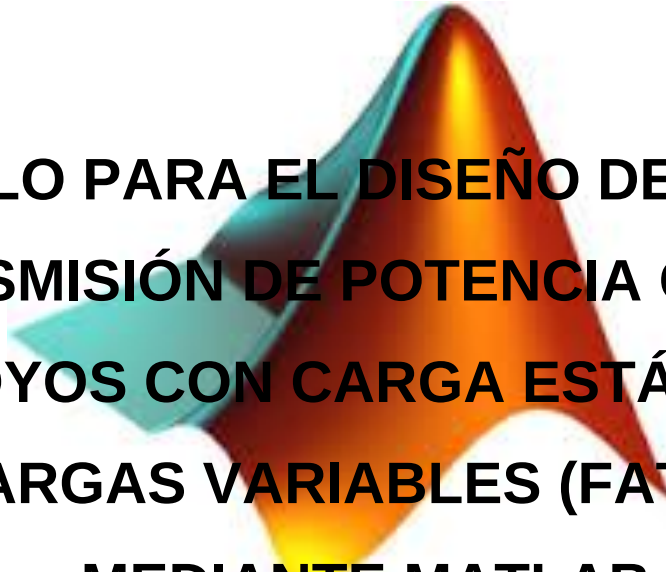
NSTITUTO TECNOLÓGICO DE TUXTLA GUTIÉRREZ

RESIDENCIA PROFESIONAL



MODULO PARA EL DISEÑO DE EJES DE TRANSMISIÓN DE POTENCIA CON DOS APOYOS CON CARGA ESTÁTICA Y CARGAS VARIABLES (FATIGA) MEDIANTE MATLAB.

INTEGRANTES:

CARLOS DE JESÚS CARRIÓN INDILÍ

LUIS ÁNGEL LÓPEZ RODRÍGUEZ

INDICE

INTRODUCCION 4
JUSTIFICACION 5
OBJETIVOS 6
ALCANCES Y LIMITACIONES 7
Alcances 7
Limitaciones 7
- **Cargas estáticas** 7
- **Cargas variables** 8

PROBLEMAS A RESOLVER 8
FUNDAMENTO TEORICO 9
Criterios de diseño de ejes 9
CONCENTRADORES DE ESFUERZO 11
TEORÍA DEL ESFUERZO CORTANTE MÁXIMO (TECM). 18
TEORÍA DE LA ENERGÍA DE DISTORSIÓN (TED). 19
TEORÍAS Y CRITERIOS DE FALLA POR CARGAS DINÁMICAS 20
CÁLCULO DE LA PRIMERA VELOCIDAD CRÍTICA DE UN EJE ESCALONADO CON DOS APOYOS SIMPLES 27
CÁLCULO DEL DIAMETRO CONSTANTE DE UN EJE 30
Ecuación para el cálculo del diámetro del eje 33
MATLAB 34
Como entrar y salir de MatLab 34
Para obtener ayuda 34
Operadores 35
Variables en MatLab 35
ACERCA DE GUIDE 36
Archivos .m 36
Control de flujo 40
Gráficos 42
Otros aspectos interesantes 46
FUNCIONAMIENTO DE LA PROGRAMACION DEL CÁLCULO DEL DIAMETRO RECOMENDADO PARA UN EJE ESCALONADO PARA CARGAS VARIABLES Y UN EJE CON DIAMETRO CONSTANTE PARA CARGAS ESTÁTICAS. 48

CÁLCULO DEL DIÁMETRO RECOMENDADO PARA UN EJE CON DIÁMETROS VARIABLES Y TRES CARGAS PARA CARGAS VARIABLES. 52
Generalidades del programa 59
CONCLUSIONES 63
ANEXOS 64
Validación del programa 64
BIBLIOGRAFIA 74

INTRODUCCION

La residencia profesional es una herramienta que permite consolidar la formación académica y profesional de un alumno, es también un factor estratégico en la tarea de impulsar el desarrollo, así como para mejorar los conocimientos que conducen a que los demás compañeros puedan reafirmar sus resultados en base a las investigaciones.

Todo esto contribuye en la formación integral del estudiante, el desarrollar valores favorecen la inserción al mercado de trabajo y la aplicación del conocimiento científico, humanístico, tecnológico al promover el acercamiento activo de la universidad con la sociedad; al mismo tiempo, coadyuva en la solución de los problemas actuales del desarrollo.

En la actualidad vivimos en un mundo en donde los avances científicos y tecnológicos requieren de innovación, rapidez y precisión, donde el tiempo es el principal reto a vencer.

El presente proyecto tiene la finalidad de aplicar los conocimientos adquiridos durante el periodo de la carrera de Ing. Mecánica para luego ponerlos en práctica y de esta manera poder realizar un software que calcule el diámetro que requiere un eje con dos apoyos bajo especificaciones dadas, tanto para cargas estáticas como para cargas variables.

Con este software se pretende facilitar los cálculos que conlleva realizar la búsqueda del diámetro recomendado del eje que estará sometido a cargas, tanto estáticas como variables, ya que dichos cálculos representan horas de trabajo lo cual nos puede llevar a tener errores de sustitución, con el uso del equipo de cómputo y del programador podemos reducir las horas de trabajo y el margen de error.

JUSTIFICACION

La vida cotidiana nos trae retos por lo cual debemos afrontarlos y buscar día a día las posibles soluciones. El ser un Ing. Mecánico en muchas ocasiones debemos realizar cálculos matemáticos y nos resulta laborioso y hasta complicado, por lo tanto nos vemos en la necesidad de buscar opciones los cuales nos faciliten los procedimientos, por lo tanto nosotros tomamos los siguientes puntos:

- Creamos un software especializado que dé solución a este problema.
- Analizar los datos establecidos en el problema.
- Analizar el manejo del programa.
- Informar al usuario que manejará el software

OBJETIVOS

- Analizar que los alumnos tengan un buen dominio del software.
- Desarrollar un programa que ayude a corroborar y a certificar los resultados del cálculo del diámetro requerido bajo ciertas condiciones iniciales.
- Ayudar a que el usuario se le facilite el resultado de una forma más sencilla.
- Disminuir el margen de error en los resultados.
- Disminuir el tiempo de cálculo.

ALCANCES Y LIMITACIONES

Alcances

- El desarrollo de este software permite llevar a la práctica conocimientos adquiridos a través de las asignaturas del área de Ingeniería Mecánica.
- La utilización de este programa de software que se desarrolló en este proyecto permite un mínimo de variación en nuestros resultados de acuerdo a los cálculos tradicionales.
- Los resultados son obtenidos de forma más rápida.
- Facilita el proceso del cálculo del problema.
- La forma de insertar los datos es simple y sencilla.

Limitaciones

- **Cargas estáticas**

- El software solo calcula un eje con un máximo de tres masas.
- Para la obtención del diámetro recomendado, el diámetro del eje debe ser constante.
- Solo calcula el diámetro recomendado para ejes simplemente apoyados, donde los apoyos están al principio y al final del eje

- **Cargas variables**

- El software solo calcula un eje con un máximo de tres masas.
- El software realiza el cálculo hasta con 2 cambios de diámetro.
- Solo calcula el diámetro recomendado para ejes simplemente apoyados, donde los apoyos están al principio y al final del eje

PROBLEMAS A RESOLVER

Como ya hemos mencionado este programa ayudara tanto a los estudiantes como a los ingenieros a hacer de manera más rápida y sencilla el cálculo del diámetro recomendado, ahorrando mucho tiempo ya que este cálculo es muy tardado y más cuando se trata de un eje escalonado, también este programa nos hará una pequeña gráfica de como se ve el eje con los datos que se les proporcione.

Por otra parte este programa también puede encontrar el diámetro de un eje, siempre y cuando este eje sea de diámetro constante para el caso estático o con un máximo de tres diámetros para el caso dinámico, también ahorraremos mucho tiempo y lo podremos resolver de una manera muy sencilla y rápida con solo ingresar los datos que normalmente se nos proporciona, además de que nos dará una pequeña visualización de cómo se vería el eje ya con el diámetro encontrado.

FUNDAMENTO TEORICO

Un eje de transmisión es un elemento de sección circular cuya función es la de transmitir movimiento y potencia. La transmisión del movimiento se realiza a través de otros elementos tales como engranes, poleas, cadenas, etc.

Diseñar un eje consiste básicamente en la determinación del diámetro correcto del eje para asegurar una rigidez y una resistencia satisfactorias, cuando el eje transmite potencia bajo diferentes condiciones de carga.

Criterios de diseño de ejes

El diseño de un eje debe estudiarse a partir de los siguientes puntos de vista:

1. Análisis por resistencia.
    - Bajo cargas estáticas.
    - Bajo cargas dinámicas.
2. Análisis por rigidez.
    - Cálculo de deformaciones.
    - Velocidades críticas.

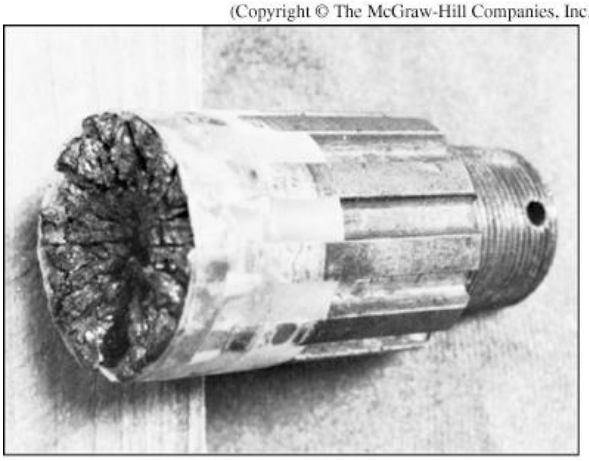
A nosotros lo que nos interesa es el análisis por resistencia.

Una carga estática es una fuerza estacionaria o un par de torsión que se aplica a un elemento.
Para ser estacionaria, la fuerza o el par de torsión no deben cambiar su magnitud, ni el punto de o los puntos de aplicación, ni de dirección. Una carga estática produce tensión o compresión axial, una carga cortante, una carga flexionante, una carga torsional o cualquier combinación de estas.

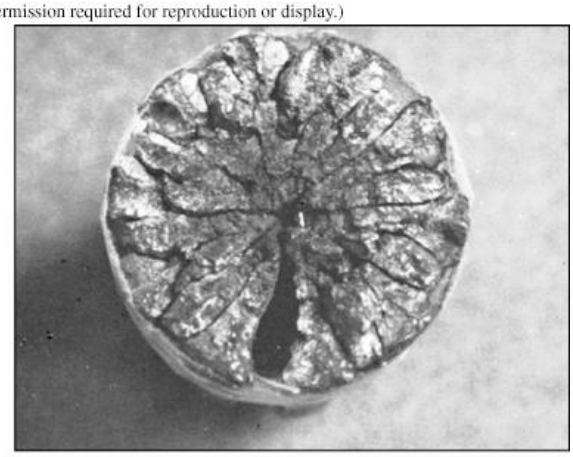
Para que se considere estática, la carga no pude cambiar de ninguna manera.

La falla puede significar que una parte de un elemento se ha separado en dos o más piezas; se ha distorsionado permanentemente, arruinando de esta de esta manera su geometría; se ha degradado su confiabilidad; o se ha comprometido su función, por cualquier razón. Cuando un diseñador habla de falla puede referirse a cualquiera o todas estas posibilidades. Las fotografías siguientes ilustran algunas fallas.

(Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.)

