_3 - e_4) \quad (75)$$

$$S e_2 - I\dot{f}_7 - B_2 f_8 - \frac{q_9}{C_2} = \frac{1}{T} \left(S e_1 - I\dot{f}_2 - B_1 f_3 - \frac{q_4}{C_1} \right) \quad (76)$$

El sistema de ecuaciones queda descrito por.

$$S e_2 - I_2 \dot{f}_7 - B_2 f_8 - \frac{q_9}{C_2} = \frac{1}{T} \left(S e_1 - I_1 \dot{f}_2 - B_1 f_3 - \frac{q_4}{C_1} \right) \quad (77)$$

$$\dot{q}_9 = \frac{p_7}{I_2} \quad (78)$$

$$\dot{q}_4 = \frac{p_2}{I_1} \quad (79)$$

Si asumimos que conocemos el comportamiento de uno de los extremos del Bond Graph (se da como dato o fácilmente calculable) en sistema se reduce a dos ecuaciones:

$$Se_2 - I\dot{f}_7 - B_2 f_8 - \frac{q_9}{C_2} = \frac{1}{T} \left(Se_1 - I\dot{f}_2 - B_1 f_3 - \frac{q_4}{C_1} \right) \quad (80)$$

$$f_B = T f_A \quad (81)$$

Cambiando los valores de:

$$f_1 = f_2 = f_3 = f_4 = f_5 = f_A \quad (82)$$

$$f_6 = f_7 = f_8 = f_9 = f_{10} = f_B \quad (83)$$

Reestructurando las ecuaciones.

$$\dot{f}_B = \frac{1}{I_2} \left[Se_2 - B_2 f_B - \frac{q_B}{C_2} - \frac{1}{T} \left(Se_1 - I_1 \dot{f}_A - B_1 f_A - \frac{q_A}{C_1} \right) \right] \quad (84)$$

$$f_B = T f_A \quad (85)$$

Reformulando el Algoritmo de Python se tiene

```
#Importación de Librerías
from sympy import *
from pylab import *

tsim=float(input('Ingrese el tiempo de simulación [s] : '))
n=int(input('Ingrese número de discretizaciones : '))

#PRE PROCESO
#Datos geométricos
L2=1;    #[m] Longitud manivela
L3=2;    #[m] Longitud biela
#Parámetros Bond Graph
Se1=0 #Fuente de esfuerzo manivela, Momento
Se2=0 #Fuente de esfuerzo corredera, Fuerza
B1=0 #Puerta de resistencia manivela
```

```

B2=0 #Puerta de resistencia corredera
k1=0 #Puerta de capacitancia rigidez manivela k1=E.A/L2 ; k1=1/C1
k2=0 #Puerta de capacitancia rigidez corredera k2=E.A/L4 ; k2=1/C2; E:módulo de elasticidad, A:
Área
m2 = 1.0;          #[kg]    Masa manivela
I1 = m2*L2**2/3;  #Puerta de inercia manivela
I2 = 1.0;          #[kgm2]   Masa de la corredera

#PROCESOS
dt=float(tsim/(n-1)); #[s] Paso temporal

#Definir vectores vacios
ti = [0.0]*(n+3); #tiempo discreto
q1 = [0.0]*(n+3); #desplazamiento angular discreto manivela
f1 = [0.0]*(n+1); #velocidad angular manivela discreta
df1 = [0.0]*(n);  #aceleración angular manivela discreta
T = [0.0]*(n);    #Transformador discreto
q2 = [0.0]*(n+1); #desplazamiento corredera discreto
f2 = [0.0]*(n+1); #velocidad corredera discreta
f4 = [0.0]*(n+1); #velocidad corredera discreta

#Flujo de entrada
q1[0]=sin(ti[0]);
#Condiciones iniciales
q2[0]=0;
f2[0]=0;

for i in range(0,n,1):
    ti[i+1]=ti[i]+dt; #Nodos temporales ti: t1,t2,...,tn,tn+1

#Aproximación de primera derivada por diferencias finitas
for i in range(0,n+3,1):
    q1[i]=sin(ti[i]);

for i in range(0,n+1,1):
    f1[i]=(q1[i+1]-q1[i])/dt;

for i in range(0,n,1):
    df1[i]=(f1[i+1]-f1[i])/dt;

for i in range(1,n,1):
    # Ecuaciones de estado
    T[i]=L2*(0.5*(L2/L3)*sin(2*q1[i])/(1-((L2/L3)*sin(q1[i]))**2)**(1/2)+sin(q1[i]));
    # Forma discreta de Euler
    q2[i+1]=q2[i]+(dt)*(T[i])*f1[i];
    f2[i+1]= f2[i]+ (dt)*((Se2-B2*f2[i]-k2*q2[i]-(Se1-I1*df1[i]-B1*f1[i]-k1*q1[i])/abs(T[i]))/I2);
    f4[i+1]= (T[i])* f1[i];

#POST PROCESOS
subplot(3,1,1)
plot(ti[0:n],q2[0:n])

```

```

ylabel('Desplazamiento [m]')
title('Movimiento corredera')
text(2,0,'Vía Transformador')
grid(linestyle='dotted')
subplot(3,1,2)
plot(ti[0:n],f4[0:n])
xlabel('Tiempo [s]')
ylabel('Velocidad [m/s]')
grid(linestyle='dotted')
subplot(3,1,3)
plot(ti[0:n],f2[0:n])
xlabel('Tiempo [s]')
ylabel('Velocidad [m/s]')
text(2,-0.5,'Vía Dinámica')
grid(linestyle='dotted')
show()

