

Análisis de vibraciones mecánicas para resortes helicoidales en paralelo empleando el software Octave

Analysis of vibration mechanics for parallel helicoidal springs with the Octave software

José L. Rodríguez-Muñoz^a, Sandra I. González-Beltrán^b, Carlos D. Muñoz-Hernández^c, José S. Pacheco-Cedeño^d, Carlos E. Borja-Soto^e

Abstract:

A spring is a mechanical element that has the ability to accumulate mechanical energy to release it with the goal to exert force, provide flexibility or reduce mechanical vibrations. Within the field of engineering, there is a need to analyze the behavior of simple harmonic motion (MAS), as well as the development of viable tools to improve the teaching-learning process in this subject. For this reason, this work aims to develop a code in the Octave software that allows analyzing the behavior of a parallel mass-spring system. The code is based on input conditions such as: mass, gravity acceleration, time, as well as values of the stiffness constants that the user can select according to each proposed configuration and as result, the graphs of the equations of position, velocity and acceleration are provided for each case.

Keywords:

MAS, Spring, Behavior, Parallel, Octave

Resumen:

Un resorte es un elemento mecánico que posee la capacidad de acumular energía mecánica para liberarla oportunamente con el fin de ejercer una fuerza, brindar flexibilidad o reducir las vibraciones mecánicas. Dentro del campo de la ingeniería, existe la necesidad de analizar el comportamiento del movimiento armónico simple (MAS), así como el desarrollo de herramientas viables para mejorar el proceso de enseñanza-aprendizaje en este tema. Por tal motivo, este trabajo tiene como objetivo desarrollar un código en el software Octave que permita analizar el comportamiento de un sistema masa-resorte en paralelo. El código es basado en condiciones de entrada como son: masa, aceleración de la gravedad, tiempo, así como valores de las constantes de rigidez que el usuario puede seleccionar acorde a cada configuración propuesta y como resultado, se proporcionan los gráficos de las ecuaciones de posición, velocidad y aceleración para cada caso.

Palabras Clave:

MAS, Resorte, Comportamiento, Paralelo, Octave

Introducción

Un resorte es un elemento mecánico que posee la capacidad de acumular energía mecánica para liberarla oportunamente con el fin de ejercer una fuerza, brindar

flexibilidad o reducir las vibraciones mecánicas (Gerez, 2014). Existe una amplia gama de tipos de resortes, dentro de los más comúnmente empleados en el campo de la ingeniería mecánica, se encuentran los resortes helicoidales.

^a Autor de Correspondencia, Universidad Autónoma del Estado de Hidalgo, <https://orcid.org/0000-0002-4108-9414>, Email: jose_rodriguez@uaeh.edu.mx

^b Universidad Autónoma del Estado de Hidalgo, <https://orcid.org/0009-0004-9547-8652>, Email: go435664@uaeh.edu.mx

^c Universidad Autónoma del Estado de Hidalgo, <https://orcid.org/0009-0001-8361-3608>, Email: mu435922@uaeh.edu.mx

^d Universidad Autónoma del Estado de Hidalgo, <https://orcid.org/0000-0002-3400-518X>, Email: sergio_pacheco@uaeh.edu.mx

^e Universidad Autónoma del Estado de Hidalgo, <https://orcid.org/0000-0003-3385-8348>, Email: carlos_borja@uaeh.edu.mx

Los resortes helicoidales tienen la característica de almacenar una gran cantidad de energía y un mínimo nivel de rozamiento, lo que permite que tengan una vida útil prolongada y sean resistentes a las fuerzas de tensión y compresión. Dentro de los materiales comúnmente empleados para la fabricación de resortes, se encuentran los aceros al carbono, aceros inoxidable, aceros al cromo-silicio, aceros al cromo-vanadio y bronce, los cuales son comúnmente empleados en aplicaciones que requieran alta resistencia a la fatiga, alta resistencia a la corrosión, alta durabilidad o se encuentren expuestos a esfuerzos elevados y larga duración a las cargas de impacto (Martínez, 2013).

Uno de los parámetros importantes a estudiar en el movimiento armónico simple, en el que se emplean resortes helicoidales, es la deformación que sufren y las fuerzas que interactúan en cada uno de ellos. En este sentido, Maestre et al. (2015) emplearon el método de las diferencias finitas para analizar el comportamiento de un movimiento armónico simple de un sistema masa-resorte y los resultados obtenidos se compararon con los obtenidos de manera analítica. Sus resultados muestran que el método de diferencias finitas es muy eficiente y eficaz en la solución de este tipo de problemas. Ríos et al. (2017) realizaron una simulación experimental para la enseñanza del movimiento oscilatorio mediante el software Tracker versión 4.87. Para su análisis se adquirieron datos experimentales del movimiento armónico simple obtenidos de un péndulo y el movimiento armónico amortiguado de un sistema masa resorte suspendido en forma vertical. Con base a sus resultados, ellos concluyen que los resultados obtenidos del modelo desarrollado en el software, son muy similares a los resultados obtenidos por los instrumentos de medición. Scheck (2013) empleó el método de la transformada de Laplace para modelar el movimiento armónico simple de un sistema masa-resorte. Él concluye que el método de la transformada de Laplace facilita notablemente la resolución de ecuaciones diferenciales de cualquier orden, así como un análisis más rápido y certero para el análisis de problemas de vibraciones. Fernández et al. (2010) realizaron la medición de la aceleración de la gravedad en un sistema masa-resorte. Ellos emplearon la ley de Hooke y la teoría del movimiento armónico simple en su análisis, encontrando una incertidumbre del 1%, en comparación con los resultados experimentales. Yu y Hao (2011) realizaron el análisis de vibraciones en resortes helicoidales de sección no circular. El efecto de la inercia, deformaciones, torsión fueron evaluadas empleando el software Mathematica el método de la búsqueda de la raíz de Muller para determinar las frecuencias naturales. El modelo desarrollado es comparado con el método de elemento finito y los

resultados obtenidos muestran una buena concordancia entre ambos métodos.