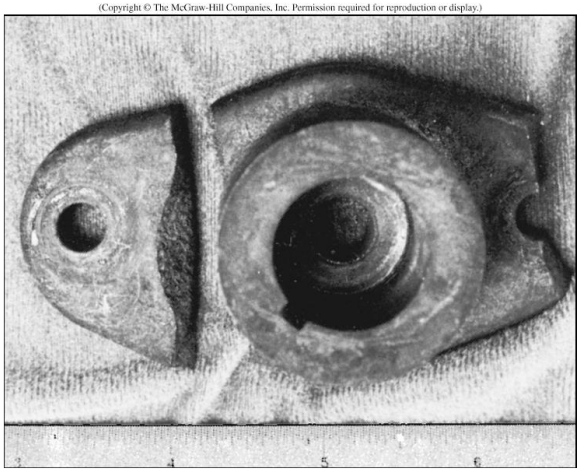


(a)

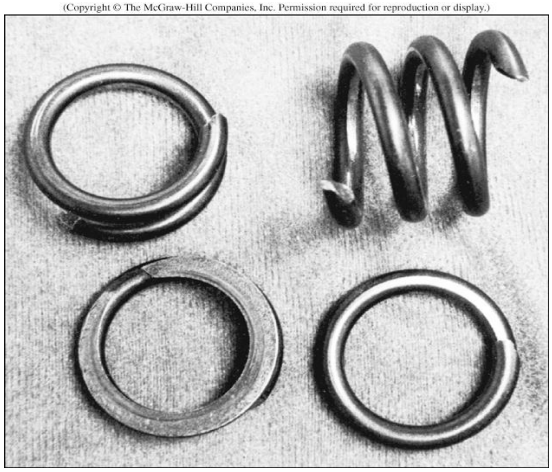


(b)

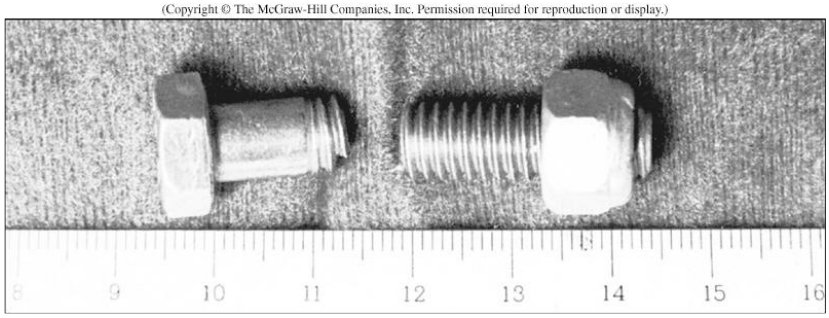
(Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.)



(Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.)



(Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.)



Se han propuesto varios criterios teóricos con el objeto de obtener una correlación adecuada entre la vida o duración estimada del componente y la que realmente se logra en las condiciones de carga de servicio para aplicaciones tanto en materiales frágiles como dúctiles.

Para materiales dúctiles se tienen (criterios de fluencia):

a).- Teoría del esfuerzo cortante máximo (Criterio de fluencia de Tresca).

b).- Teoría de la energía de distorsión (Von Mises-Hencky).

CONCENTRADORES DE ESFUERZO

Las causas de concentraciones de esfuerzos (llamadas también elevadores de esfuerzos) son muy variadas y numerosas. Se deben principalmente a acabados superficiales, inclusiones no metálicas y a otras causas.

Una concentración de esfuerzos es cualquier condición que causa que el esfuerzo local sea mayor que el esfuerzo nominal.

La geometría o forma del espécimen, es uno de los factores más importantes que contribuyen a la concentración del esfuerzo con bases muy racionales.

En el caso de que no sea posible el uso de técnicas analíticas, se usan métodos experimentales (fotoelasticidad, extensómetros, recubrimientos frágiles), en los cuales se prueban elementos reales que proporcionan datos de mucha utilidad para el diseño. Actualmente se continúan los trabajos con el fin de obtener dichos factores para el caso de esfuerzos combinados. A continuación podemos observar en las figuras (1) y (2) como se distribuyen los esfuerzos:

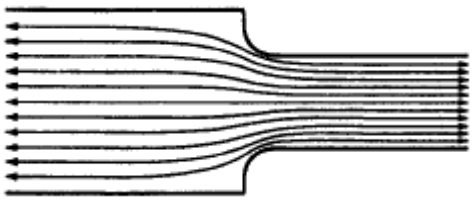


Figura (1).- Concentración de esfuerzos causada por un cambio repentino en la sección transversal.

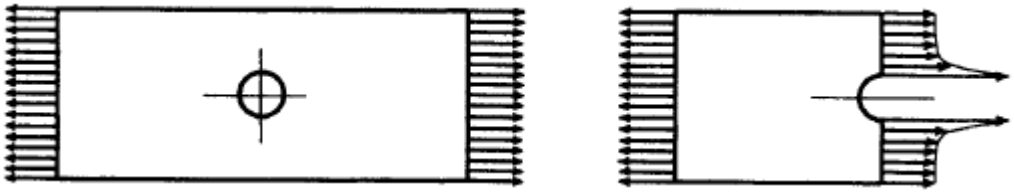


Figura (2).- Concentración de esfuerzos para una barra cargada en tensión y con un agujero.

El factor de concentración de esfuerzos está definido por la relación:

$$K_t = \frac{\sigma_{\max}}{\sigma_o} \quad \text{---------- (1)}$$

K_t = factor de concentración de esfuerzos teórico, dado solamente de acuerdo a lageometría del espécimen.

$\sigma_{máx}$ = esfuerzo normal máximo en la sección de interés.

σo = esfuerzo nominal en la sección de interés.

Para cortante se tiene:

$$K_{ts} = \frac{\tau_{\max}}{\tau_o} \quad \text{---------- (2)}$$

K_{ts} = factor teórico para corte.

$\tau_{máx}$. = esfuerzo cortante máximo en la sección de interés.

$\tau\ o$ = esfuerzo cortante nominal en la sección de interés.

Los factores de concentración de esfuerzos para diferentes condiciones de carga se muestran en las figuras (4 a 10).
Si todos los materiales fueran uniformemente homogéneos y estuvieran libres sus superficies de rayas o marcas, podría justificarse usar K_t "tal cual" para el cálculo de esfuerzos por fatiga, ya que éste depende solamente de su geometría. Sin embargo, los materiales no son homogéneos y en la superficie no están libres de defectos.
Las pruebas de fatiga han demostrado que el factor teórico de concentración de esfuerzos raramente se obtiene (excepto para algunos aceros de alta resistencia). En su lugar se utiliza un valor menor que K_t. Por lo tanto es necesario definir un factor de concentración de esfuerzosdebido a la fatiga, designado por *K f.*

$$K_f = \frac{\sigma_{\max}(con \quad muesca)}{\sigma_{\max(\text{sin} \quad muesca)}} \quad \text{---------- (3)}$$

$\sigma_{máx}$ = esfuerzo máximo por carga fluctuante con muesca o sin muesca.
Debido a que se requiere hacer un número ilimitado de pruebas para generar valores de K_f, es muy deseable relacionar el valor teórico de K_t con el de fatiga K_f para diferentes tamaños demuesca, materiales y tratamientos térmicos.
R.E. Peterson sugiere la siguiente relación conocida como factor *q* de sensibilidad a la muesca:

$$q = \frac{K_f - 1}{K_t - 1} \quad \text{---------- (4)}$$

O por torsión o corte

$$q = \frac{K_{fs} - 1}{K_{ts} - 1}$$ ---------- (5)

Resolviendo tenemos:

$$K_f = 1 + q(K_t - 1)$$ ---------- (6)

$$K_{fs} = 1 + q(K_{ts} - 1)$$ ---------- (7)

La sensibilidad de las muescas *q* también se puede definir a partir de la fórmula de **Kun-Hardrath**, en función de la constante de **Neuber** y del radio de la muesca *r*, ambos expresadosen pulgadas.

$$q = \frac{1}{1 + \frac{\sqrt{a}}{\sqrt{r}}}$$ ---------- (8)

Valores de *a* se representan en la siguiente tabla para aceros.

S_{ut} (kpsi)	$\sqrt{a}$ (pul$^{0.5}$)
50	0.130
55	0.118
60	0.108
70	0.093
80	0.080
90	0.070
100	0.062
110	0.055
120	0.049
130	0.044
140	0.039
160	0.031
180	0.024
200	0.018
220	0.013
240	0.009

Tabla 1.- Valores de $\sqrt{a}$

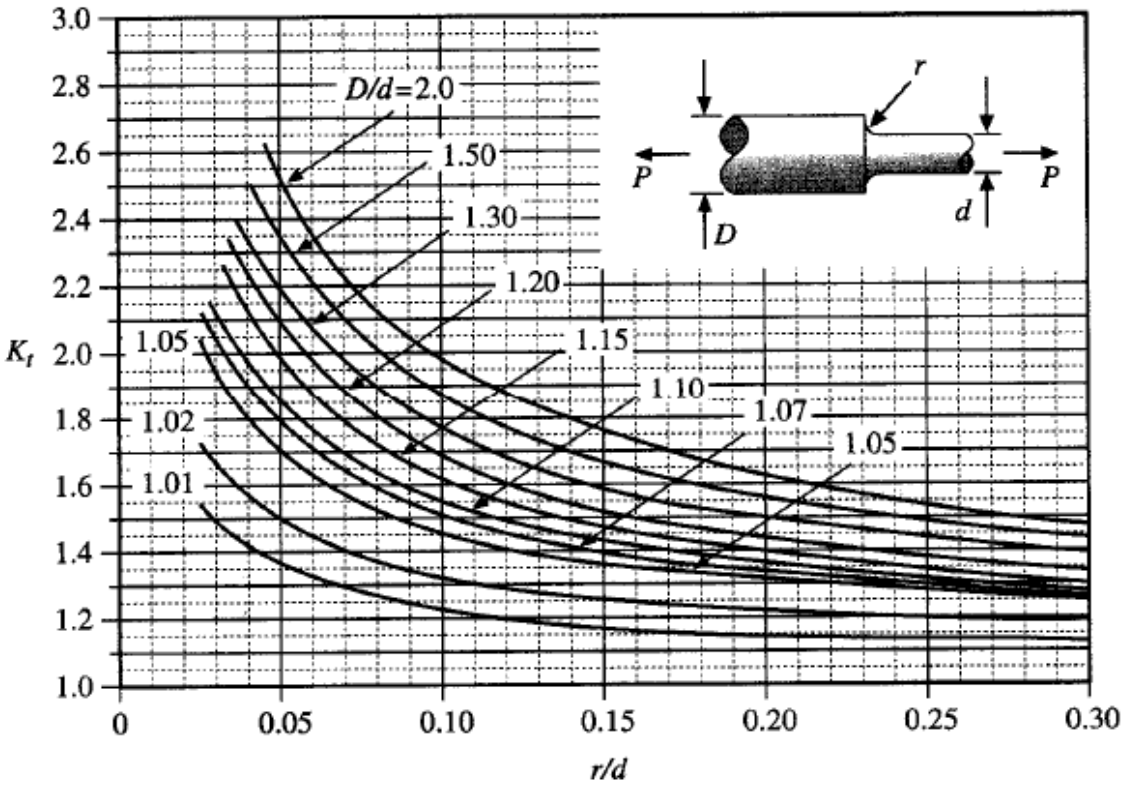


Figura (3).- Factor de concentración de esfuerzos para un eje con filete en tensión axial.

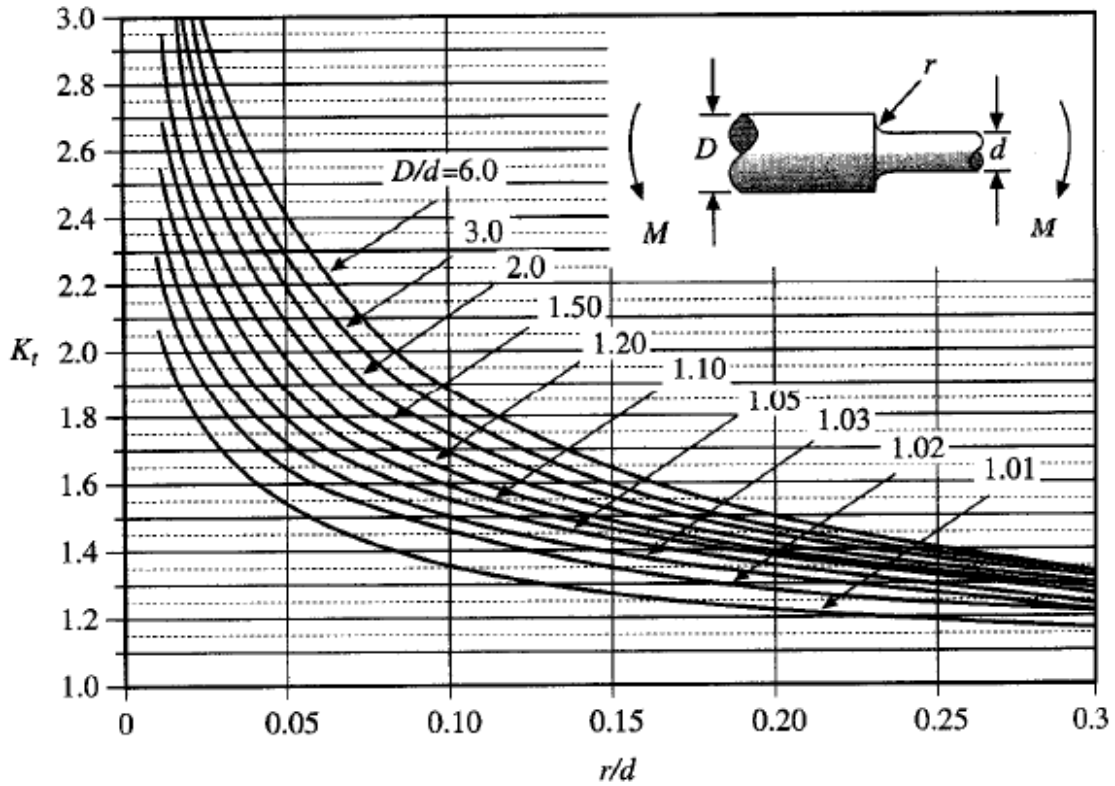


Figura (4).- Factor de concentración de esfuerzos para un eje con un filete en flexión.