```

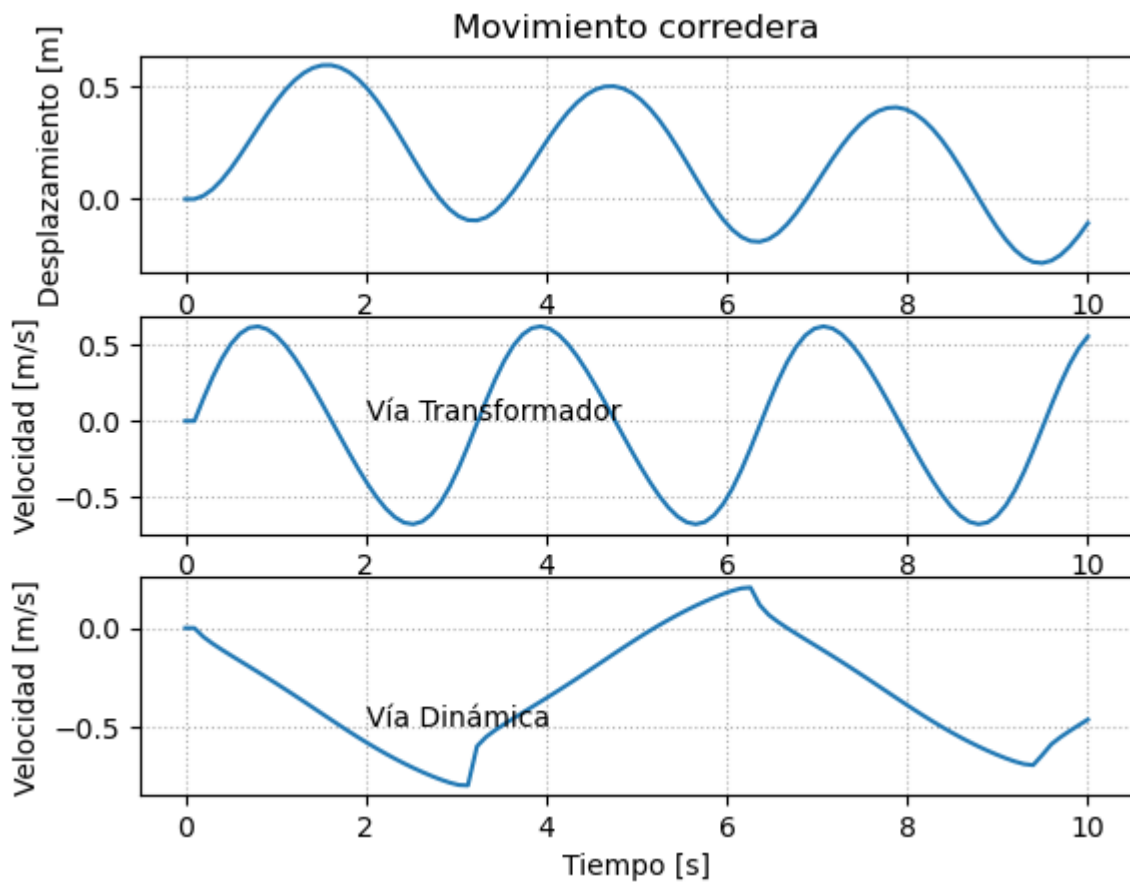


Figura 51 Movimiento descrito por Bond Graph.

Capítulo 5.

Análisis y Resultados

5.1 Resultados

Los resultados obtenidos mediante el método analítico y por Bond Graph se muestran a continuación, analizando los dos métodos con las mismas condiciones producen los siguientes comportamientos con una velocidad angular de la manivela es constante $\omega = 5 \text{ rad/s}$

Para $L_2 \leq L_3$, la ecuación de movimiento para la corredera es, ver ecuación (57):

$$x_4(t) = L_2 \left(\cos \omega t + \frac{L_2}{4L_3} \cos 2\omega t \right) \quad (86)$$

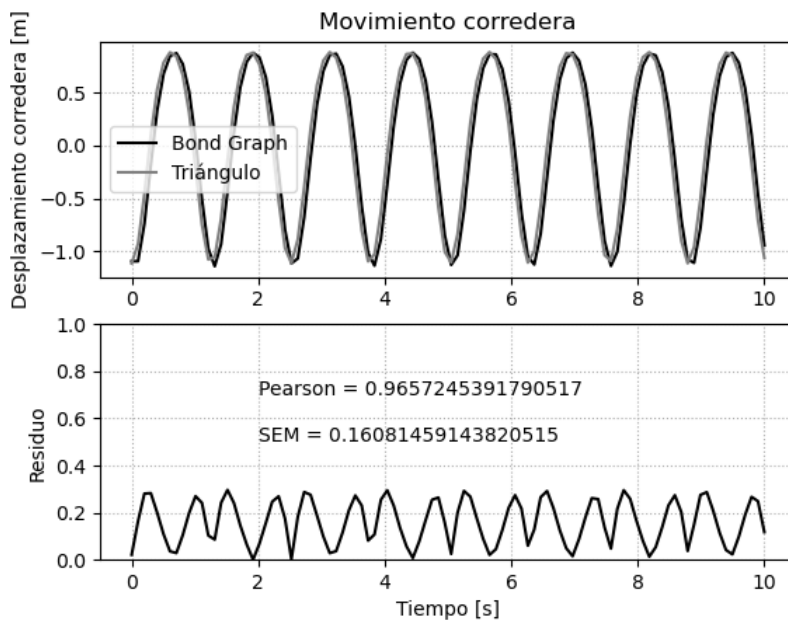


Figura 52 Análisis comparativo Desplazamiento de la corredera.

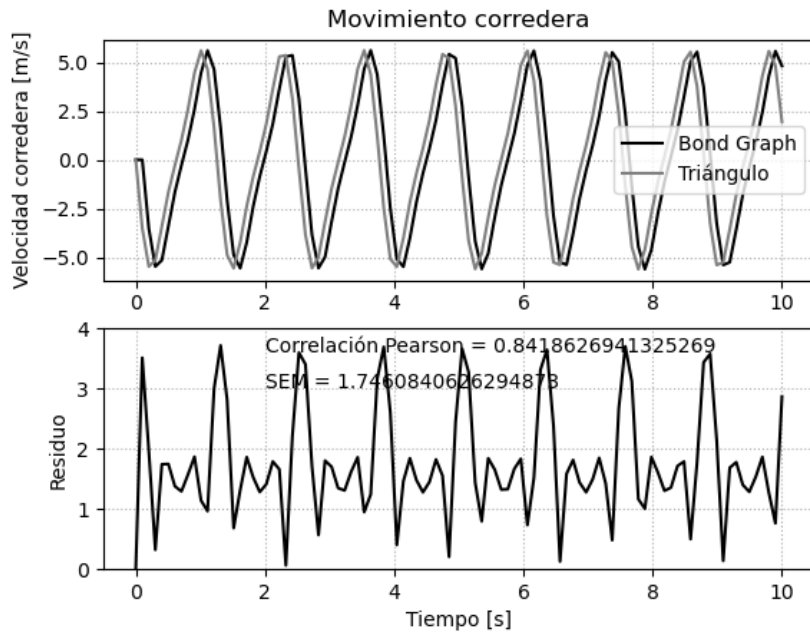


Figura 53 Análisis comparativo Velocidad de la corredera.