Otros estudios se basan en desarrollar herramientas computacionales y pruebas experimentales sobre el movimiento armónico simple y análisis de vibraciones, con el objetivo de mejorar el proceso de enseñanza-aprendizaje en esta área de estudio. En este sentido, Musik (2017) desarrolló un modelo computacional basado en datos experimentales para determinar la posición, velocidad y aceleración en un MAS, con el objetivo de desarrollar un sistema didáctico en tiempo real. Con base al modelo desarrollado, el autor determinó que el proceso de enseñanza resulta más sencillo y los estudiantes pueden adquirir conceptos físicos de una manera más sencilla y lo aplique en su vida profesional. Tapia (2018) realizó un estudio para determinar la constante elástica del resorte empleando el método estático de la Ley de Hooke y el método dinámico, empleando los conceptos de movimiento armónico simple, con el objetivo de generar recursos didácticos de bajo costo. Los resultados obtenidos muestran que ambos modelos resultan ser adecuados para el cálculo de la constante elástica del resorte y pueden resultar en una herramienta viable para mejorar el proceso de enseñanza-aprendizaje en este tema. Amrani y Paradis (2010) determinaron que los modelos computacionales para el análisis de vibraciones mecánicas benefician el proceso de enseñanza-aprendizaje y estimulan a que los estudiantes tengan más claros los conceptos relacionados a este tema. García (2012) desarrolla y aplica una unidad didáctica para la enseñanza relacionada al movimiento armónico simple. La propuesta es basada en la realización de talleres didácticos, prácticas de laboratorio y simulaciones con Physics, Power Point y Excel. Sus resultados revelan una mejora significativa en el desempeño y de forma cualitativa, un aumento notable en la motivación por aprender y una satisfacción al comprobar que las predicciones funcionan y contribuyen a la enseñanza de este fenómeno. Bongianino y Cistac (2012) desarrollaron un instrumento virtual de aprendizaje para ser utilizado como simulador de un sistema mecánico aplicado en un sistema masa-resorte, con el objetivo de mejorar el proceso de enseñanza-aprendizaje en este campo de la ingeniería. El simulador fue desarrollado en el lenguaje Java y ellos encontraron que esta herramienta resulta ser de gran utilidad para entender y mejorar los conocimientos en este tema. López et al. (2017) realizaron el modelado y simulación del movimiento oscilatorio de un sistema masa-resorte, mediante la solución simbólica y grafica empleando Maple y los resultados obtenidos fueron comparados con la solución analítica. Sus resultados demuestran que el uso de Maple ayuda a resolver ecuaciones diferenciales ordinarias involucradas en la

solución de problemas de vibraciones, así como herramienta para que los estudiantes desarrollan técnicas para la formulación de problemas, así como la creatividad y el diseño de nuevos dispositivos mecánicos. Es evidente que el uso de herramientas computacionales y desarrollo de modelos matemáticos que ayuden a predecir el comportamiento en un sistema oscilatorio armónico simple, es de gran importancia en el campo de la ingeniería. Por tal motivo, este trabajo tiene como objetivo desarrollar un código en el software Octave que permita analizar el comportamiento de un sistema masa-resorte en paralelo. El código es basado en condiciones de entrada como son: masa, gravedad, tiempo, así como valores de las constantes de rigidez que el usuario puede seleccionar acorde a cada sistema analizado y como resultado, se proporcionan los gráficos de las ecuaciones de posición, velocidad y aceleración para cada caso.

Metodología de análisis

Los resortes en paralelo presentan la peculiaridad de estar dispuestos de forma que todos sus extremos iniciales están acoplados por un lado y los extremos finales por el otro. En la Figura 1 se muestra el diagrama esquemático y diagrama de cuerpo libre de un sistema masa-resorte en paralelo. Es importante mencionar que, cuando el sistema se encuentra en forma vertical, no se requiere de una fuerza externa para poder deformar los resortes, como comúnmente se realiza en los sistemas colocados de forma horizontal. Por tal motivo, en el sistema masa-resorte analizado de forma vertical, se coloca un peso en el extremo de los resortes, de tal manera que dicho peso es el encargado de ejercer la fuerza necesaria para generar su estiramiento.

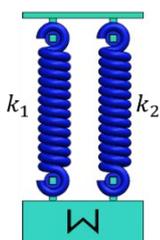


Figura 1. Sistema masa-resorte en paralelo. Fuente: Elaboración propia

En la Figura 2 se ilustra la deformación del resorte, así como el diagrama de cuerpo libre del mismo. Por simplicidad del sistema, se supone un caso unidimensional y que los resortes se encuentran situados en forma vertical y cuelgan de un peso w . Además, la masa está unida al techo a través de dos resortes de constantes k_1 y k_2 . Cuando la masa desciende una cantidad x , los dos muelles se estirarán en la misma cantidad Δx_1 y Δx_2 .

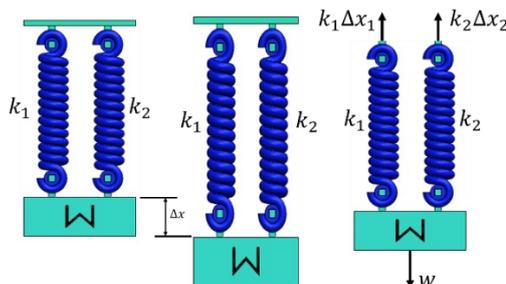


Figura 2. Sistema masa-resorte en paralelo. Fuente: Elaboración propia

Para el análisis del sistema masa-resorte, es necesario determinar la constante recuperadora, también conocida como rigidez equivalente del sistema, la cual se obtiene a partir de la siguiente expresión:

$$k_{eq} = k_1 + k_2 + \dots + k_n \tag{1}$$

Donde k_{eq} , es la constante equivalente del sistema y k_1 a k_n , son las constantes de rigidez empleadas para cada resorte. Cuando los resortes se encuentren en paralelo, la fuerza a la cual está sometido el sistema, se obtiene como la suma de las fuerzas individuales aplicadas a cada resorte:

$$F_{sis} = F_1 + F_2 + \dots + F_n \tag{2}$$

Por otra parte, cuando se trabaja con un sistema de resortes en paralelo, la deformación o alargamiento producido por cada resorte son prácticamente iguales, la cual se obtiene a partir de la ecuación 3:

$$x_T = x_1 = x_2 = x_n = \frac{F_{sis}}{K_{eq}} \tag{3}$$

Otro de los parámetros importantes en el análisis de vibraciones mecánicas, es el periodo, el cual, es el tiempo requerido para realizar un ciclo o para efectuar un viaje de ida y vuelta y sus unidades se dan en segundos o milésimas de segundo, dependiendo de que tan rápido se cambie la onda. Dicha expresión se muestra a continuación:

$$T = \frac{1}{f} \tag{4}$$

Por otra parte, la frecuencia se puede establecer como el número de ciclos que se llevan a cabo durante un segundo. Desde un punto de vista matemático, la frecuencia se puede definir como el recíproco del periodo. Si se le aplica energía a un sistema masa-resorte, el sistema vibrará a su frecuencia natural, y el nivel de las vibraciones dependerá de la fuerza de la fuente de energía y de la absorción inherente al sistema (White, 2010). Por lo tanto, la frecuencia natural de vibración de un sistema masa-resorte no amortiguado se puede obtener de la siguiente forma:

$$F_n = \frac{1}{2\pi} \sqrt{\frac{k}{m}} \tag{5}$$

Donde F_n , k y m , son la frecuencia natural de vibración en Hertz (Hz), la constante de rigidez del resorte en N/m y la masa del sistema en kg, respectivamente. Es importante señalar que, si la rigidez aumenta, la frecuencia natural también aumentará, y si la masa aumenta, la frecuencia natural disminuirá. Una manera de conocer la trayectoria o movimiento en un sistema masa-resorte, es realizando un gráfico de su posición. Para ello, es necesario emplear la ecuación simplificada del movimiento armónico simple, de la siguiente manera:

$$x(t) = A \cos(\omega t + \phi) \tag{6}$$

Donde A se denomina amplitud del movimiento en metros, y es el máximo desplazamiento que la masa puede efectuar respecto a su posición inicial o de equilibrio $x=0$, t es el tiempo en segundos, ϕ es el ángulo de fase en radianes. El término ω , se refiere a la frecuencia angular, la cual representa la velocidad de cambio de la fase de movimiento en rad/s y se puede obtener a partir de la siguiente expresión:

$$\omega = \sqrt{\frac{k}{m}} \tag{7}$$

Los otros parámetros que se deben considerar durante el análisis de un sistema de vibración simple, es la velocidad y la aceleración, las cuales se obtienen por medio de la primera y segunda derivada de la posición, respectivamente y dichas expresiones se muestran a continuación:

$$v(t) = -A\omega \sin(\omega t + \phi) \tag{8}$$

$$a(t) = -A\omega^2 \cos(\omega t + \phi) \tag{9}$$

Resultados

Un código es desarrollado en el software Octave con el objetivo de analizar y evaluar los parámetros más importantes en el MAS, como son: deformaciones, fuerzas y rigidez en un sistema masa-resorte en paralelo. El código es basado en parámetros de entrada, constantes de rigidez del resorte, obteniendo como parámetros de salida los gráficos correspondientes a las ecuaciones de posición, velocidad y aceleración. En el diagrama de flujo mostrado en la Figura 3 se ilustra la metodología de solución para el análisis de MAS para resortes en paralelo.

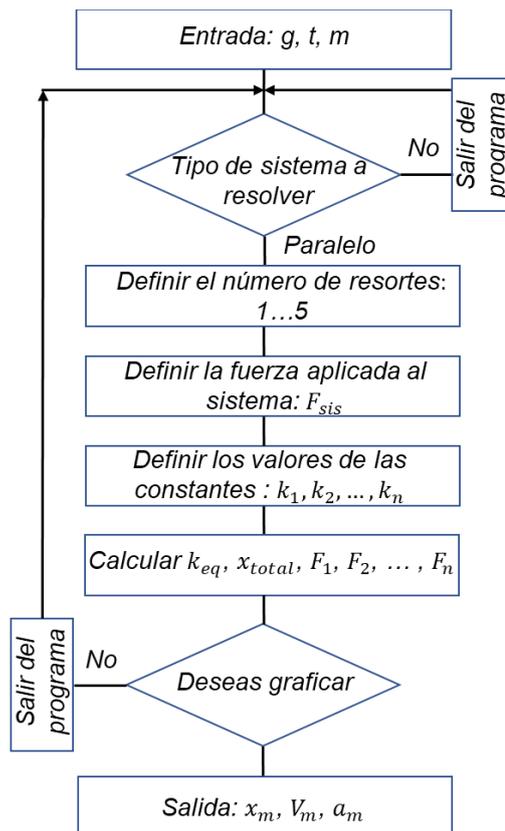


Figura 3. Diagrama de flujo del MAS para resortes en paralelo.
Fuente: Elaboración propia

Validación del código

Para la validación de los resultados, se desarrolló un modelo matemático para tres tipos de sistemas distintos del MAS; A) con dos resortes, B) con tres resortes y C) con cuatro resortes y los resultados obtenidos son comparados con los valores obtenidos con el código desarrollado en el software en Octave. En la Tabla 1 se muestran las condiciones de entrada empleadas en las tres configuraciones antes mencionadas. Además, se establece que los resortes cuelgan de una masa de 2 kg y los valores de sus constantes de rigidez son iguales a sus números de orden expresados en centenas de N/m.