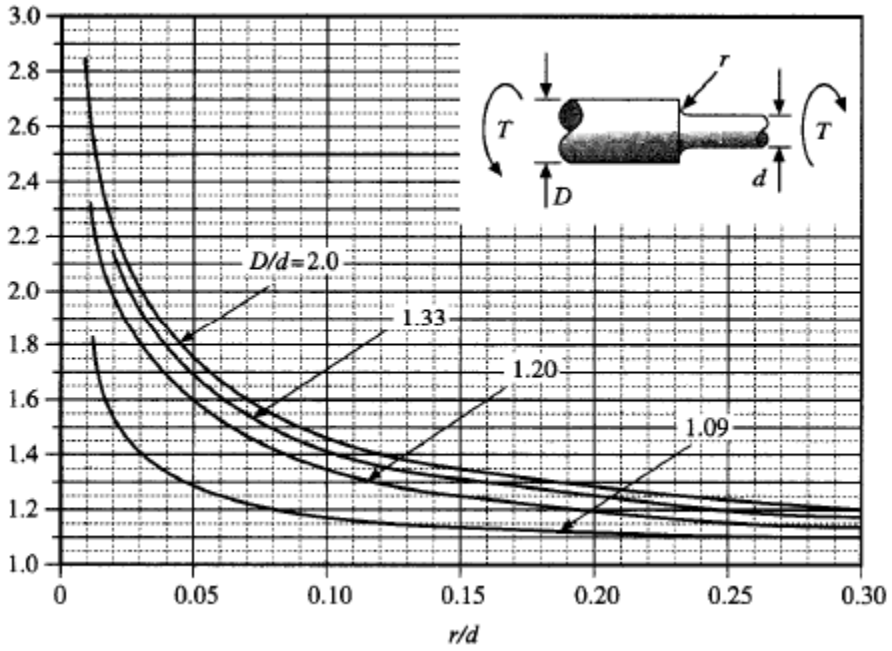


Figura (5).- Factor de concentración de esfuerzos para un eje con un filete en torsión.

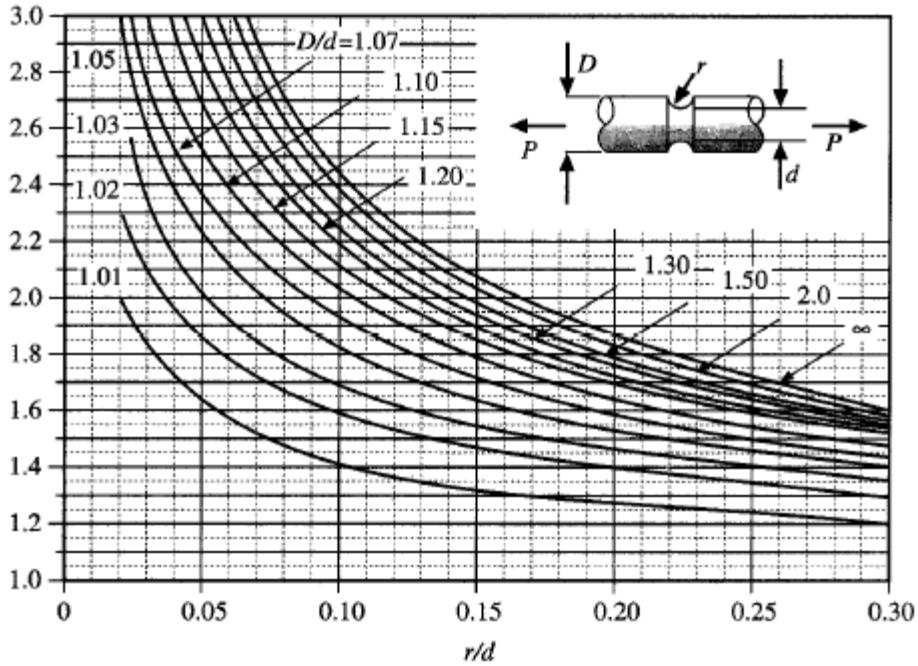


Figura (6).- Factor de concentración de esfuerzos para un eje con una ranura en tensión axial.

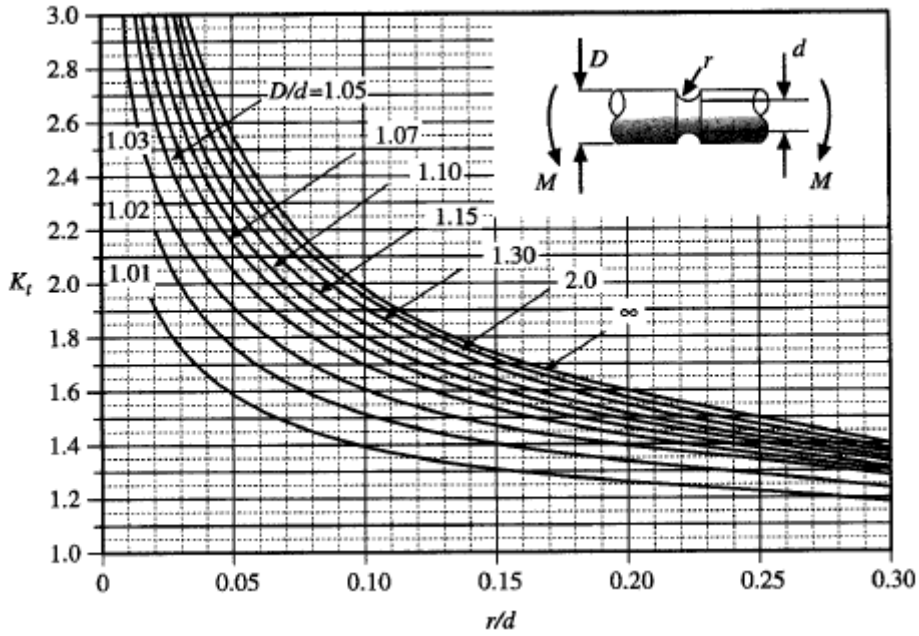


Figura (7).- Factor de concentración de esfuerzos para un eje con una ranura en flexión.

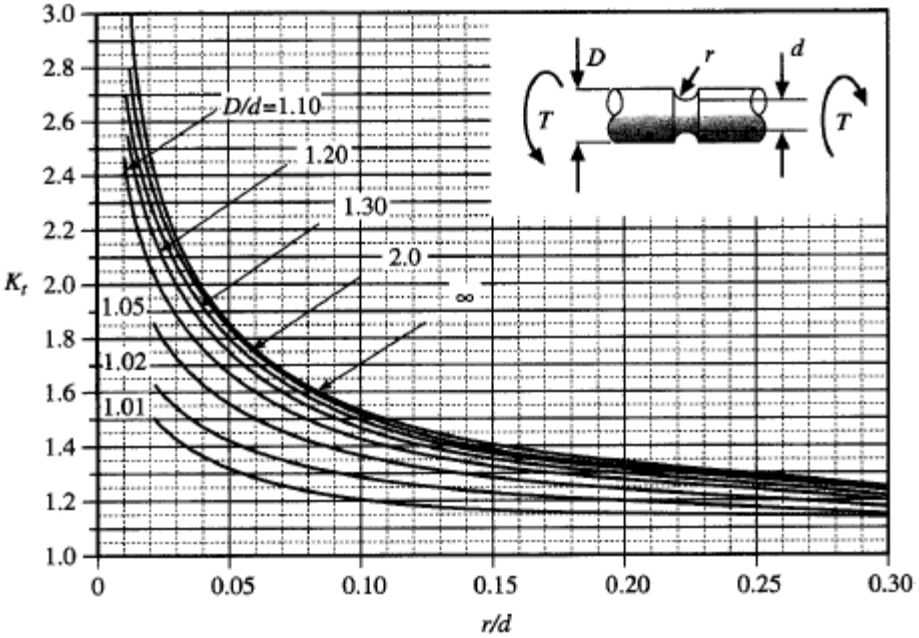


Figura (8).- Factor de concentración de esfuerzos para un eje con una ranura en torsión.

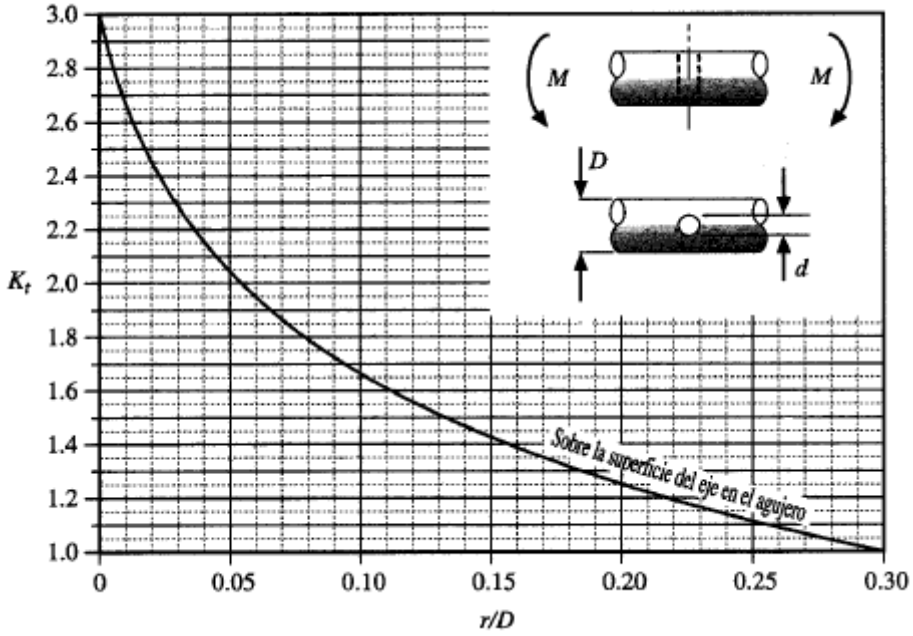


Figura (9).- Factor de concentración de esfuerzos para un eje con agujero transversal en flexión.