De igual manera para simular el comportamiento de este modelo se puede variar las condiciones de entrada, principalmente el tipo de señal que se imprime sobre la manivela. Para el caso donde la función de velocidad angular se comporta como una función lineal descrita en la ecuación $\theta = \omega \cdot t$, muy común al comportamiento de un motor, lo que es una velocidad constante.

Si utilizamos una entrada de $\theta = \sin t$, tenemos en el desplazamiento de la corredera.

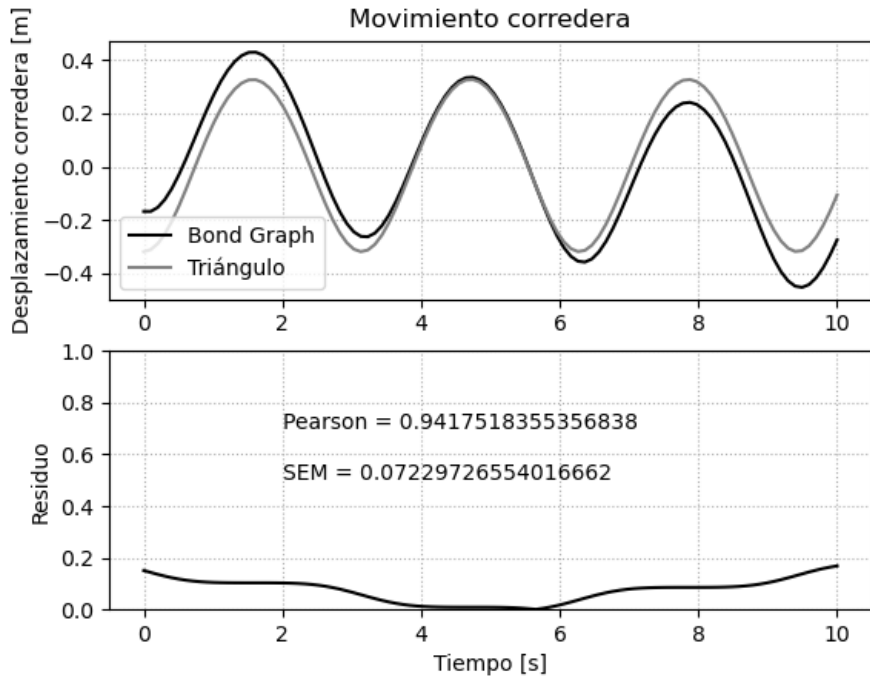


Figura 54 Análisis comparativo Desplazamiento de la corredera.

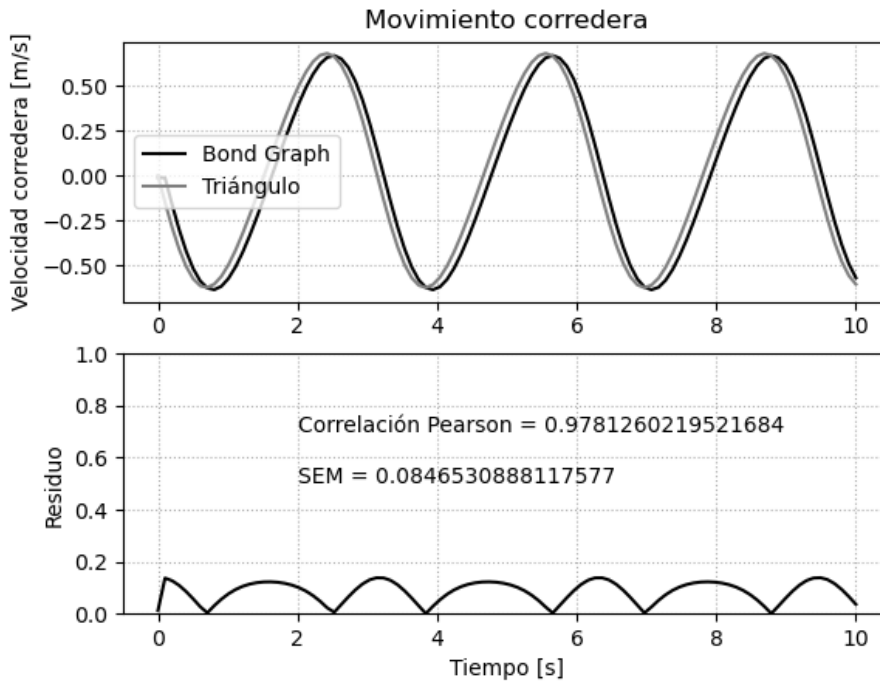


Figura 55 Análisis comparativo Velocidad de la corredera.