Tipo de sistema	Masa (kg)	Valores de la constante de rigidez
A	2	k ₁ =100
		k ₂ =200
B	2	k ₁ =100
		k ₂ =200
		k ₃ =300
C	2	k ₁ =100
		k ₂ =200
		k ₃ =300
		k ₄ =400

Tabla 1. Condiciones de entrada empleados en cada sistema.
Fuente: Elaboración propia

Los valores de la constante de rigidez equivalente, la deformación y la fuerza aplicada en cada resorte para ambos modelos se muestran en la Tabla 2. Es notorio que el incremento del número de resortes, ocasiona que la constante equivalente de rigidez del sistema también se incremente. Cuando el sistema opera con dos resortes, la constante de rigidez equivalente es de 300 N/m, mientras al usar tres y cuatro resortes, las constantes de rigidez equivalentes se incrementan a valores de 600 N/m y 1000 N/m, respectivamente. También, se puede observar que la deformación y la fuerza aplicada se reduce al incluir un mayor número de resortes al sistema. Esto tiene que ver a que, entre mayor número de resortes se empleen, mayor resistencia presenta el sistema a ser deformado, aunque, desde el punto de vista de diseño, se debe analizar qué tipo de configuración resulta más adecuada para su implementación. Los resultados demuestran que el código desarrollado en Octave predice al 100% los resultados de deformación y fuerza aplicada para cada sistema.

Tipo de sistema	k (N/m)	k_{eq} (N/m)	Modelo		Octave	
			Δx (cm)	F (N)	Δx (cm)	F (N)
A	$k_1=100$	300	6.54	6.54	6.54	6.54
	$k_2=200$			13.08		
B	$k_1=100$	600	3.27	3.27	3.27	6.54
	$k_2=200$			6.54		
	$k_3=300$			9.81		
C	$k_1=100$	1000	1.96	1.96	1.96	3.92
	$k_2=200$			3.92		
	$k_3=300$			5.88		
	$k_4=400$			7.84		

Tabla 2. Validación del MAS para un sistema en paralelo. Fuente: Elaboración propia

En la Figura 4a-c se muestran los resultados correspondientes a la gráfica de posición del MAS para un sistema de resortes en paralelo. La Figura 4a presenta los valores correspondientes a la deformación para el sistema A, donde la máxima deformación obtenida es de 6.54 cm. Para el caso del sistema B (Figura 4b), la deformación que se origina con el uso de tres resortes, es de 3.27 cm, esto representa la mitad de la deformación obtenida por el sistema A. Para el caso del sistema C (Figura 4c), la deformación es 3.3 veces menor que la obtenida con el sistema A y 1.86 veces menor, respecto a los valores obtenidos con el sistema B. Con base a los resultados, se puede concluir que la deformación es menor a medida que se incrementa el número de resortes, esto se debe principalmente al hecho que incrementar el número de resortes, incrementa la resistencia a la deformación y se reduzca la fuerza aplicada a cada resorte.

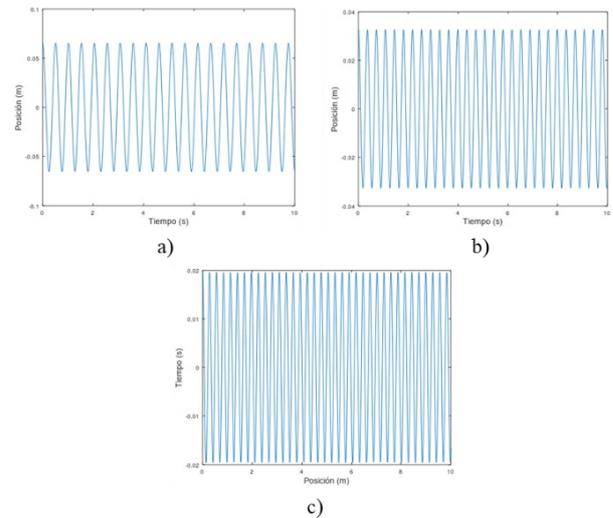


Figura 4. Posición en el MAS para las tres sistemas: a) dos resortes, b) tres resortes y c) cuatro resortes. Fuente: Elaboración propia

La velocidad del sistema masa-resorte del MAS en paralelo se ilustra en la Figura 5. La velocidad con la que el sistema regresa a su posición original se reduce con el incremento con el número de resortes. Para el sistema con dos resortes, la máxima velocidad es de 0.7999 m/s. cuando se emplean tres resortes, la velocidad se reduce hasta un valor de 0.5664 m/s, mientras para el caso de cuatro resortes, la máxima velocidad alcanzada por el sistema es de 0.4386 m/s. De los resultados se puede concluir que la velocidad del sistema B se reduce 1.41 veces, respecto al sistema A, mientras la velocidad del sistema C es 1.82 veces menor, respecto al sistema A y 1.29 veces menor, en comparación al sistema B.

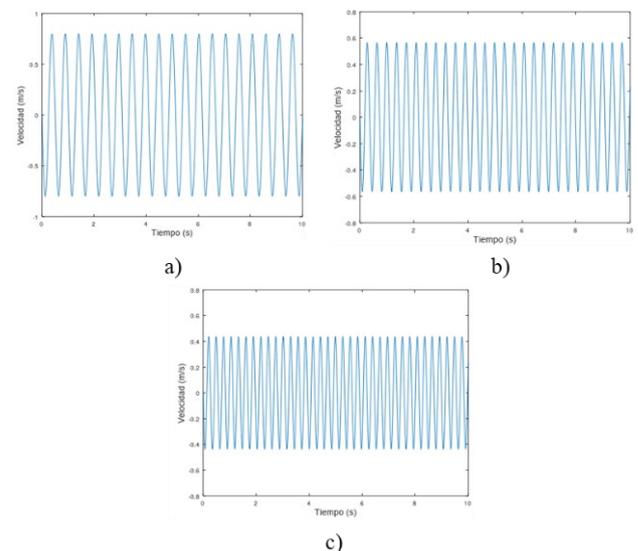


Figura 5. Velocidad en el MAS para los tres sistemas: a) dos resortes, b) tres resortes y c) cuatro resortes. Fuente: Elaboración propia

En la Figura 6a-c se ilustran los valores correspondientes a la aceleración para el MAS al emplear resortes en paralelo. Los resultados muestran que la máxima

aceleración alcanzada por los tres sistemas es de 9.8 m/s^2 . Esto se debe a que la ecuación de la aceleración depende en gran medida de la amplitud del sistema, el cual se reduce a medida que se incrementa el número de resortes.