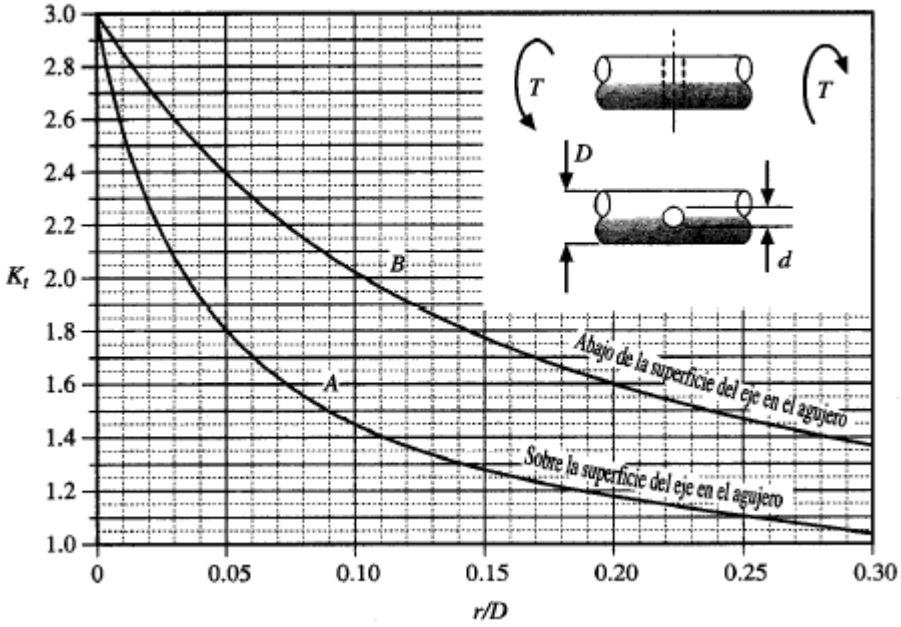


Figura (10).- Factor de concentración de esfuerzos para un eje con un agujero transversal en torsión.

TEORÍA DEL ESFUERZO CORTANTE MÁXIMO (TECM).

Esta teoría expresa que la falla en una pieza sujeta a un estado multiaxial de esfuerzos, ocurrirá cuando el esfuerzo cortante máximo desarrollado en ella, iguale o exceda al esfuerzo cortante máximo correspondiente al momento de la falla en el ensayo de tensión simple, efectuado con una probeta del mismo material.

Si la nomenclatura $\sigma_1 \geq \sigma_2 \geq \sigma_3$ se usa para los esfuerzos principales, el esfuerzo cortante máximo dice que la fluencia ocurrirá cuando:

$$\tau_{\max} \geq \frac{S_y}{2} \quad \text{---------- (9)}$$

O bien, recordando que $\sigma_1 - \sigma_3 = 2\tau_{máx}$, tenemos:

$$\sigma_1 - \sigma_3 \geq S_y \quad \text{---------- (10)}$$

Donde:
$S_y =$ Resistencia a la fluencia del material
El esfuerzo seguro de acuerdo con esta teoría será:

$$\boxed{\sigma_1 - \sigma_3 = \frac{S_y}{n_s}} \quad \text{---------- (11)}$$

Para un estado de **esfuerzos biaxial**, donde $\sigma_3 = 0$, tenemos:

$$\sigma_1 - \sigma_2 = \frac{S_y}{n_s} \quad \text{---------- (12)}$$

$n_s =$ Factor de seguridad.

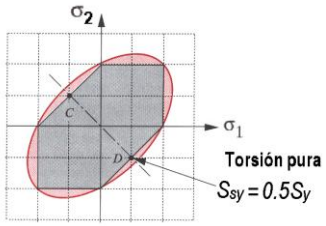


Figura (11).- Representación gráfica de la teoría del esfuerzo cortante máximo (TECM) Para un estado de esfuerzos biaxial.

TEORÍA DE LA ENERGÍA DE DISTORSIÓN (TED).

Esta teoría postula que la falla es causada por la energía elástica asociada con la deformación por cortante.
El esfuerzo cortante octaédrico esta dado por:

$$\tau_{oct} = \frac{1}{3}\sqrt{(\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_3 - \sigma_1)^2} \quad \text{---------- (13)}$$

El esfuerzo octaédrico producido por una tensión uniaxial $(\sigma_2 = \sigma_3 = 0)$ es:

$$\tau_{oct} = \frac{\sqrt{2}}{3}\sigma_1 \quad \text{---------- (14)}$$

El esfuerzo octaédrico máximo ocurre en:

$$(\tau_{oct})_{límite} = \frac{\sqrt{2}}{3}\sigma_e \quad \text{---------- (15)}$$

Donde $S_e =$ esfuerzo de Von Mises.
Para un estado de esfuerzos triaxial:

$$\sigma_e = \frac{1}{\sqrt{2}}\left[(\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_3 - \sigma_1)^2\right]^{1/2} \quad \text{---------- (16)}$$

Para un estado de esfuerzos biaxial, suponiendo $\sigma_3 = 0$

$$\sigma_e = (\sigma_1^2 + \sigma_2^2 - \sigma_1\sigma_2)^{1/2} \quad \text{---------- (17)}$$

En función de los esfuerzos aplicados, para el caso biaxial:

$$\sigma_e = \sqrt{\sigma_x^2 + \sigma_y^2 - \sigma_x\sigma_y + 3\tau_{xy}^2} \quad \text{---------- (18)}$$

De esta forma, la teoría de la energía de distorsión predice la falla si

$\sigma_e \geq S_y$ ---------- (19)

El esfuerzo seguro con dicha teoría será:

$\sigma_e = \frac{S_y}{n_s}$ ---------- (20)

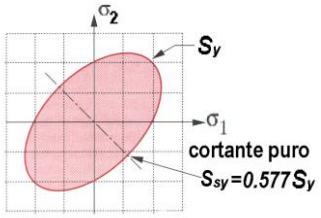


Figura (12).- Representación gráfica de la teoría de la energía de distorsión (TED) para un estado de esfuerzos biaxial.

TEORÍAS Y CRITERIOS DE FALLA POR CARGAS DINÁMICAS

CARGAS DINÁMICAS.

Existe una condición en la cual las cargas varían o fluctúan entre ciertos niveles con el tiempo, estas clases de carga que ocurren en elementos de máquinas producen esfuerzos que se llaman esfuerzos variables, repetidos, alternantes o fluctuantes.

La mayoría de las fallas de los elementos de máquinas implican condiciones de carga que varían con el tiempo. Sin embargo las condiciones de carga estática analizadas anteriormente son muy importantes, pues proporcionan la base para comprender el tema de la fatiga.

FATIGA.

A menudo, se encuentra que los elementos de máquinas han fallado bajo la acción de esfuerzos repetidos; no obstante, el análisis revela que los esfuerzos máximos estuvieron por debajo de la resistencia última del material y con mucha frecuencia incluso por debajo de la resistencia de fluencia. La característica más notable de estas fallas consiste en que los esfuerzos se repitieron un gran número de veces. Por lo tanto, a la falla se le llama falla por fatiga.

Cuando un elemento falla estáticamente, por lo general desarrollan una deflexión muy grande, puesto que el esfuerzo sobrepasó el limite elástico; por ello, la parte se remplaza antes de que en realidad ocurra la fractura.

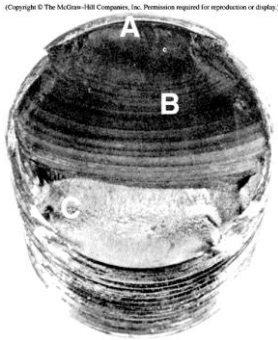
De esta manera la falla estática proporciona una advertencia visible. Pero una falla por fatiga no proporciona una advertencia, es repentina y total y, por ende, peligrosa. La fatiga es un fenómeno complejo, consistente en la propagación de grietas en una micro escala al principio, y luego muy rápida a medida que las grietas alcanzan una longitud crítica. Tres factores se requieren para que se dé una falla por fatiga:

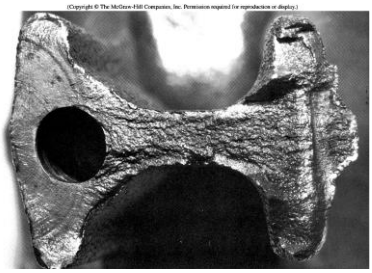
1.- Un esfuerzo de tensión suficientemente grande.

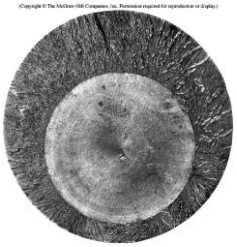
2.- Una variación de esfuerzos de suficiente amplitud.

3.- Un número de ciclos de aplicación de la carga suficientemente elevado.

Existen otros factores tales como: concentración de esfuerzos, corrosión, esfuerzos residuales, esfuerzos combinados, etc., que pueden alterar la resistencia a la fatiga del elemento. En las fotos siguientes se ilustran estas fallas:







LÍMITE DE RESISTENCIA A LA FATIGA

Para determinar la resistencia de materiales bajo la acción de cargas de fatiga, las muestras se someten afuerzas repetidas o variables de magnitudes especificadas, mientras se cuentan los ciclos o inversiones delesfuerzo hasta su destrucción. El dispositivo de ensayo a la fatiga que se emplea con mayor frecuencia es la **máquina de viga rotatoria de alta velocidad de R. R. Moore**. El primer ensayo se hace con un esfuerzo quees un poco menor que la resistencia ultima del material. El segundo ensayo se realiza con un esfuerzo que esmenor que el primero. Este proceso se continua y los resultados se grafican como la resistencia a la fatiga $\mathbf{S_f}$ contra el logaritmo del número total de ciclos a la falla ***N***, para cada espécimen. Estas gráficas se llaman
"**diagramas S-N**".

A continuación se representa un diagrama S-N para cierto acero y la probeta respectiva:

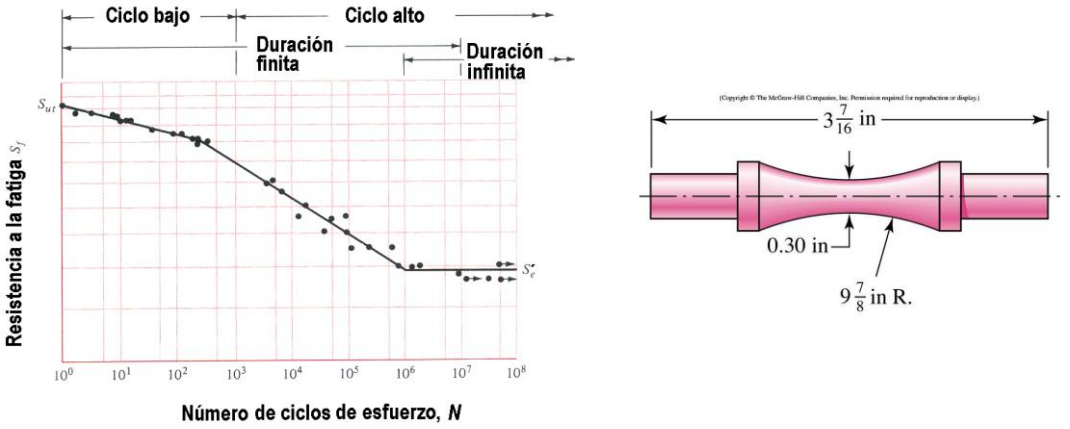


Figura (13).- Diagrama S-N trazado a partir de los resultados de pruebas de fatiga axial con inversión completa.

Como se puede observar, la grafica se hace horizontal después de que el material se sometió a esfuerzosdurante un cierto número de ciclos y para el caso de los aceros y más allá de este cambio no ocurrirá la falla,sin importar qué tan grande sea el número de ciclos. La resistencia correspondiente al cambio en la gráfica se llama **límite de resistencia Se** o **limite de fatiga**. La grafica nunca se hace horizontal para los metales noferrosos y aleaciones.

En la actualidad, determinar los limites de fatiga mediante ensayos a la fatiga es una rutina, aunque resulta unprocedimiento extenso y costoso, por lo que se usa una primera aproximación de **Se** usando **S'e** (Límite de resistencia a la fatiga en una probeta de viga rotatoria, de la grafica S-N se observa que para el caso de los aceros **S'e** es:

$$S'_e = \begin{cases} 0.5S_{ut} & S_{ut} \le 200\ kpsi\ (1400MPa) \\ 100\ kpsi & S_{ut} \ge 200\ kpsi \\ 700MPa & S_{ut} > 1400MPa \end{cases} \quad \text{---------- (21)}$$

FACTORES QUE MODIFICAN LA RESISTENCIA A LA FATIGA

Las muestras para el ensayo en maquina rotativa en el laboratorio, se preparan con mucho cuidado y seensayan bajo condiciones muy controladas. **Marín** identificó efectos de la condición superficial, el tamaño, la carga, la temperatura y varios otros puntos y propone la siguiente expresión para encontrar **Se** a partir **S'e**:

$$S_e = k_a k_b k_c k_d k_e S'_e$$ ---------- (22)

$S'e$=Límite de resistencia a la fatiga de la muestra de viga rotatoria.
Se = Límite de resistencia a la fatiga en el elemento mecánico.
ka= Factor de acabado de la superficie.
kb= Factor de tamaño.
kc= Factor de carga.
kd= Factor de temperatura.
ke= Factor de efectos diversos.