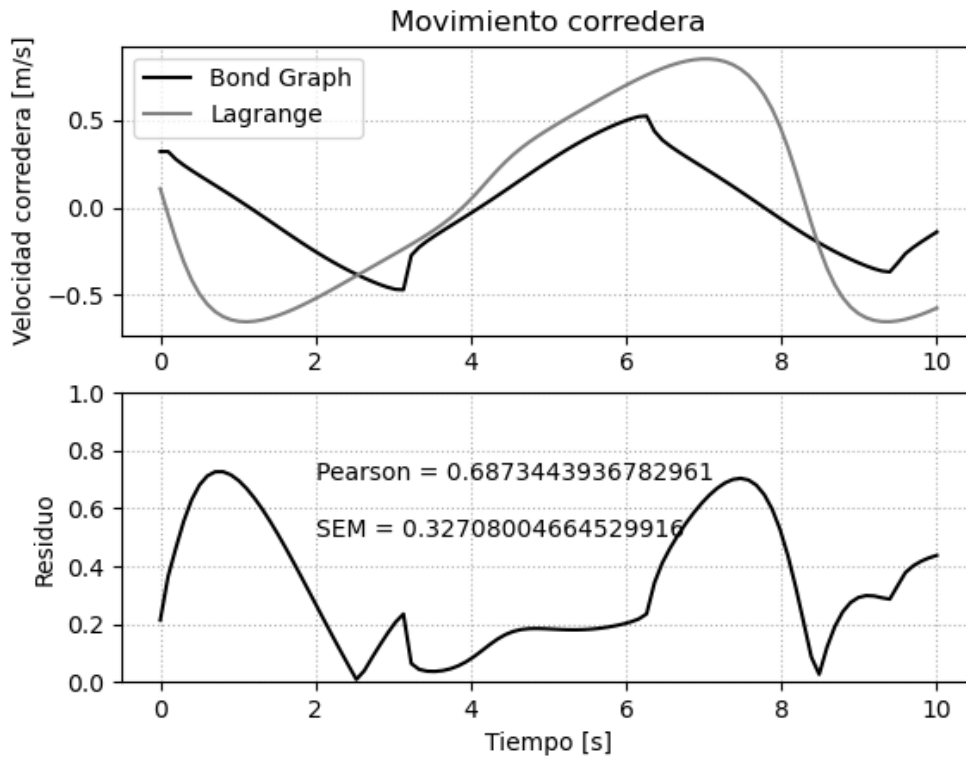


Figura 56 Análisis comparativo Velocidad de la corredera.

5.2 Análisis y Comparación de resultados

Para comparar las dos técnicas matemáticas se determina el error porcentual en contraste, también se realiza una comparación grafica de superposición de curvas (figura 52) para poder observar de mejor manera las diferencias entre los dos métodos.

Para el caso del desplazamiento, en las dos técnicas, los valores obtenidos son restados en su media, a fin de superponer y comparar.

Por otro lado, la orientación de los Bond determina el signo, por lo que estos desplazamientos se pueden asumir que son el valor absoluto de los términos calculados, de este modo, los valores y las curvas son aproximados.

La validación cinemática se la realiza comparando las posiciones y velocidades de los elementos a analizar, para este caso se la realiza comparando los resultados

obtenidos con el método de Bond Graph con un método clásico como es el del triángulo.

El código de programación en Python para la validación cinemática es el siguiente:

```
#Importación de Librerías
from sympy import *
from pylab import *

tsim=float(input('Ingrese el tiempo de simulación [s] : '))
n=int(input('Ingrese número de discretizaciones : '))

#PRE PROCESO
#Datos geométricos
L2=1;    #[cm] Longitud manivela
L3=2;    #[cm] Longitud biela

#PROCESOS METODO TRIANGULO
dt=float(tsim/(n-1)); #[s] Paso temporal
#Definir vectores vacíos
ti = [0.0]*(n+3); #tiempo discreto
O = [0.0]*(n+3); #desplazamiento angular manivela discreto
w = [5.0]*(n+3); #velocidad angular discreta manivela
a = [0.0]*(n);    #aceleración angular discreta manivela
x4 = [0.0]*(n+3); #desplazamiento lineal discreta corredera
v4 = [0.0]*(n+1); #velocidad lineal discreta corredera
residuog=[0.0]*(n+1); #error del desplazamiento
residuof=[0.0]*(n+1); #error de velocidad

#Condiciones iniciales para la manivela
O[0]=w[0]*ti[0]; #sin(ti[0]);
w[0]=w[0]; #Flujo de entrada

for i in range(0,n+2,1):
    ti[i+1]=ti[i]+dt; #Nodos temporales ti: t1,t2,...tn,tn+1

suma1=0;
suma2=0;
#Aproximación de primera derivada por diferencias finitas
for i in range(0,n+3,1):
    O[i]= w[i]*ti[i]; #w[i]*ti[i]; #sin(ti[i]);
    #x4[i]=L2*cos(w[i]*ti[i])+L2**2*cos(2*w[i]*ti[i])/L3;
    x4[i]=L2*cos(O[i])+(L3**2-(L2*sin(O[i]))**2)**(1/2));

for i in range(0,n+1,1):
    w[i]=(O[i+1]-O[i])/dt; #O[i]=sin(ti[i])
    v4[i]=w[i]*L2*(0.5*(L2/L3)*sin(2*O[i])/(1-((L2/L3)*sin(O[i]))**2)**(1/2)+sin(O[i]));
    suma1=suma1+x4[i];
```

```

suma2=suma2+v4[i];

media1=suma1/(n+1);
media2=suma2/(n+1);
x=media1-x4;
v=media2-v4;

#PRE PROCESO METODO BOND GRAPH
#Parámetros Bond Graph
Se1=0 #Fuente de esfuerzo manivela
Se2=0 #Fuente de esfuerzo corredera
B1=0 #Puerta de resistencia manivela
B2=0 #Puerta de resistencia corredera
k1=0 #Puerta de capacitancia rigidez manivela k1=E.A/L2 ; k1=1/C1
k2=0 #Puerta de capacitancia rigidez corredera k2=E.A/L4 ; k2=1/C2; E:módulo de elasticidad, A:
Área
m2 = 1.0;      #[kg]      Masa manivela
I1 = m2*L2**2/3; #Puerta de inercia manivela
I2 = 1.0;      #[kgm2]    Masa de la corredera

#Definir vectores vacios
ti =[0.0]*(n+3); #tiempo discreto
q1 =[0.0]*(n+3); #desplazamiento manivela discreto
f1 =[5.0]*(n+3); #velocidad angular manivela discreta
df1 =[0.0]*(n); #aceleración angular manivela discreta
T =[0.0]*(n); #Transformador discreto
q2 =[0.0]*(n+1); #desplazamiento corredera discreto
f2 =[0.0]*(n+1); #velocidad angular corredera discreta
f4 =[0.0]*(n+1); #velocidad angular corredera discreta

#Flujo de entrada
q1[0]= f1[0]*ti[0]; #f1[0]*ti[0]; #sin(ti[0]);
#Condiciones iniciales
q2[0]=0;
f2[0]=0;

for i in range(0,n,1):
    ti[i+1]=ti[i]+dt; #Nodos temporales ti: t1,t2,...tn,tn+1

#Aproximación de primera derivada por diferencias finitas
for i in range(0,n+3,1):
    q1[i]= f1[i]*ti[i];#f1[i]*ti[i];#sin(ti[i]);
for i in range(0,n+1,1):
    f1[i]=(q1[i+1]-q1[i])/dt;

for i in range(0,n,1):
    df1[i]=(f1[i+1]-f1[i])/dt;

suma1=0;
suma2=0;