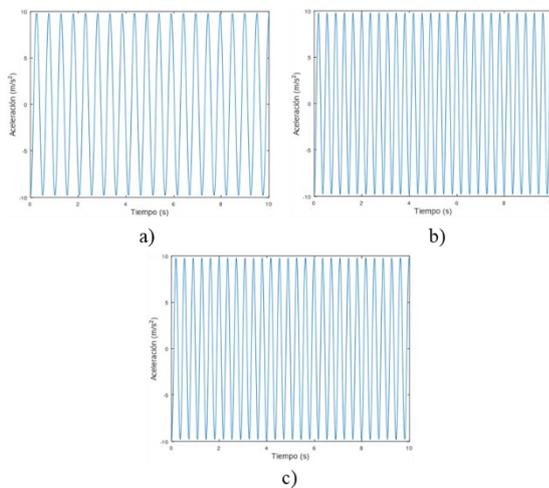


Figura 6. Aceleración en el MAS para los tres sistemas: a) dos resortes, b) tres resortes y c) cuatro resortes. Fuente: Elaboración propia

Conclusiones

El desarrollo de un código que permite analizar el comportamiento de un sistema masa-resorte en paralelo es desarrollado en el software Octave. El código es basado en condiciones de entrada como son: masa, aceleración de la gravedad, tiempo, así como valores de las constantes de rigidez que el usuario puede seleccionar acorde a cada sistema analizado y como resultado, se proporcionan los gráficos de las ecuaciones de posición, velocidad y aceleración para cada caso y los resultados obtenidos se comparan con el modelo desarrollado de manera analítica. Con base a los resultados se concluye lo siguiente:

El código predice correctamente los valores de la constante equivalente de rigidez y la deformación para cada tipo de sistema.

La deformación se incrementa con el incremento del número de resortes. La mayor deformación de 6.54 cm se obtiene con el sistema con dos resortes, mientras la menor deformación de 1.98 cm se logra con el sistema con tres resortes. También se observa una reducción en la fuerza aplicada al sistema a medida que se incrementa también el número de resortes.

La velocidad se ve afectada con el incremento del número de resortes, la máxima velocidad de 0.7999 m/s se obtiene al usar el sistema con dos resortes. Al aumentar la cantidad de resortes a cada sistema, se puede encontrar que la velocidad con el sistema de tres resortes, la velocidad se reduce hasta un valor de 0.5664 m/s , mientras para el caso de cuatro resortes, la máxima velocidad alcanzada por el sistema es de 0.4386 m/s . De los resultados se puede concluir que la velocidad del

sistema B se reduce 1.41 veces, respecto al sistema A, mientras la velocidad del sistema C es 1.82 veces menor, respecto al sistema A y 1.29 veces menor, en comparación al sistema B.

La aceleración obtenida por los tres sistemas fue de 9.8 m/s^2 . Esto se debe a que la ecuación de la aceleración depende en gran medida de la amplitud del sistema, el cual se reduce a medida que se incrementa el número de resortes y por ende, que la aceleración permanezca constante.

Finalmente, el desarrollo de este código servirá como herramienta de análisis en sistemas que incluyan MAS y sobre todo, para determinar los parámetros que afectan su comportamiento. La ventaja del presente código, respecto a los softwares comerciales es que puede usarse de forma libre, ser de fácil acceso para los estudiantes y de bajo costo. Sin embargo, el software está limitado a determinar las deformaciones y fuerzas aplicadas a sistemas en paralelo de hasta cinco resortes, por lo que en trabajos futuros se pretende desarrollar nuevos módulos que permitan determinar las fallas más comunes que se presentan en los sistemas de vibración mecánica.

Referencias

Amrani, D. y Paradis, P. (2010). Use of Computer-Based Data Acquisition to Teach Physics Laboratories: Case study Simple harmonic Motion. *Latin American Journal of Physics Education*. 4(3), 511-514. Recuperada de <https://dialnet.unirioja.es/servlet/articulo?codigo=3696830>

Bongianino R., Cistac G. (2012). Simulador gráfico de movimiento armónico como instrumento para integrar conocimiento. *Revista iberoamericana de educación en tecnología y tecnología en educación*. No 7, 42-49. Recuperada de: <http://sedici.unlp.edu.ar/handle/10915/18287>

Fernández Y., Guariste M., Correa P. (2010). Medición de la aceleración de la gravedad: masa unida a un resorte. *Física experimental I*, 1-8. Recuperada de <https://users.exa.unicen.edu.ar/catedras/fisexp1.bak/files/2010%201-Fernandez-Guariste-Correa-Resorte-Informe.pdf>

García R. (2012). Creación e implementación de una unidad didáctica sobre la enseñanza del movimiento armónico simple a estudiantes de grado decimo. Tesis de maestría, Universidad Nacional de Colombia, Colombia. Recuperada de <https://repositorio.unal.edu.co/bitstream/handle/unal/11298/01186637.2012.pdf?sequence=1&isAllowed=y>

Gerez C. (2014). Resortes mecánicos helicoidales. Recuperada de: [Resortes_mecanicos_helicoidales_-_Carlos_Gerez_10-14-libre.pdf](https://resortes_mecanicos_helicoidales_-_Carlos_Gerez_10-14-libre.pdf) (d1wqxts1xzle7.cloudfront.net)

López M.A., Hernández J.A., Barroso L.A., Trujillo J.C. (2017). Computo simbólico y gráfico: estudio del sistema masa-resorte. *Revista brasileira de Ensino de Física*, 39 (2). Recuperada de <https://www.scielo.br/j/rbef/a/HxxJH3gsKtj5cWpNbBqST9P/?lang=es&format=html>

Maestre C., Alfaro E., Racebo F. (2015). Solución aproximada del oscilador armónico simple utilizando el método de las diferencias finitas. *Revista del programa de las matemáticas*, No 1, Vol 2, 59-64. Recuperada de:

<http://investigaciones.uniatlantico.edu.co/revistas/index.php/MATUA/article/view/1351>

Martínez E. (2013). Cálculo de resortes helicoidales de compresión. Tesis, Universidad Politécnica de Cartagena, Cartagena, Colombia. Recuperada de: <https://repositorio.upct.es/bitstream/handle/10317/3379/pfc5169.pdf>