Factor de acabado de la superficie K_a.

Este factor toma en cuenta la calidad del acabado del elemento mecánico y la resistencia a la tensión. Este factor se determina mediante la expresión:

$$k_a = aS_{ut}^{b}$$ ---------- (23)

Los valores de *a* y *b* se obtienen de la tabla siguiente:

Acabado de superficie	Factor *a* Kpsi (MPa)	Exponente *b*
Esmerilado (rectificado)	1.34(1.58)	- 0.085
Maquinado o estirado en frío	2.70(4.51)	- 0.265
Laminado en caliente	14.4(57.7)	- 0.718
Forjado	39.9(272)	- 0.995

Tabla 2.- Factores de acabado de superficie.

Factor de tamaño K_b.

Este factor toma en cuenta la sección transversal del elemento. Para secciones circulares a flexión y torsión se tiene:

$$k_b = \begin{cases} 0.879d^{-0.107} & 0.11pul < d \le 2pul \\ 0.91d^{-0.157} & 2pul < d \le 10pul \\ 1.24d^{-0.107} & 2.79mm \le d \le 51mm \\ 1.51d^{-0.157} & 51mm < d \le 254mm \end{cases}$$ ---------- (24)

Para cargas axiales: kb = 1

Factor de carga K_c.

Este factor depende de la forma en que se aplica la carga. Los valores medios son:

$$k_c = \begin{cases} 1 & flexión \\ 0.85 & carga\ axial \\ 0.59 & torsión \end{cases}$$ ---------- (25)

Cuando existan cargas combinadas de flexión y torsión usar $K_c = 1$

Factor de temperatura K_d.

Este factor depende de la temperatura de operación de los elementos mecánicos, ya que cuando esta temperatura es menor que la del ambiente, la fractura por fragilidad es una posibilidad latente; y cuando la temperatura es mayor debemos investigar la fluencia del material debido a que esta disminuye con la temperatura.

Temperatura °C	k_d	Temperatura °F	k_d
20	1.000	70	1.000
50	1.010	100	1.008
100	1.020	200	1.020
150	1.025	300	1.024
200	1.020	400	1.018
250	1.000	500	0.995
300	0.975	600	0.963
350	0.943	700	0.927
400	0.900	800	0.872

Tabla 3.- Factores de temperatura

Si no se conoce la temperatura del lugar de trabajo use: $K_d = 1$

Factor de efectos diversos K_e.

Este factor toma en cuenta la reducción en el límite de resistencia a la fatiga debido a efectos tales como:

a).- Procesos de manufactura.

b).- Esfuerzos residuales.

c).- Recubrimientos.

d).- Corrosión.

Un enfoque utilizado con frecuencia consiste en emplear el factor de efectos diversos como un factor de reducción de la resistencia. Con este enfoque se define mediante la expresión:

$K_e = \frac{1}{K_f}$ ---------- (26)

De donde $K_f = 1 + q(K_t - 1)$ ---------- (27)

Que a su vez

$$q = \frac{1}{1+\frac{\sqrt{a}}{\sqrt{r}}}$$ ---------- (28)

De donde los valores de *a* se representan en la siguiente tabla para aceros:

S_{ut} (kpsi)	$\sqrt{a}$ (pul$^{0.5}$)
50	0.130
55	0.118
60	0.108
70	0.093
80	0.080
90	0.070
100	0.062
110	0.055
120	0.049
130	0.044
140	0.039
160	0.031
180	0.024
200	0.018
220	0.013
240	0.009

Tabla 1.- Valores de $\sqrt{a}$

FATIGA DE CICLO ALTO PARA DURACIÓN FINITA.

En muchas aplicaciones el número de ciclos de esfuerzo que se aplica sobre un componente durante su vida útil se sitúa entre **103** y **107**. Un ejemplo son las bisagras de las puertas de automóviles. Como la resistencia baja rápidamente en este rango, un enfoque que no toma en cuenta esta baja es inherentemente defectuoso.

La resistencia a la fatiga en cualquier localización entre S_f y S_e se puede expresar como sigue:

$S_f = aN^b$ ---------- (2.28)

S f= Resistencia a la fatiga para cualquier valor de ciclos ***N***.

N = número de ciclos de duración.

$$a = \frac{(0.9S_{ut})^2}{S_e}$$ ---------- (2.29)

$$b = -\frac{1}{3}\log\frac{0.9S_{ut}}{S_e}$$ ---------- (2.30)

Se = Límite de resistencia a la fatiga en el elemento mecánico a diseñar.

Para un esfuerzo completamente invertido σ*a,* el número de ciclos de duración correspondiente se determina a partir de la ecuación

$$N = \left(\frac{\sigma_V}{a}\right)^{1/b}$$ ---------- (2.31)

En la ecuación (3.12) se debe usar el esfuerzo equivalente de Von Vises:

Para esfuerzos combinados: - $$(\sigma_V)_e = \sqrt{\sigma_V^2 + 3\tau_V^2}$$ ---------- (2.32)

CRITERIO DE GOODMAN.

Este criterio propone la conexión del límite a la fatiga modificado (Se*)* sobre el eje de esfuerzo alternante con la resistencia última a la tensión (Sut) sobre el eje de esfuerzo medio, mediante una línea recta. Matemáticamente se tiene:

$$\frac{S_V}{S_e} + \frac{S_m}{S_{ut}} = 1$$ ---------- (2.33)

Introduciendo el factor de seguridad *ns* se obtiene:

$$\frac{\sigma_V}{S_e} + \frac{\sigma_m}{S_{ut}} = \frac{1}{n_s}$$ ---------- (2.34)

CÁLCULO DE LA PRIMERA VELOCIDAD CRÍTICA DE UN EJE ESCALONADO CON DOS APOYOS SIMPLES

En esta parte del proyecto analizaremos el diámetro recomendado para ejes con hasta 2 cambios de sección (3 diámetros) y hasta con 3 fuerzas que se aplicaran a este, a continuación se presenta un bosquejo del mismo.

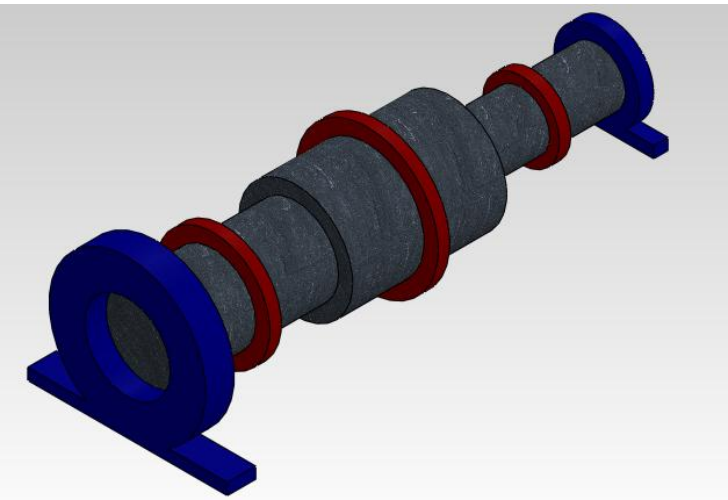


Figura 14. Eje escalonado con tres masas y apoyos siemples

En la siguiente figura presentamos el eje con los datos algebraicos:

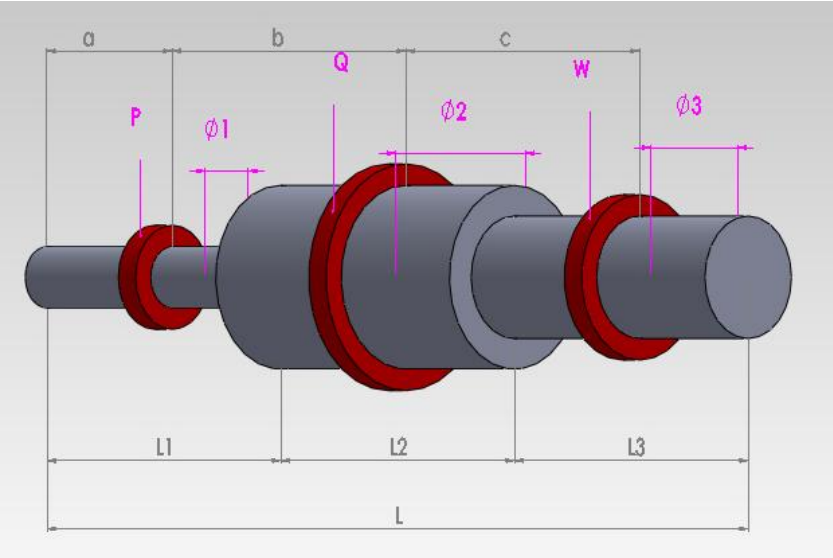


Figura 15. Eje con sus valores algebraicos.

Dónde:

P = Primera fuerza (N ó Lb)

Q = Segunda fuerza (N ó Lb)

W = Tercer fuerza (N ó Lb)

L_1 = Longuitud del primer tramo (m ó in)

L_2 = Longuitud del segundo tramo (m ó in)

L_3 = Longuitud del tercer tramo (m ó in)

L = Longuitud total del tramo (m ó in)

$\emptyset_1$ = Diámetro del primer tramo (m ó in)

$\emptyset_2$ = Diámetro del segundo tramo (m ó in)

$\emptyset_3$ = Diámetro del tercer tramo (m ó in)

d = Distancia del primer apoyo a la primera fuerza (m ó in)

e = Distancia de la primera fuerza a la gunda fuerza (m ó in)

f = Distancia de la segunda fuerza a la tercera fuerza (m ó in)

Ahora encontraremos las reacciones en los apoyos que se encuentran en el punto “A” y “E”:

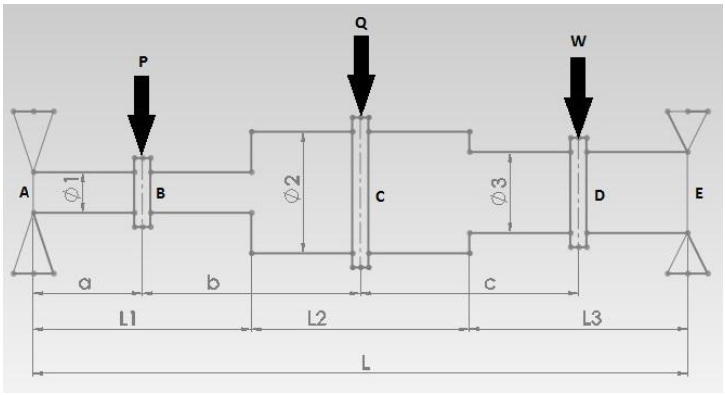


Figura 16. Cuerpo libre del eje.

$$\sum M_A = -P(a) - Q(a+b) - W(a+b+c) + R_E(L) = 0$$

$$R_E = P\left(\frac{a}{L}\right) + Q\left(\frac{a+b}{L}\right) + W\left(\frac{a+b+c}{L}\right) \dots\dots\dots\dots Ec(1)$$

$$\sum M_E = W[L-(a+b+c)] + Q[L-(a+b)] + P(L-a) - R_A(L) = 0$$

$$R_A(L) = (W+Q+P)L - W(a+b+c) - Q(a+b) - P(a)$$

$$R_A = W + Q + P - W\left(\frac{a+b+c}{L}\right) - Q\left(\frac{a+b}{L}\right) - P\left(\frac{a}{L}\right) \dots\dots\dots\dots Ec.\,(2)$$

El siguiente paso consiste en realizar cortes en cada sección para poder encontrar los momentos.