```

```

for i in range(1,n,1):
    T[i]=L2*(0.5*(L2/L3)*sin(2*q1[i])/(1-((L2/L3)*sin(q1[i]))**2)**(1/2)+sin(q1[i]));
    # Forma discreta de Euler
    q2[i+1]=q2[i]+(dt)*(T[i])*f1[i]
    f2[i+1]= f2[i]+ (dt)*((Se2-B2*f2[i]-k2*q2[i]-(Se1-l1*df1[i]-B1*f1[i]-k1*q1[i])/abs(T[i]))/l2)
    f4[i+1]= (T[i])* f1[i];
    suma1=suma1+q2[i];
    suma2=suma2+f4[i];

#Análisis estadístico
media1=suma1/(n);
media2=suma2/(n);
q=q2-media1;
f=media2-f4;

sumaq=0;
sumaf=0;
for i in range(1,n,1):
    sumaq=sumaq+q[i];
    sumaf=sumaf+f[i];

mediaq=sumaq/(n);
mediaf=sumaf/(n);

SRq=[0.0]*(n+1); #
SRf=[0.0]*(n+1); #
STq=[0.0]*(n+1); #
STf=[0.0]*(n+1); #

for i in range(1,n,1):
    SRq[i]=(q[i]-x[i])**2; #Cuadrados del residuo
    SRf[i]=(f[i]-v[i])**2; #Cuadrados del residuo
    STq[i]=(q[i]-mediaq)**2; #Cuadrados del total
    STf[i]=(f[i]-mediaf)**2; #Cuadrados del total

#Variables auxiliares
sumaq=0;
sumaf=0;
sumaTq=0;
sumaTf=0;

for i in range(1,n,1):
    sumaq=sumaq+SRq[i]; #Suma de cuadrados del residuo
    sumaf=sumaf+SRf[i]; #Suma de cuadrados del residuo
    sumaTq=sumaTq+STq[i]; #Suma de cuadrados total
    sumaTf=sumaTf+STf[i]; #Suma de cuadrados total

Pearsonq=sqrt(abs(1-sumaq/sumaTq));
Pearsonf=sqrt(abs(1-sumaf/sumaTf));

```

```

# Análisis de error
for i in range(0,n,1):
    residuoq[i]=abs((q[i]-x[i]));
    residuof[i]=abs((f[i]-v[i]));

sumaq=0;
sumaf=0;
for i in range(1,n,1):
    sumaq=sumaq+residuoq[i]; #Suma residuo error estandar medio
    sumaf=sumaf+residuof[i]; #Suma residuo error estandar medio
mediaq=sumaq/(n);
mediaf=sumaf/(n);

```

#POST PROCESOS

```

figure(1)
subplot(2,1,1)
plot(ti[0:n],q[0:n],"k",ti[0:n],x[0:n],"grey")
ylabel('Desplazamiento corredera [m]')
title('Movimiento corredera')
grid(linestyle='dotted')
legend(["Bond Graph" ,"Triángulo" ])
subplot(2,1,2)
plot(ti[0:n],residuoq[0:n],"k")
xlabel('Tiempo [s]')
ylabel('Residuo')
text(2,0.7,'Pearson = '+str(Pearsonq))
text(2,0.5,'SEM = '+str(mediaq))
grid(linestyle='dotted')
ylim(0,1)

```

```

figure(2)
subplot(2,1,1)
plot(ti[0:n],f[0:n],"k",ti[0:n],v[0:n],"grey")
ylabel('Velocidad corredera [m/s]')
title('Movimiento corredera')
grid(linestyle='dotted')
legend(["Bond Graph" ,"Triángulo" ])
subplot(2,1,2)
plot(ti[0:n],residuof[0:n],"k")
xlabel('Tiempo [s]')
ylabel('Residuo')
text(2,3.6,'Correlación Pearson = '+str(Pearsonf))
text(2,3,'SEM = '+str(mediaf))
grid(linestyle='dotted')
ylim(0,4)
show()

```

Adicionalmente el análisis dinámico está asociado a la determinación de la aceleración, esta se logra derivando la ecuación de estado que determina la velocidad en sus componentes normal y tangencial, dando como resultado un código de programación al que se le añade las variables a las aceleraciones.