Musik P. (2017). Development of computer-based experimntal set on simple harmonic motion of mass on springs. Journal of education technology, 16(4), 1-11. <https://files.eric.ed.gov/fulltext/EJ1160639.pdf>

Ríos V., Montero G., Román A., García A. (2017). Simulación experimental para la enseñanza del movimiento oscilatorio. Latin-American Journal of Physics Education Vol 11, No 1. Recuperada de: <https://dialnet.unirioja.es/servlet/articulo?codigo=6019783>

Scheck L.M. (2013). Transformada de Laplace: Resolución de sistema masa-resorte. Recuperada de <http://lcr.uns.edu.ar/fvc/NotasDeAplicacion/FVC-ScheckLucianoEmmanuel.pdf>

Tapia I.E. (2018). Aprendizaje significativo en el estudio de la constante elástica del resorte con recursos de bajo costo. Polo del conocimiento No 4, Vol 3, 163-184. Recuperada de <file:///C:/Users/user/Downloads/474-1144-2-PB.pdf>

White G. (2010). Introducción al análisis de vibraciones. Azima DLI, 300 TradeCenter, Suite 4610, Ma 01801 USA.

Yu A.M., Hao Y. (2011). Free vibration analysis of cylindrical helical springs with noncircular cross-sections. Journal of Sound and Vibration, 11, 2628-2639. <https://doi.org/10.1016/j.jsv.2010.12.015>

Apéndice

Código

```

clc;
clear all;
g=9.81;
t=0:0.01:10;
S=input('Qué tipo de sistema quieres resolver: 1)Serie, 2) Paralelo, 3) Mixto y 4) salir=')
while S==1
endwhile
while S==2
R=input('Cuántos resorte quieres resolver:2,3,4 O 5=')
F=input('Qué fuerza quieres aplicar=')
if R==2
k1=input('Cuál es el valor de k1=')
k2=input('Cuál es el valor de k2=')
Keq=k1+k2
xTotal=F/Keq
f1=xTotal*k1
f2=xTotal*k2
T=input('Deseas calcular y graficar: 0) Salir, 1)Posición, 2)Velocidad 3)Aceleración =')
while T==1
C=input('Cuentas con el valor de la masa: 1)Si o 2)No=')
if C==1
A=xTotal
m=input('ingrese el valor de la masa=')
w=sqrt(Keq/m)
fi=input('Defina el valor del ángulo de fase (fi)=')
Per=(2*pi)/w
frec=1/Per
plot(t,A*cos((w*t)+fi))
xlabel('Tiempo (s)',FontName,'Arial',FontSize,14)

```

```

ylabel('Posición (m)',FontName,'Arial',FontSize,14)
title('Gráfica de Posición')
endif
if C==2
A=xTotal
m=F/g
w=sqrt(Keq/m)
Per=(2*pi)/w
frec=1/Per
fi=input('Defina el valor de fi=')
plot(t,A*cos((w*t)+fi))
xlabel('Tiempo(s)',FontName,'Arial',FontSize,14)
ylabel('Posición(m)',FontName,'Arial',FontSize,14)
title('Gráfica de Posición')
endif
T=input('Deseas calcular y graficar: 0) Salir, 1)Posición, 2)Velocidad, 3)Aceleración=')
endwhile
while T==2
C=input('Cuentas con el valor de la masa: 1)Si o 2)No=')
if C==1
A=xTotal
m=input('Ingrese el valor de la masa=')
w=sqrt(Keq/m)
fi=input('Defina el valor de fi=')
Per=(2*pi)/w
frec=1/Per
plot(t,((-A*w)*sin((w*t)+fi)))
xlabel('Tiempo (s)',FontName,'Arial',FontSize,14)
ylabel('Velocidad (m/s)',FontName,'Arial',FontSize,14)
title('Grafica de Velocidad')
endif
if C==2
A=xTotal
m=F/g
w=sqrt(Keq/m)
Per=(2*pi)/w
frec=1/Per
fi=input('Defina el valor de fi=')
plot(t,((-A*w)*sin((w*t)+fi)))
xlabel('Tiempo (s)',FontName,'Arial',FontSize,14)
ylabel('Velocidad (m/s)',FontName,'Arial',FontSize,14)
title('Gráfica de Velocidad')
endif
T=input('Deseas calcular y graficar: 0) Salir, 1)Posición, 2)Velocidad, 3)Aceleración=')
endwhile
while T==3
C=input('Cuentas con el valor de la masa: 1)Si o 2)No=')
if C==1
A=xTotal
m=input('Ingrese el valor de la masa=')
w=sqrt(Keq/m)
fi=input('Defina el valor de fi=')
Per=(2*pi)/w
frec=1/Per
plot(t,((-w^2*A)*cos((w*t)+fi)))
xlabel('Tiempo (s)',FontName,'Arial',FontSize,14)
ylabel('Aceleración (m/s^2)',FontName,'Arial',FontSize,14)
title('Gráfica de Aceleración')
endif
if C==2
A=xTotal
m=F/g
w=sqrt(Keq/m)
Per=(2*pi)/w
frec=1/Per
fi=input('Defina el valor de fi=')
plot(t,((-w^2*A)*cos((w*t)+fi)))
xlabel('Tiempo (s)',FontName,'Arial',FontSize,14)
ylabel('Aceleración (m/s^2)',FontName,'Arial',FontSize,14)
title('Gráfica de Aceleración')
endif