- **Primer corte**

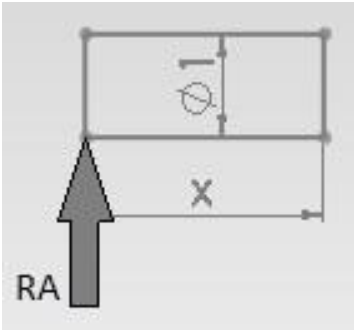


Figura 17. Primer corte

$$M_1 = R_A(L1) - (P(L1-d))$$

- **Segundo corte**

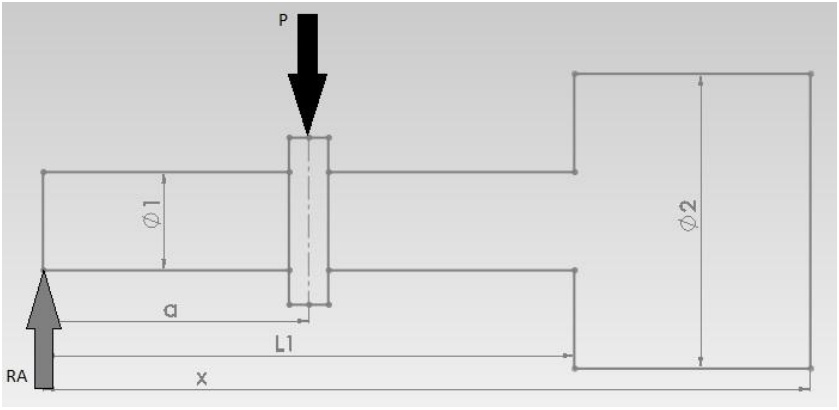


Figura 18. Segundo corte

$M_2 =$ `RA*(L1+L2)-P*((L1+L2)-d)-Q*((L1+L2)-(d+e));`

- **Tercer corte**

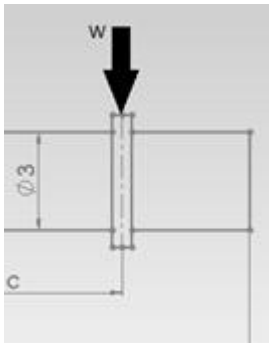


Figura19. Tercer corte

$M_3 =$ `RE*L3-W*((d+e+f)-(L1+L2));`

CÁLCULO DEL DIAMETRO CONSTANTE DE UN EJE

En esta parte del proyecto encontraremos el diámetro de un eje que tendrá hasta tres pesos(fuerzas), el diámetro va a ser constante a través de todo el eje, para poder encontrar dicho diámetro usaremos la teoría de energía de distorsión, a continuación presentamos el eje a analizar:

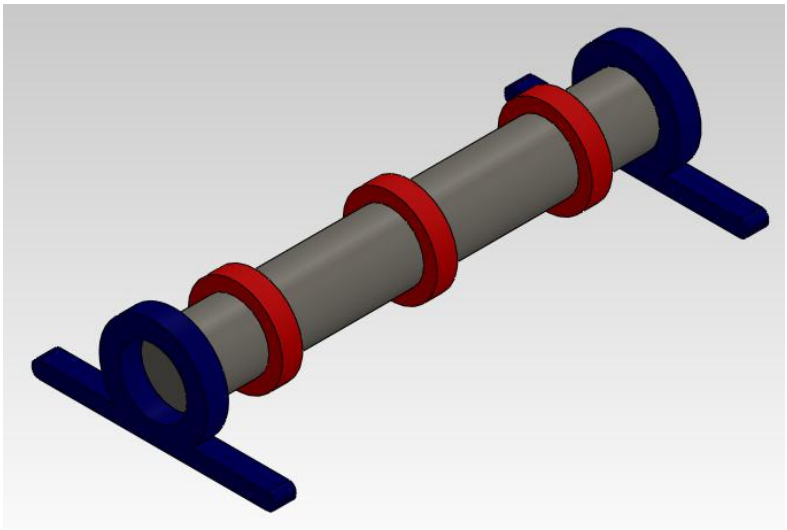


Figura 20. Eje a analizar.

En la siguiente figura presentamos el eje con los datos algebraicos:

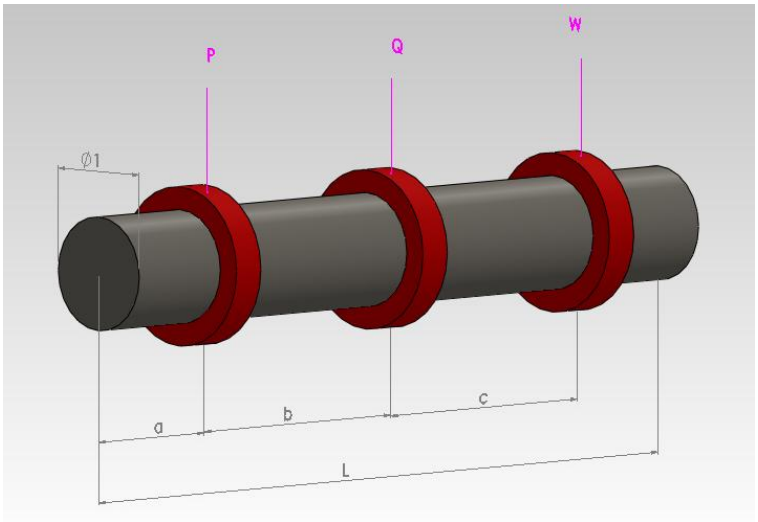


Figura 21. Eje con valores algebraicos

Donde:

P = Primera fuerza (N ó Lb)

Q = Segunda fuerza (N ó Lb)

W = Tercer fuerza (N ó Lb)

d = Distancia del primer apoyo a la primera fuerza (m ó in)

e = Distancia de la primera fuerza a la segunda fuerza (m ó in)

f = Distancia de la segunda fuerza a la tercera fuerza (m ó in)

L = Longuitud total del eje (m ó in)

ϕ_1 = Diámetro del eje (m ó in)

Ahora encontraremos las reacciones en los apoyos que se encuentran en los puntos “A” y “E”.

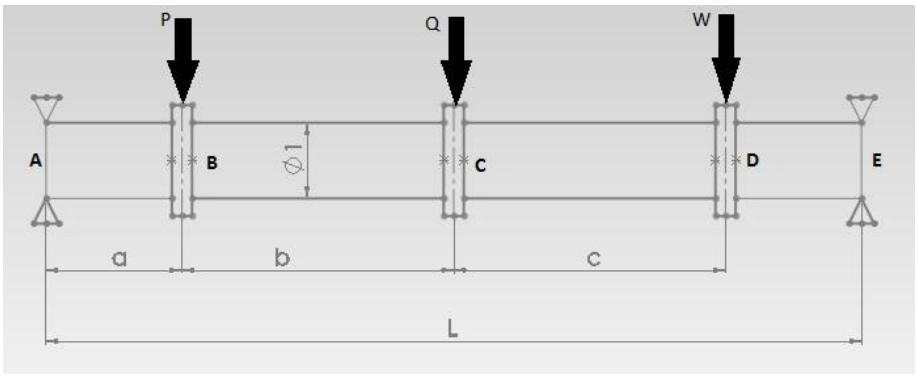


Figura 22. Diagrama de cuerpo libre del eje.

$$\sum M_A = -P(a) - Q(a+b) - W(a+b+c) + R_E L = 0$$

$$\therefore R_E = P\left(\frac{a}{L}\right) + Q\left(\frac{a+b}{L}\right) + W\left(\frac{a+b+c}{L}\right) \ldots\ldots \text{Ec. (26)}$$

$$\sum M_E = W[L-(a+b+c)] + Q[L-(a+b)] + P[L-a] - R_A L = 0$$

$$R_A L = WL - W(a+b+c) + QL - Q(a+b) + PL - P(a)$$

$$R_A L = (W+Q+P)L - W(a+b+c) - Q(a+b) - P(a)$$

$$R_A = W + Q + P - W\left(\frac{a+b+c}{L}\right) - Q\left(\frac{a+b}{L}\right) - P\left(\frac{a}{L}\right) \ldots\ldots\ldots \text{Ec. (27)}$$

El siguiente paso consiste en realizar cortes en cada sección para poder encontrar los momentos.

- **Primer corte**

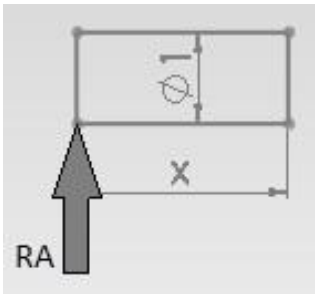


Figura 23. Primer corte

$$M_1 = R_A(L1) - (P(L1-d))$$

- **Segundo corte**

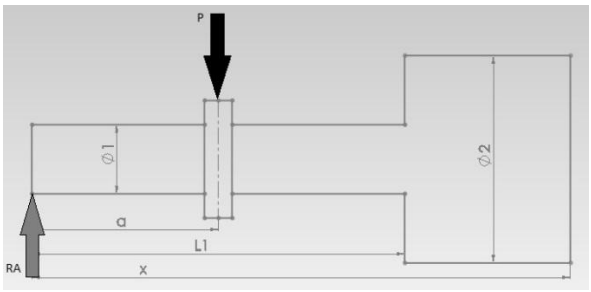


Figura 24. Segundo corte

$M_2 =$ `RA*(L1+L2)-P*((L1+L2)-d)-Q*((L1+L2)-(d+e));`

- **Tercer corte**

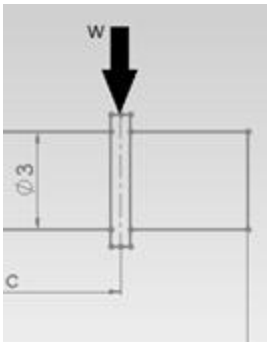


Figura 25. Tercer corte

$M_3 =$ `RE*L3-W*((d+e+f)-(L1+L2));`

Ecuación para el cálculo del diámetro del eje

Una vez que hemos encontrado las deflexiones en los puntos “B”, “C” y “D” ocasionadas por las fuerzas, sustituimos las deflexiones en la ecuación de Rayleigh-Ritz:

$$\omega_c = \sqrt{\frac{g(P\delta_B + Q\delta_C + W\delta_D)}{P\delta_B^2 + Q\delta_C^2 + W\delta_D^2}} \ldots\ldots\ldots \text{Ec. (42)}$$

MATLAB

Es un lenguaje de alto nivel orientado al desarrollo de cálculos técnicos. Integra cálculo, visualización y programación en un entorno interactivo de fácil manejo donde los problemas y las soluciones se expresan en la notación matemática habitual.

El elemento de información básico en MATLAB es una tabla a la que no hace falta asignar dimensión con antelación. Esto permite abordar problemas que requieren una formulación vectorial o matricial en mucho menos tiempo de lo que se tardaría con un lenguaje escalar no interactivo.

Como entrar y salir de MatLab

Para entrar en MATLAB basta pulsar dos veces seguidas el botón izquierdo del ratón sobre el icono correspondiente. Aparece entonces la ventana de comandos y, en su interior, el indicador (prompt) >>.

MATLAB está listo para recibir nuestras instrucciones. Para salir del programa basta teclear quit y pulsar la tecla Enter (toda instrucción en MATLAB debe concluir pulsando esta tecla por lo que, de ahora en adelante, omitiremos dicha acción). Otra opción es desplegar el menú File y desplazar la barra hasta la opción Exit MATLAB

Para obtener ayuda

El comando (help) constituye la forma más básica de conocer la sintaxis y el comportamiento de una función en particular. Su formato es:

(helpnombre_de_funcion)

Y la información solicitada aparece directamente en la ventada de comandos. En dicha información MATLAB emplea letras mayúsculas para referirse a la función y a los nombres de variables, con objeto de destacarlos del resto del texto. Sin embargo, a la hora de utilizar la función en cuestión siempre escribiremos su nombre con letras minúsculas. MATLAB distingue entre ambos tipos de letras (case sensitive) y todos los nombres de función son, de hecho, en minúscula.

Si no queremos que la información aparezca en la ventana de comandos podemos recurrir a una ventana auxiliar, la ventana de ayuda. Ésta se invoca mediante la orden helpwin o seleccionando la opción HelpWindow del menú Help. Se puede acceder directamente a una función concreta mediante la instrucción helpwinnombre_de_funcion. Una ventaja de la ventana de ayuda es que provee enlaces con otras cuestiones relacionadas con la información que hayamos solicitado.