El código de programación en Python para la validación dinámica es el siguiente:

```
#Importación de Librerías
from sympy import *
from pylab import *

tsim=float(input('Ingrese el tiempo de simulación [s] : '))
n=int(input('Ingrese número de discretizaciones : '))

#PRE PROCESO MÉTODO DE LAGRANGE
#Datos geométricos
L2 = 1;          #[m]   Longitud manivela
L3 = 2;          #[m]   Longitud biela
s = L3/2;        #[m]   Centro de masa de la biela respecto a la corredera
#Datos parámetros pasivos
m = 1.0;         #[kg]   Masa corredera
k2 = 0.0;        #[N/m]  Rigidez manivela
k3 = 0.0;        #[N/m]  Rigidez biela
r = 0.0;         #[Ns/m] fricción corredera
m2 = 1.0;        #[kg]   Masa manivela
m3 = 1.0;        #[kg]   Masa biela
J2 = m2*L2**2/3; #        Inercia manivela
J3 = m3*L3**2/3; #        Inercia biela

#Cargas externas
F=0.0;          #[N]   Fuerza externa aplicada en la corredera
M=0.0;          #[N.m] Momento en la manivela

#PROCESOS
dt=float(tsim/(n-1)); #[s] Paso temporal

#Definir vectores vacíos
ti = [0.0]*(n+3); #tiempo discreto
O = [0.0]*(n+3); #desplazamiento angular discreto manivela
w = [0.0]*(n+1); #velocidad angular discreta manivela
a = [0.0]*(n+1); #aceleración angular discreta manivela
x4 = [0.0]*(n+3); #desplazamiento lineal discreta corredera
v4 = [0.0]*(n+1); #velocidad lineal discreta corredera
a4 = [0.0]*(n);   #aceleración lineal discreta corredera
a4_n = [0.0]*(n); #aceleración lineal discreta corredera
a4_t = [0.0]*(n); #aceleración lineal discreta corredera
N = [0.0]*(n+3); #Variable auxiliar numerador
```

```

D = [0.0]*(n+3); #Variable auxiliar denominador
residuoa=[0.0]*(n+1); #error del desplazamiento
residuof=[0.0]*(n+1); #error de velocidad

#Condiciones iniciales para la manivela
O[0]=sin(ti[0]);
w[0]=cos(ti[0]); #Flujo de entrada

for i in range(0,n+2,1):
    ti[i+1]=ti[i]+dt; #Nodos temporales ti: t1,t2,...tn,tn+1

#Aproximación de primera derivada por diferencias finitas
for i in range(0,n,1):
    O[i+1]=w[i]*dt+O[i]; #O[i+1]=w[i]*dt+O[i], donde O[i]=sin(ti[i]) Forma discreta Euler
    N[i]=F*L2*sin(O[i])-M-((m+m3)*L2**2-(J3+m3*s**2)*(L2/L3)**2)*(w[i])**2*sin(O[i])*cos(O[i]);
#Numerador
    D[i]=((m+m3)*L2**2*(sin(O[i]))**2+(J3+m3*s**2)*(L2*cos(O[i])/L3)**2+J2); #Denominador
    w[i+1]=(N[i]/D[i])*dt+w[i]; #Forma discreta Euler

suma1=0;
suma2=0;
for i in range(0,n,1):
    x4[i]=L2*cos(O[i])+(L3**2-(L2*sin(O[i]))**2)**(1/2);
    v4[i]=w[i]*L2*(0.5*(L2/L3)*sin(2*O[i])/(1-((L2/L3)*sin(O[i]))**2)**(1/2)+sin(O[i]));
    a[i]=N[i]/D[i];
    a4_t[i]=a[i]*L2*(0.5*(L2/L3)*sin(2*O[i])/(1-((L2/L3)*sin(O[i]))**2)**(1/2)+sin(O[i]));
    a4_n[i]=(w[i])**2*L2*((L2/L3)*cos(2*O[i])/(1-
    ((L2/L3)*sin(O[i]))**2)**(1/2)+0.5*(L2/L3)**2*sin(2*O[i])/(1-((L2/L3)*sin(O[i]))**2)**(3/2)+cos(O[i]));
    a4[i]=a4_t[i]+a4_n[i];
    suma1=suma1+x4[i];
    suma2=suma2+v4[i];

media1=suma1/(n+1);
media2=suma2/(n+1);
x=media1-x4;
v=media2-v4;

#PRE PROCESO METODO BOND GRAPH
#Parámetros Bond Graph
Se1=0 #Fuente de esfuerzo manivela
Se2=0 #Fuente de esfuerzo corredera
B1=0 #Puerta de resistencia manivela
B2=0 #Puerta de resistencia corredera
k1=0 #Puerta de capacitancia rigidez manivela k1=E.A/L2 ; k1=1/C1
k2=0 #Puerta de capacitancia rigidez corredera k2=E.A/L4 ; k2=1/C2; E:módulo de elasticidad, A:
Área
m2 = 1.0; #[kg] Masa manivela
I1 = m2*L2**2/3; #Puerta de inercia manivela
I2 = 1.0; #[kgm2] Masa de la corredera

```



```

#Definir vectores vacios
ti =[0.0]*(n+3); #tiempo discreto
q1 =[0.0]*(n+3); #desplazamiento manivela discreto
f1 =[5.0]*(n+3); #velocidad angular manivela discreta
df1 =[0.0]*(n); #aceleración angular manivela discreta
T =[0.0]*(n); #Transformador discreto
q2 =[0.0]*(n+1); #desplazamiento corredera discreto
f2 =[0.0]*(n+1); #velocidad angular corredera discreta
f4 =[0.0]*(n+1); #velocidad angular corredera discreta
#Flujo de entrada
q1[0]= sin(ti[0]); #f1[0]*ti[0]; #sin(ti[0]);
#Condiciones iniciales
q2[0]=0;
f2[0]=0;

for i in range(0,n,1):
    ti[i+1]=ti[i]+dt; #Nodos temporales ti: t1,t2,...tn,tn+1

#Aproximación de primera derivada por diferencias finitas
for i in range(0,n+3,1):
    q1[i]= sin(ti[i]);#f1[i]*ti[i];#sin(ti[i]);
for i in range(0,n+1,1):
    f1[i]=(q1[i+1]-q1[i])/dt;

for i in range(0,n,1):
    df1[i]=(f1[i+1]-f1[i])/dt;

suma1=0;
suma2=0;
for i in range(1,n,1):
    T[i]=L2*(0.5*(L2/L3)*sin(2*q1[i])/(1-((L2/L3)*sin(q1[i]))**2)**(1/2)+sin(q1[i]));
    # Forma discreta de Euler
    q2[i+1]=q2[i]+(dt)*(T[i])*f1[i]
    f2[i+1]= f2[i]+ (dt)*((Se2-B2*f2[i]-k2*q2[i]-(Se1-l1*df1[i]-B1*f1[i]-k1*q1[i])/abs(T[i]))/l2)
    f4[i+1]= (T[i])* f1[i];
    suma1=suma1+q2[i];
    suma2=suma2+f2[i];

#Análisis estadístico
media1=suma1/(n);
media2=suma2/(n);
q=q2-media1;
f=f2-media2;

sumaq=0;
sumaf=0;
for i in range(1,n,1):
    sumaq=sumaq+q[i];
    sumaf=sumaf+f[i];

```

```

mediaq=sumaq/(n);
mediaf=sumaf/(n);

SRq=[0.0]*(n+1); #
SRf=[0.0]*(n+1); #
STq=[0.0]*(n+1); #
STf=[0.0]*(n+1); #

for i in range(1,n,1):
    SRq[i]=(q[i]-x[i])**2; #Cuadrados del residuo
    SRf[i]=(f[i]-v[i])**2; #Cuadrados del residuo
    STq[i]=(x[i]-mediaq)**2; #Cuadrados del total
    STf[i]=(v[i]-mediaf)**2; #Cuadrados del total

#Variables auxiliares
sumaq=0;
sumaf=0;
sumaTq=0;
sumaTf=0;

for i in range(1,n,1):
    sumaq=sumaq+SRq[i]; #Suma de cuadrados del residuo
    sumaf=sumaf+SRf[i]; #Suma de cuadrados del residuo
    sumaTq=sumaTq+STq[i]; #Suma de cuadrados total
    sumaTf=sumaTf+STf[i]; #Suma de cuadrados total

Pearsonq=sqrt(abs(1-sumaq/sumaTq));
Pearsonf=sqrt(abs(1-sumaf/sumaTf));

# Análisis de error
for i in range(0,n,1):
    residuoq[i]=abs((q[i]-x[i]));
    residuof[i]=abs((f[i]-v[i]));

sumaq=0;
sumaf=0;
for i in range(1,n,1):
    sumaq=sumaq+residuoq[i]; #Suma residuo error estandar medio
    sumaf=sumaf+residuof[i]; #Suma residuo error estandar medio
mediaq=sumaq/(n);
mediaf=sumaf/(n);

figure(1)
subplot(2,1,1)
plot(ti[0:n],f[0:n],"k",ti[0:n],v[0:n],"grey")
ylabel('Velocidad corredera [m/s]')
title('Movimiento corredera')
grid(linestyle='dotted')
legend(["Bond Graph" ,"Lagrange" ])