```

```

T=input('Deseas calcular y graficar: 0) Salir, 1)Posición, 2)Velocidad,
3)Aceleración=')
endwhile
endif
if R==3
k1=input('Cuál es el valor de k1=')
k2=input('Cuál es el valor de k2=')
k3=input('Cuál es el valor de k3=')
Keq=k1+k2+k3
xTotal=F/Keq
f1=xTotal*k1
f2=xTotal*k2
f3=xTotal*k3
T=input('Deseas calcular y graficar: 0) Salir, 1)Posición, 2)Velocidad
3)Aceleración =')
while T==1
C=input('Cuentas con el valor de la masa: 1)Sí o 2)No=')
if C==1
A=xTotal
m=input('Ingrese el valor de la masa=')
w=sqrt(Keq/m)
fi=input('Defina el valor de fi=')
Per=(2*pi)/w
frec=1/Per
plot(t,A*cos((w*t)+fi))
xlabel('Tiempo (s)','FontName','Arial','FontSize',14')
ylabel('Posición (m)','FontName','Arial','FontSize',14')
title('Gráfica de Posición')
endif
if C==2
A=xTotal
m=F/g
w=sqrt(Keq/m)
Per=(2*pi)/w
frec=1/Per
fi=input('Defina el valor de fi=')
plot(t,A*cos((w*t)+fi))
xlabel('Tiempo (s)','FontName','Arial','FontSize',14')
ylabel('Posición (m)','FontName','Arial','FontSize',14')
title('Gráfica de Posición')
endif
T=input('Deseas calcular y graficar: 0) Salir, 1)Posición, 2)Velocidad,
3)Aceleración=')
endwhile
while T==2
C=input('Cuentas con el valor de la masa: 1)Sí o 2)No=')
if C==1
A=xTotal
m=input('Ingrese el valor de la masa=')
w=sqrt(Keq/m)
fi=input('Defina el valor de fi=')
Per=(2*pi)/w
frec=1/Per
plot(t,(-A*w)*sin((w*t)+fi))
xlabel('Tiempo (s)','FontName','Arial','FontSize',14')
ylabel('Velocidad (m/s)','FontName','Arial','FontSize',14')
title('Gráfica de Velocidad')
endif
if C==2
A=xTotal
m=F/g
w=sqrt(Keq/m)
Per=(2*pi)/w
frec=1/Per
fi=input('Defina el valor de fi=')
plot(t,(-A*w)*sin((w*t)+fi))
xlabel('Tiempo (s)','FontName','Arial','FontSize',14')
ylabel('Velocidad (m/s)','FontName','Arial','FontSize',14')
title('Gráfica de Velocidad')
endif
T=input('Deseas calcular y graficar: 0) Salir, 1)Posición, 2)Velocidad,
3)Aceleración=')
endwhile
while T==3
C=input('Cuentas con el valor de la masa: 1)Sí o 2)No=')
if C==1
A=xTotal
m=input('Ingrese el valor de la masa=')
w=sqrt(Keq/m)
fi=input('Defina el valor de fi=')
Per=(2*pi)/w
frec=1/Per
plot(t,((-w^2*A)*cos((w*t)+fi)))
xlabel('Tiempo (s)','FontName','Arial','FontSize',14')
ylabel('Aceleración (m/s^2)','FontName','Arial','FontSize',14')
title('Gráfica de Aceleración')
endif
if C==2
A=xTotal
m=F/g
w=sqrt(Keq/m)
Per=(2*pi)/w
frec=1/Per
fi=input('Defina el valor de fi=')
plot(t,((-w^2*A)*cos((w*t)+fi)))
xlabel('Tiempo (s)','FontName','Arial','FontSize',14')
ylabel('Aceleración (m/s^2)','FontName','Arial','FontSize',14')
title('Gráfica de Aceleración')
endif
T=input('Deseas calcular y graficar: 0) Salir, 1)Posición, 2)Velocidad,
3)Aceleración=')
endwhile
if R==4
k1=input('Cuál es el valor de k1=')
k2=input('Cuál es el valor de k2=')
k3=input('Cuál es el valor de k3=')
k4=input('Cuál es el valor de k4=')
Keq=k1+k2+k3+k4
xTotal=F/Keq
f1=xTotal*k1
f2=xTotal*k2
f3=xTotal*k3
f4=xTotal*k4
T=input('Deseas calcular y graficar: 0) Salir, 1)Posición, 2)Velocidad
3)Aceleración =')
while T==1
C=input('Cuentas con el valor de la masa: 1)Sí o 2)No=')
if C==1
A=xTotal
m=input('Ingrese el valor de la masa=')
w=sqrt(Keq/m)
fi=input('Defina el valor de fi=')
Per=(2*pi)/w
frec=1/Per
plot(t,A*cos((w*t)+fi))
ylabel('Tiempo (s)','FontName','Arial','FontSize',14')
xlabel('Posición (m)','FontName','Arial','FontSize',14')
title('Grafica de Posición')
endif
if C==2
A=xTotal
m=F/g
w=sqrt(Keq/m)
Per=(2*pi)/w
frec=1/Per
fi=input('Defina el valor de fi=')
plot(t,A*cos((w*t)+fi))
xlabel('Tiempo (s)','FontName','Arial','FontSize',14')
ylabel('Posición (m)','FontName','Arial','FontSize',14')
title('Gráfica de Posición')
endif
T=input('Deseas calcular y graficar: 0) Salir, 1)Posición, 2)Velocidad,
3)Aceleración=')
endwhile
while T==2
C=input('Cuentas con el valor de la masa: 1)Sí o 2)No=')
if C==1
A=xTotal
m=input('Ingrese el valor de la masa=')
w=sqrt(Keq/m)
fi=input('Defina el valor de fi=')
Per=(2*pi)/w
frec=1/Per
plot(t,A*cos((w*t)+fi))
ylabel('Tiempo (s)','FontName','Arial','FontSize',14')
xlabel('Posición (m)','FontName','Arial','FontSize',14')
title('Gráfica de Posición')
endif
if C==2
A=xTotal
m=F/g
w=sqrt(Keq/m)
Per=(2*pi)/w
frec=1/Per
fi=input('Defina el valor de fi=')
plot(t,A*cos((w*t)+fi))
xlabel('Tiempo (s)','FontName','Arial','FontSize',14')
ylabel('Posición (m)','FontName','Arial','FontSize',14')
title('Gráfica de Posición')
endif
T=input('Deseas calcular y graficar: 0) Salir, 1)Posición, 2)Velocidad,
3)Aceleración=')
endwhile
while T==3
C=input('Cuentas con el valor de la masa: 1)Sí o 2)No=')
if C==1