Operadores

Suma	+	Menor que	<
Potencia	^	Mayor que	>
Resta	-	Menor o igual que	<=
Multiplicación	*	Mayor o igual que	>=
División	/	Diferente	~=
División izquierda	\	And (y)	&
Traspuesta	C=A’	Or (ó)	|
Igual que	==	Not (no)	~

Tabla 3.3. Símbolos de los operadores

Variables en MatLab

- Todas las variables están almacenadas en 32 bit
- No distingue entre variables reales y enteras
- Distingue mayúsculas, en *Matlab*A≠ a
- Los nombres tienen que empezar por una letra no obstantese puede usar además de letras nombres y símbolos
- Todo el cálculo se hace en doble precisión format puede serusado para cambiarlo.
- Un “;” al final de la línea no muestra el resultado de laoperación
- Permite la definición de variables globales y locales
- Lista de variables who, Borrarlas clear, más informaciónWhos

ACERCA DE GUIDE

- Las interfaces de usuario de (GUI-GraphicalUser Interface en inglés), es la forma en que el usuario interactúa con el programa o el sistema operativo de una computadora. Una GUI contiene diferentes elementos gráficos tales como; botones, campos de texto, menús, gráficos, etc.
- Existen diferentes lenguajes de programación que permiten crear una GUI tales como: C, Visual Basic, TK, etc., solo por mencionar algunos. Todos ellos permiten usar diferentes controles y maneras de programarlos. MatLab nos permite realizar GUIs de una manera muy sencilla usando GUIDE (GRaphicalUser Interface DevelopmentEnviroment).

Archivos .m

Los archivos M son archivos de texto (archivos ASCII) que contienen instrucciones propias de MATLAB. Se llaman así porque su nombre tiene (obligatoriamente) la extensión m.

Como archivos de texto, los archivos M se pueden crear con cualquier procesador. Recomendamos, no obstante, utilizar el editor que trae incorporado MATLAB. Para ello, cuando queramos crear un nuevo archivo M, debemos seguir el siguiente esquema:

Creación de un nuevo archivo .m	
Llamada al editor	Si todavía no estamos en nuestro directorio de trabajo, nos pasamos a él.
	Tecleamos edit (aparece una ventana llamada MATLAB Editor/Debugger y en su interior otra de nombre Untitled1).
Creación del archivo	Escribimos el texto del archivo.
Guardamos lo	En el menú File elegimos Save As (aparece una ventana llamada Guardar Como).

escrito	Si queremos guardar el archivo en un USB, ascendemos de nivel hasta llegar a Mi Pc. Una vez ahí pinchamos en USB (A:).
	En el campo Nombre de archivo escribimos el nombre que queramos ponerle, sin la extensión.
	Pulsamos el botón Guardar.
Salimos del editor	En el menú File elegimos Exit Editor/Debugger

Tabla 3.4. Como crear un archivo .m

Para editar un archivo ya existente, con objeto de revisarlo y/o modificarlo, seguimos el mismo esquema pero con las siguientes variantes:

- Invocamos el editor con edit [a:\]nombre_del_archivo.
- Si cambiamos el texto y queremos guardar las modificaciones en el mismo archivo, elegimos (Save) en el menú File. Si queremos guardarlas en otro archivo, elegimos (Save As) en el menú File y seguimos los mismos pasos del esquema anterior.

En cualquier momento podemos conocer los archivos M existentes en nuestro directorio de trabajo tecleando what. También podemos revisar el contenido de un archivo M sin necesidad de llamar al editor. Basta teclear type [a:\]nombre_del_archivo. Hay dos tipos de archivos M: de guión y de función. Los archivos M de guión son los más simples:

- No tienen argumentos de entrada ni de salida.
- Son útiles para automatizar bloques de instrucciones y cálculos que deben efectuarse repetidamente.
- Pueden operar sobre datos ya existentes en el espacio de trabajo o sobre datos que ellos mismos introduzcan.
- Cualquier variable creada por uno de estos archivos permanece en el espacio de trabajo una vez que finaliza su lectura.

En cambio, los archivos M de función:

- Aceptan argumentos de entrada y salida.
- Sirven para extender el lenguaje de MATLAB creando nuestras propias funciones.
- Tienen su propio espacio de trabajo reservado, donde pueden definirse variables propias (locales) que no afectan al espacio de trabajo general.

La estructura del texto de un archivo M de guión es totalmente libre (siempre que se ajuste, por supuesto, a las instrucciones de MATLAB). Además, podemos y debemos incluir comentarios que aclaren el contenido del archivo. Tales comentarios han de ir precedidos por el símbolo de tanto por ciento: %.

Por ejemplo, para crear un archivo M de guión que reproduzca el ejemplo de construcción por bloques, escribiremos lo siguiente:

```
% Ejemplo de construcción de una matriz por bloques.

% Apuntes sobre MATLAB, pagina 9.

B1 = [1 -3 5; 2 3 4; -7 8 9]

B2 = 5*eye(3)

% Como B1 y B2 van a estar a un mismo nivel,

% deben tener el mismo numero de filas.

B3 = zeros(2,1)

B4 = ones(2,3)

B5 = fliplr(2*eye(2))

% Como B3, B4 y B5 van a estar a un mismo nivel,

% deben tener el mismo número de filas.

A = [B1 B2; B3 B4 B5]

% Los dos niveles deben tener el mismo número de columnas.
```

Si hemos guardado el archivo con el nombre bloques.m, bastará escribir bloques en la línea de comandos para que se ejecuten de una vez todas las instrucciones.

La estructura de un archivo M de función es un poco más complicada. Sus elementos básicos son una línea de definición, una línea H1, texto de ayuda, el cuerpo de la función y comentarios adicionales. Por ejemplo, para crear una función que calcule la superficie de un círculo, éste podría ser el contenido del archivo:

```
function s = supcirc(r)

% SUPCIRC Superficie de un círculo

% SUPCIRC(R), donde R es una matriz, calcula la superficie de los círculos

% cuyos radios son los coeficientes de la matriz.

s = pi * r.^2;
```

La primera línea de texto es la línea de definición. En ella hay que destacar cuatro cosas:

- La palabra clave function. Indica a MATLAB que el archivo M contiene una función.
- El argumento de salida s. En este caso sólo hay uno, pero podría haber más. Cuando esto ocurre, deben encerrarse todos entre corchetes y separarse por comas. También puede ocurrir que no haya argumentos de salida y en tal caso utilizaremos dos corchetes vacíos: [ ].
- El nombre de la función supcirc. Las reglas para los nombres de las funciones son las mismas que para las variables. Importante: el nombre del archivo M debe coincidir con el de la función.
- El argumento de entrada r. Va encerrado entre paréntesis. Puede haber más de uno, en cuyo caso irán separados por comas.

Éstos son otros ejemplos válidos de líneas de definición:

```
function [x,y,z] = esfera(theta, phi, rho)

function [ ] = sinsalida(x)
```

Control de flujo

MATLAB tiene cinco construcciones de control de flujo:

- enunciados if,
- enunciados switch,
- bucles while,
- bucles for y
- enunciados break.

El enunciado if evalúa una expresión lógica y ejecuta un grupo de instrucciones según el valor de la misma. En su forma más simple, su estructura es

if*expresion_logicainstrucciones*

end

Si la expresión es verdadera, MATLAB ejecuta todas las órdenes entre las líneas if y end. Si la expresión es falsa, MATLAB no ejecuta dichas instrucciones. Los comandos opcionales elseif y else permiten la ejecución alternativa de varios grupos de instrucciones. Por ejemplo:

if*expresion_1*

instrucciones_1

elseif*expresion_2*

instrucciones_2

else

instrucciones_3

end

Hace lo siguiente:

1. Evalúa *expresion_1*.

2. Si es verdadera, ejecuta *instrucciones_1* y termina. Si es falsa, evalúa *expresion_2*.

3. Si *expresion_2* es verdadera, ejecuta *instrucciones_2* y termina. Si es falsa, ejecuta *instrucciones_3*.

En toda expresión lógica aparecen los llamados operadores relacionales: < (menor que), <= (menor o igual que), > (mayor que), >= (mayor o igual que), == (igual que) y ~= (distinto de). Estos operadores permiten comparar dos escalares, dos matrices de la misma dimensión (elemento a elemento) o un escalar con una matriz (cada elemento de la matriz se compara con el escalar). En el primer caso, la expresión vale 1 si es verdadera y 0 si es falsa. En los otros dos, la expresión da un matriz de ceros y unos de igual dimensión que la o las matrices implicadas. Se entenderá que la expresión es verdadera cuando todos los coeficientes sean unos. Dos expresiones lógicas pueden a su vez compararse entre si mediante los operadores lógicos & y |. El enunciado expresion_1 & expresion_2 es verdadero si y sólo si ambas expresiones lo son, mientras que el enunciado expresion_1 | expresion_2 es verdadero si y sólo si al menos una de las dos expresiones lo es. El operador ~ sirve para negar una expresión. Así, ~expresión es verdadera si y sólo si expresión es falsa y viceversa.

El enunciado switch ejecuta ciertas instrucciones en función del valor de una variable o expresión. Su estructura es la siguiente:

switch*expresion*

    case*valor_1*

        instrucciones_1

    case*valor_2*

        instrucciones_2

    otherwise

        otras_instrucciones

end

Como vemos el enunciado consta de:

1. La palabra switch seguida de una expresión a evaluar.

2. Varios grupos case. Cada grupo consiste en una primera línea con la palabra clave case seguida de un posible valor para *expresion*. Las líneas siguientes del grupo contienen las instrucciones a ejecutar en caso de que *expresion*tome ese valor. Sólo se ejecuta el primer grupo case cuyo valor coincide con el de la expresión.

3. Un grupo otherwise. Consiste en una primera línea con dicha palabra clave seguida de otras líneas con las órdenes a ejecutar en caso de que el valor de *expresion*no coincida con ninguno de los establecidos en los grupos case.

4. Una línea con la palabra clave end. El bucle while ejecuta una instrucción o un grupo de instrucciones mientras que cierta expresión de control sea verdadera. Su estructura es la siguiente:

while*expresion*

Instrucciones

end

Gráficos

MATLAB dispone de un gran número de herramientas que permiten visualizar vectores y matrices como gráficos. Aquí tan sólo describimos, brevemente, algunos de ellos.

Para crear un gráfico bidimensional utilizamos el comando plot, el cual admite distintos formatos. Si y es un vector, plot(y) produce un gráfico lineal a trozos de los elementos de y frente a los índices de dichos elementos. Si x e y son dos vectores de la misma longitud, plot(x,y) produce un gráfico lineal a trozos de y frente a x. Por ejemplo, para dibujar la gráfica de la función seno en el intervalo [0,2pi] escribiremos

```
t = 0: pi/100:2*pi;

y = sin(t);

plot(t,y)
```

Múltiples parejas x-y permiten crear varias gráficas, cada una de un color distinto, con una sola llamada a plot. Probemos con

y2 = sin(t-0.25);

y3 = sin(t-0.5);

plot(t,y,t,y2,t,y3)

Es posible especificar el color, cómo se unen los puntos (estilo de línea) y cómo se marcan los puntos (marcadores) con

plot(x,y,'color_estilo_marcador')

Dónde: color_estilo_marcador es una cadena de uno, dos o tres caracteres construida en base a las siguientes posibilidades:

- Los colores disponibles son c, m, y, r, g, b, w y k, que corresponden a turquesa, magenta, amarillo, rojo, verde, azul, blanco y negro.
- Los estilos disponibles son -, --, :, -. y none, que corresponden a continuo, con guiones, punteado, con guiones y puntos y sin línea.
- Los marcadores más habituales son +, o, * y x.

La instrucción

plot(x,y,'y:+')

Dibuja la gráfica en color amarillo, punteada y con el marcador + en cada uno de los puntos correspondientes a los datos. Si especificamos un marcador pero no un estilo de línea, MATLAB sólo dibuja los marcadores.

La función plot abre automáticamente una ventana gráfica (figure window) si no hay ninguna abierta en la pantalla. Si ya existe una ventana gráfica, plot la utiliza por defecto, borrando todo lo que en ella hubiera dibujado. Para abrir una nueva ventana gráfica y que pase a ser la ventana en uso, escribimos figure. Para que una ventana gráfica ya existente pase a ser la ventana en uso escribimos figure(n), donde n es el número que aparece en la barra superior de la ventana. Los resultados de los comandos gráficos que ejecutemos a continuación se mostrarán en dicha ventana.