```

```
subplot(2,1,2)
plot(ti[0:n],residuof[0:n], "k")
xlabel('Tiempo [s]')
ylabel('Residuo')
text(2,0.7,'Pearson = '+str(Pearsonf))
text(2,0.5,'SEM = '+str(mediaf))
grid(linestyle='dotted')
ylim(0,1)
show()
```

Capítulo 6.

Conclusiones y Recomendaciones

6.1 Conclusiones

Aunque existe un infinito número de posibilidades de plantear un Bond Graph adecuado a un mecanismo, el más condensado (pequeño) siempre es el mejor ya que este permite visualizar de mejor manera los elementos a resolver, un Bond Graph extenso o un algoritmo ciertamente permite calcular otro tipo de variables, pero muchas veces no justifica ya que estas se pueden obtener fácilmente por otros procedimientos, antes que incurrir en la utilización de estos Bond Graphs.

Las metodologías para el desarrollo del Bond Graph son adaptativas, no existe un método común para plantear un Bond Graph, lo que los autores de libros hacen es dar pautas para poder enfrentarse a la hora de graficar diferentes mecanismos. La metodología planteada aquí recopila pseudo metodologías de muchos autores que a la hora de la representación de un Bond Graph dejan muchos vacíos.

La carencia de elementos apropiados en la metodología de Bond Graph se satisface con el conocimiento del comportamiento físico de los elementos que intervienen en el Bond Graph, algunos autores prefieren recurrir a resolver partes confusas del Bond Graph mediante otros métodos, y luego integrarlos al modelo.

El método de Bond Graph como cualquier otro método permite obtener los resultados del análisis de un mecanismo, una de las desventajas es el requerimiento avanzado de resolución de ecuaciones diferenciales, pero con su gran ventaja de poder manipular las ecuaciones y el Bond Graph para convertirlo en otro, luego de haber resuelto un problema determinado.

El método de Bond Graph permite generar una ecuación genérica, esta puede ser aplicada a diferentes sistemas o mecanismos que compartan estas características, sin variar en ningún elemento, dando solución para un sinnúmero de mecanismos y sistemas.

Un algoritmo genérico para resolver todos los tipos de mecanismos es imposible, lo que se logra concebir es un Algoritmo que permite resolver un grupo de mecanismos con similares características, como para el caso que se presenta un mecanismo de un grado de libertad que pueden relacionar las velocidades de entrada y salida del mecanismo.

6.2 Recomendaciones

Cuando se ingresa ecuaciones con gran número de variables, PYTHON crea un sin número de errores porque no declara tipos de variables como un software interpretador, las cuales no son visibles al programador; por lo que se recomienda ingresar todas las variables como tipo flotante (float) correspondiente a los valores decimales.

La gran cantidad de posibilidades para plantear un Bond Graph puede crear conflictos a la hora de su resolución, es muy recomendable analizar una estrategia para plantear el Bond Graph más apropiado que satisfaga todas las necesidades del analista.

A la hora de dibujar el Bond Graph se recomienda estar muy atento a las variables y datos que influyen en el sistema desde el punto energético del Bond Graph y desde el punto físico, esto permitirá encontrar las relaciones y ecuaciones faltantes a la hora de la resolución del sistema de ecuaciones diferenciales.

La dirección de las medias flechas de los Bonds muchas veces no son posibles determinar, lo más recomendable es ignorar el sentido verdadero, y asumir una, luego las ecuaciones determinaran si el sentido que se escogió fue el correcto reflejándose como valores de signo negativo y positivo o con la causadidad.

Referencias Bibliográficas

- [1] D. Shubra, Mechatronic Modeling and Simulation Using Bond Graphs, USA: CRC Press, 2009.
- [2] W. Borutzky, Bond Graph Modelling of Engineering Systems, London: Springer, 2011.
- [3] R. L. Norton , Diseño de Máquinas, cuarta ed., México: Person, 2011, p. 888.
- [4] D. Myszka, Máquinas y Mecanismos, cuarta ed., México: Person, 2012, pp. 84, 88, 142,353.
- [5] J. E. Shigley y J. J. Uicker, JR, Teoría de Máquinas y Mecanismos, México: McGraw Hill, 2001.
- [6] W. Borutzky, Bond Graph Methodology, London: Springer, 2010.
- [7] J. Félez, B. Suárez y G. Romero, «Open Course Ware,» 2010. [En línea]. Available: http://ocw.upm.es/ingenieria-mecanica/simulacion-en-ingenieria-mecanica/contenidos/teoria/T01_Introduccion.pdf.
- [8] D. C. Karnopp, D. L. Margolis y R. C. Rosenberg, System Dynamics, USA: John Wiley and Sons, 2012.