```

```

A=xTotal
m=input('Ingrese el valor de la masa=')
w=sqrt(Keq/m)
fi=input('Defina el valor de fi=')
Per=(2*pi)/w
frec=1/Per
plot(t,((-A*w)*sin((w*t)+fi)))
xlabel('Tiempo (s)','FontName','Arial','FontSize',14)
ylabel('Velocidad (m/s)','FontName','Arial','FontSize',14)
title('Gráfica de Velocidad')
endif
if C==2
A=xTotal
m=F/g
w=sqrt(Keq/m)
Per=(2*pi)/w
frec=1/Per
fi=input('Defina el valor de fi=')
plot(t,((-A*w)*sin((w*t)+fi)))
xlabel('Tiempo (s)','FontName','Arial','FontSize',14)
ylabel('Velocidad (m/s)','FontName','Arial','FontSize',14)
title('Gráfica de Velocidad')
endif
T=input('Deseas calcular y graficar: 0) Salir, 1)Posición, 2)Velocidad,
3)Aceleración=')
endwhile
while T==3
C=input('Cuentas con el valor de la masa: 1)Sí o 2)No=')
if C==1
A=xTotal
m=input('Ingrese el valor de la masa=')
w=sqrt(Keq/m)
fi=input('Defina el valor de fi=')
Per=(2*pi)/w
frec=1/Per
plot(t,((-w^2*A)*cos((w*t)+fi)))
ylabel('Tiempo (s)','FontName','Arial','FontSize',14)
xlabel('Aceleración (m/s^2)','FontName','Arial','FontSize',14)
title('Gráfica de Aceleración')
endif
if C==2
A=xTotal
m=F/g
w=sqrt(Keq/m)
Per=(2*pi)/w
frec=1/Per
fi=input('Defina el valor de fi=')
plot(t,((-w^2*A)*cos((w*t)+fi)))
xlabel('Tiempo (s)','FontName','Arial','FontSize',14)
ylabel('Aceleración (m/s^2)','FontName','Arial','FontSize',14)
title('Gráfica de Aceleración')
endif
T=input('Deseas calcular y graficar: 0) Salir, 1)Posición, 2)Velocidad,
3)Aceleración=')
endwhile
endif
if R==5
k1=input('Cuál es el valor de k1=')
k2=input('Cuál es el valor de k2=')
k3=input('Cuál es el valor de k3=')
k4=input('Cuál es el valor de k4=')
k5=input('Cuál es el valor de k5=')
Keq=k1+k2+k3+k4+k5
xTotal=F/Keq
f1=xTotal*k1
f2=xTotal*k2
f3=xTotal*k3
f4=xTotal*k4
f5=xTotal*k5
T=input('Deseas calcular y graficar: 0) Salir, 1)Posición, 2)Velocidad
3)Aceleración =')
while T==1
C=input('Cuentas con el valor de la masa: 1)Sí o 2)No=')
if C==1
A=xTotal
m=input('Ingrese el valor de la masa=')
w=sqrt(Keq/m)
fi=input('Defina el valor de fi=')
Per=(2*pi)/w
frec=1/Per
plot(t,A*cos((w*t)+fi))
xlabel('Tiempo (s)','FontName','Arial','FontSize',14)
ylabel('Posición (m)','FontName','Arial','FontSize',14)
title('Gráfica de Posición')
endif
if C==2
A=xTotal
m=F/g
w=sqrt(Keq/m)
Per=(2*pi)/w
frec=1/Per
fi=input('Defina el valor de fi=')
plot(t,A*cos((w*t)+fi))
xlabel('Tiempo (s)','FontName','Arial','FontSize',14)
ylabel('Posición (m)','FontName','Arial','FontSize',14)
title('Gráfica de Posición')
endif
T=input('Deseas calcular y graficar: 0) Salir, 1)Posición, 2)Velocidad,
3)Aceleración=')
endwhile
while T==2
C=input('Cuentas con el valor de la masa: 1)Sí o 2)No=')
if C==1
A=xTotal
m=input('Ingrese el valor de la masa=')
w=sqrt(Keq/m)
fi=input('Defina el valor de fi=')
Per=(2*pi)/w
frec=1/Per
plot(t,((-A*w)*sin((w*t)+fi)))
xlabel('Tiempo (s)','FontName','Arial','FontSize',14)
ylabel('Velocidad (m/s)','FontName','Arial','FontSize',14)
title('Gráfica de Velocidad')
endif
if C==2
A=xTotal
m=F/g
w=sqrt(Keq/m)
Per=(2*pi)/w
frec=1/Per
fi=input('Defina el valor de fi=')
plot(t,((-w^2*A)*cos((w*t)+fi)))
xlabel('Tiempo (s)','FontName','Arial','FontSize',14)
ylabel('Velocidad (m/s)','FontName','Arial','FontSize',14)
title('Gráfica de Velocidad')
endif
T=input('Deseas calcular y graficar: 0) Salir, 1)Posición, 2)Velocidad,
3)Aceleración=')
endwhile
while T==3
C=input('Cuentas con el valor de la masa: 1)Sí o 2)No=')
if C==1
A=xTotal
m=input('Ingrese el valor de la masa=')
w=sqrt(Keq/m)
fi=input('Defina el valor de fi=')
Per=(2*pi)/w
frec=1/Per
plot(t,((-w^2*A)*cos((w*t)+fi)))
xlabel('Tiempo (s)','FontName','Arial','FontSize',14)
ylabel('Aceleración (m/s)','FontName','Arial','FontSize',14)
title('Gráfica de Aceleración')
endif
if C==2
A=xTotal
m=F/g
w=sqrt(Keq/m)
Per=(2*pi)/w

```

```
frec=1/Per
fi=input('Defina el valor de fi=')
plot(t,((-w^2*A)*cos((w*t)+fi)))
xlabel('Tiempo (s)',FontName,'Arial',FontSize,14)
ylabel('Aceleracion (m/s^2)',FontName,'Arial',FontSize,14)
title('Grafica de Aceleracion')
endif
T=input('Deseas calcular y graficar: 0) Salir, 1)Posicion, 2)Velocidad,
3)Aceleracion=')
endwhile
endif
S=input('Que tipo de sistema quieres resolver: 1)Serie, 2) Paralelo, 3)
Mixto y 4) salir=')
endwhile
```