[9] Kozlova Z.P., Pánshina A.V., Rozenblat G.M. DINÁMICA DEL PUNTO MATERIAL. Mecánica teórica. Resolución detallada de los problemas del libro de I.V.MESCHERSKI

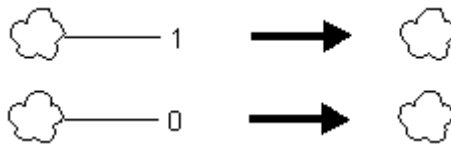
[10] Churkin V.M., Pánshina A. V. MECÁNICA ANALÍTICA. Mecánica teórica. Resolución detallada de los problemas del libro de I. V. MESCHERSKI

Anexos

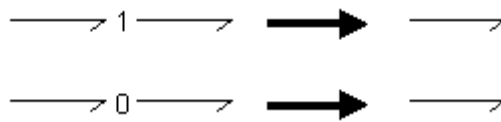
Anexo 1

Simplificación de Modelos de Bond Graph [20-sim referencia manual]

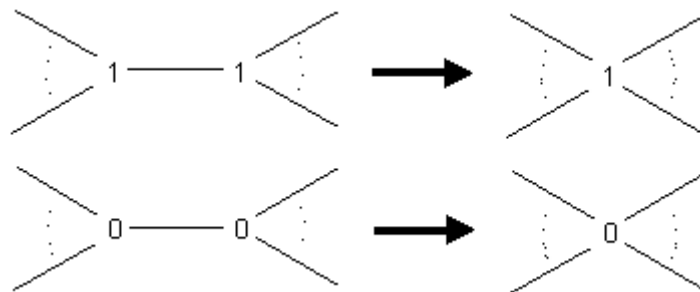
1. Eliminación de nodos perdidos o sueltos



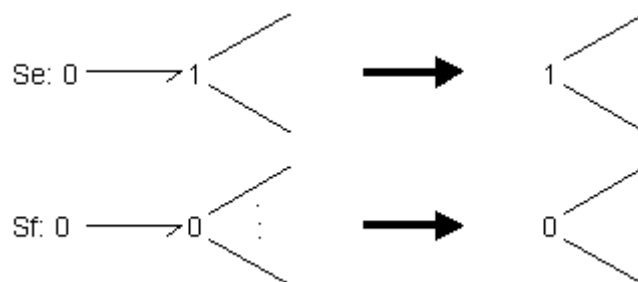
2. Eliminación de nodos



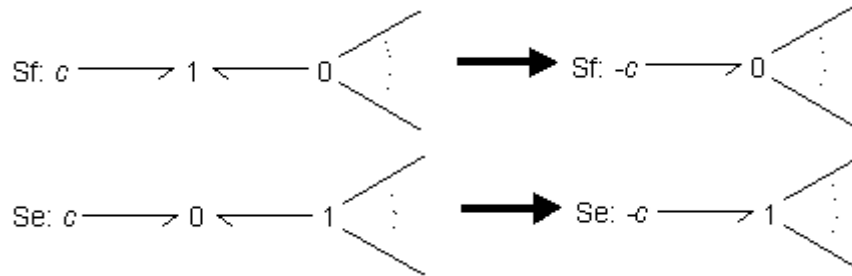
3. Fusión de nodos iguales



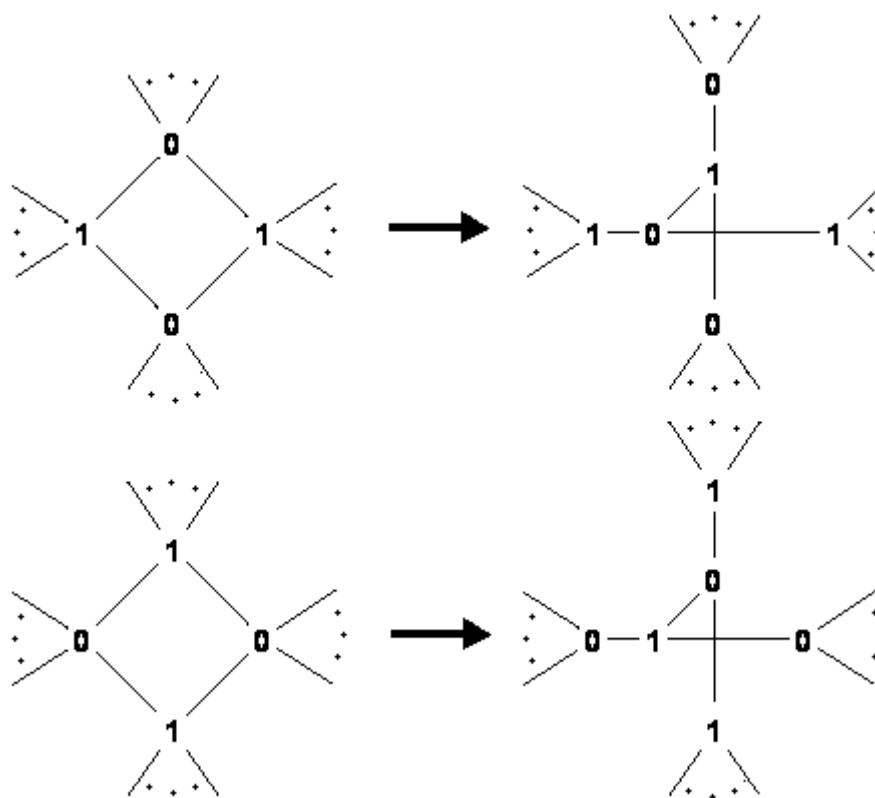
4. Eliminación de fuentes con valor cero



5. Eliminación de nodos en combinación de un elemento fuente cambian el elemento fuente.



6. Eliminación de una doble diferencia, una pareja de dos nodos tipo 0 por un nodo tipo 1.



Anexo 2

Instalación de Python y librerías matplotlib, numpy, y otras.

Descargar el programa de la página oficial de python.org

Se da doble clic en el archivo descargado python-2.7.11.msi, se prefiere la versión 2.7 por ser una versión más estable donde todas las librerías funcionan perfectamente, las versión de python 3 no tiene una compatibilidad completa para todas las librerías.

Se instala las librerías ayudándose de la herramienta pip incluida en el programa, para esto se abre el programa cmd.exe, se va a la dirección donde se encuentra la herramienta pip de python y se la ejecuta de la siguiente manera.

En el cmd se escribe

```
cd C:\Python27\Scripts
```

Esto permite llegar a la carpeta contenedora del archivo pip, luego se lo ejecuta escribiendo.

```
C:\Python27\Scripts>pip
```

Luego se instala la librería matplotlib, que a su vez instala las otras librerías que se requieren, mediante el siguiente código.

```
C:\Python27\Scripts>pip install matplotlib
```

Luego de esto se espera que termine de descargar e instalar todas la librerías y el programa python está listo para usarse.

Además de estas librería se descarga la librería sumpy de su página oficial www.sympy.org, la que se instala mediante el archivo ejecutable sympy.exe

Para empezar a programar se abre el programa IDLE (Python GUI) que se encuentra en el menú de inicio\Todos los Programas\Python27.