El comando hold permite añadir elementos diversos a un gráfico ya existente. Cuando tecleamos holdon MATLAB protege el gráfico de la ventana actual y superpone al mismo los efectos de los nuevos comandos gráficos que ejecutemos, reescalando si es necesario. Por ejemplo, escribamos

```
holdon

y4 = cos(t);

plot(t,y4,'b:*')
```

Y veremos cómo la gráfica del coseno se dibuja sobre el último gráfico creado. La orden se desactiva con hold off.

La función subplot permite dividir una ventana gráfica en varias ventanas más pequeñas y dibujar en cada una de ella gráficos distintos. Al escribir

```
subplot(m,n,p)
```

La ventana gráfica actual se divide en m por n subventanas, numeradas de izquierda a derecha y de arriba abajo, y selecciona la ventana número p como ventana actual.

La función axis permite modificar el aspecto de la caja donde se va a dibujar el gráfico. Por defecto, MATLAB encuentra el máximo y el mínimo entre los datos que le damos y elige unas dimensiones y un etiquetado de los ejes adecuado. Pero podemos establecer los límites que queramos para los ejes escribiendo axis ([xminxmaxyminymax]). Con axis square conseguimos que las dimensiones de los ejes sean las mismas. Con axis equal logramos que incrementos iguales en ambos ejes midan lo mismo. La orden axis auto nos devuelve al escalado por defecto. Si escribimos gridon el gráfico aparece en una cuadrícula. Con grid off la cuadrícula desaparece. Las funciones xlabel, ylabel y zlabel(ésta última para gráficos tridimensionales) permiten poner nombres a los ejes coordenados. Si queremos, por ejemplo, llamar al eje x por su nombre escribiremos

```
xlabel('Eje x')
```

La orden title permite poner nombre al gráfico representado. Su formato es

```
title('titulo_del_grafico')
```

MATLAB también permite representar superficies de la forma z=f(x,y). Para ello disponemos de dos comandos, mesh y surf. El funcionamiento es análogo en ambos: sobre una malla de una determinada parte del plano levantamos una serie de puntos y los unimos. La diferencia entre uno y otro comando está en la forma en que dicha unión se lleva a cabo. Mientras que mesh sólo colorea las líneas de unión, surf también da color a las porciones planas que aquellas delimitan. Supongamos que queremos representar la superficie z = xe-x2-y2 sobre el cuadrado [-2,2]x[-2,2]. Éstos son los pasos a seguir:

1. Construcción de la malla. Comenzamos creando una partición uniforme sobre el intervalo [-2,2] del eje x:

```
px = -2:0.2:2;
```

A continuación creamos otra sobre el intervalo [-2,2] del eje y:

```
py = -2:0.1:2;
```

Con estas particiones la malla tendría 21x41 puntos. Por último creamos dos matrices x e y con las siguientes características: las 21 filas de x son iguales y consisten en las abscisas de los puntos de la malla; las 41 columnas de y son iguales y consisten en las ordenadas de los puntos de la malla. Para ello escribimos [x,y] = meshgrid(px,py). Si px y py fueran iguales, bastaría escribir [x,y] = meshgrid(px).

2. Levantamiento de la función. Escribimos

```
z = x.*exp(-x.^2-y.^2);
```

3. Construcción de la gráfica. Escribimos mesh(x,y,z) o surf(x,y,z).

Antes de finalizar este apartado queremos insistir en que aquí está recogida tan sólo una ínfima parte de las posibilidades gráficas de MATLAB. Nos dejamos atrás aspectos como la representación de curvas en el espacio, la coloración, textura, iluminación, enfoque y proyección de superficies, la colocación de cámaras alrededor de una superficie que nos permitan movernos entorno a ella, la creación de gráficos de barras, áreas y sectores, la representación de campos vectoriales, la creación de contornos de superficies (curvas de nivel), el tratamiento de imágenes digitales, la modelización 3D, etc.

Otros aspectos interesantes

Finalizamos esta introducción a MATLAB con algunas cuestiones que puedan sernos de utilidad. Empezamos con el control del tiempo. La orden clock muestra un vector de 6 componentes en notación decimal. Las tres primeras corresponden a la fecha actual y las tres últimas a la hora en curso. Si lo queremos en notación entera escribiremos fix(clock). Con esta orden podemos controlar el tiempo que tarda un programa en ejecutarse. Bastará hacer, por ejemplo,

```
tp1 = fix(clock(3:6));
```

Antes de la ejecución,

```
tp2 = fix(clock(3:6));
```

Después de ella y luego comparar tp1 con tp2.

Otro aspecto al que queremos hacer referencia es la evaluación de cadenas de caracteres, pues añade potencia y flexibilidad al lenguaje MATLAB. La función eval evalúa una cadena que contiene una expresión de MATLAB o un enunciado. En su forma más simple, su sintaxis es eval('cadena'). Por ejemplo, las siguientes instrucciones generan una matriz de orden n:

```
t = '1/(i+j-1)';

fori = 1:n

for j = 1:n

a(i,j) = eval(t);

end

end
```

La función feval ejecuta una función cuyo nombre aparece en una cadena. Su sintaxis es feval('cadena',dato_numerico) y su efecto es evaluar la función de nombre cadena en dato_numerico. El siguiente ejemplo permite elegir entre tres funciones distintas a evaluar:

```
fun = ['sin','cos','log'];
```

```
k = input('Elige un numero de funcion:');

x = input('Elige un punto para evaluar la funcion:');

feval(fun(k,:),x)
```

El argumento de la función eval puede ser una concatenación de cadenas de caracteres.

El siguiente ejemplo muestra cómo crear 10 variables de nombres P1, P2,..., P10 y que cada una tenga un valor distinto:

```
for k = 1:10

eval(['P',int2str(k),'=k^2'])

end
```

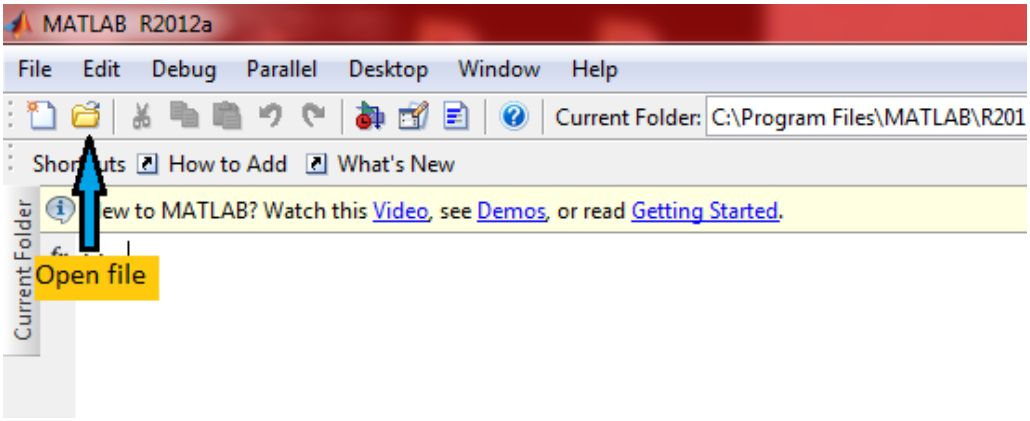
La orden int2str(k) transforma el valor numérico de k en una cadena de caracteres.

FUNCIONAMIENTO DE LA PROGRAMACION DEL CÁLCULO DEL DIAMETRO RECOMENDADO PARA UN EJE ESCALONADO PARA CARGAS VARIABLES Y UN EJE CON DIAMETRO CONSTANTE PARA CARGAS ESTÁTICAS.

La programación de “Cálculo del diámetro recomendado para un eje con diámetro constante y tres fuerzas para cargas estáticas se realiza de la siguiente manera:

Fig. 26 Carpeta de búsqueda de programa

1. Para poder ejecutar el archivo de MATLAB nombrado Estático se selecciona en la barra de herramientas la carpeta “abrir archivo” (open file) como se indica en la fig. 26



2. La carpeta “abrir archivo” nos abrirá una ventana nombrada Abrir (Open), en la cual buscaremos la ubicación del archivo (ya sea que se encuentre almacenada en la memoria de la PC o en una memoria externa) para seleccionarlo y abrirlo. Tal como se muestra en la fig. 27, existen dos archivos con el nombre Estático, como podemos apreciar uno de ellos posee la extensión FIG y la otra poseerá la extensión M. Nosotros seleccionaremos y abriremos el segundo, es decir, el archivo con extensión M.

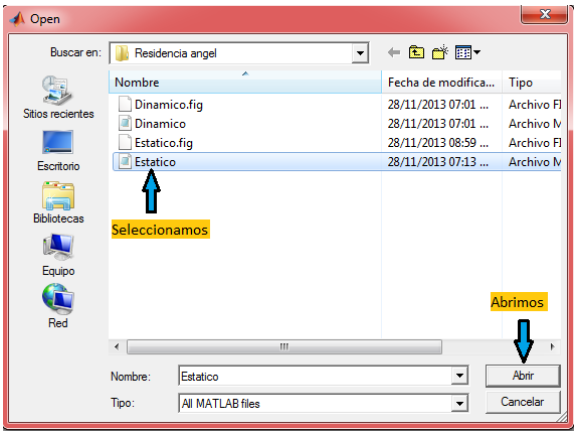


Fig. 17 Ventana Open

3. Una vez que damos abrir en la ventana “abrir”, la siguiente ventana que nos aparecerá de manera inmediata será la ventana de programación (Editor) como se muestra en la fig. 28, la cual no deberá sufrir modificaciones para que el programa siga funcionando adecuadamente. En esta ventana encontraremos en la barra de herramientas el botón “correr o ejecutar” (Run) el cual seleccionaremos para correr o ejecutar el programa.
   En el caso de que de manera accidental se altere o modifique dicha programación, bastará con cerrar la ventana antes de correrla y no guardar las modificaciones.

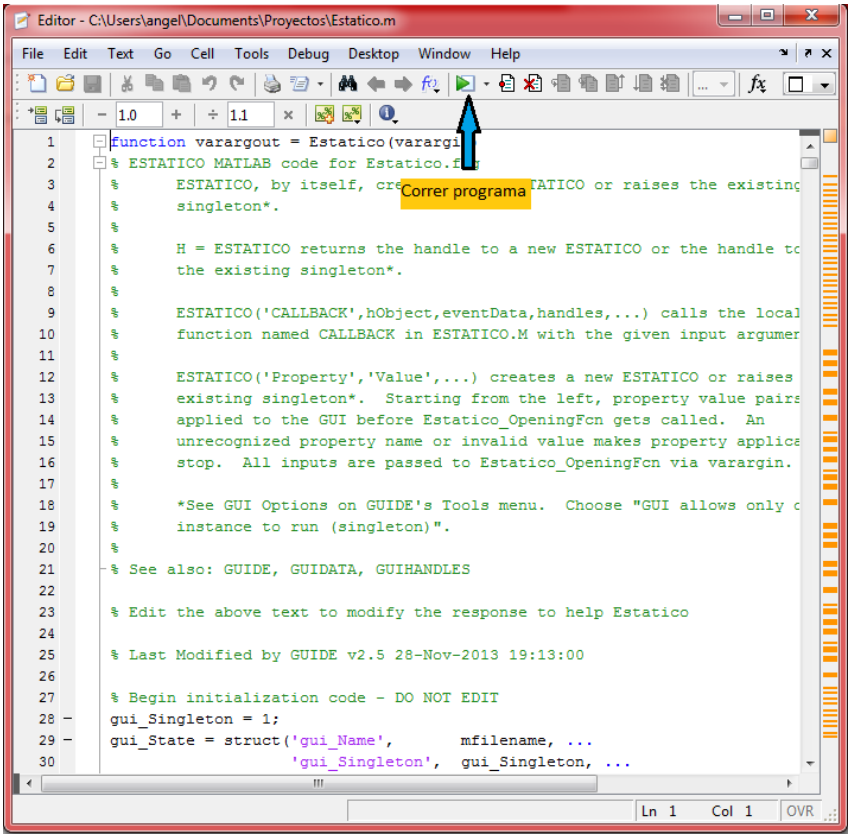


Fig. 28 Ventana de programación

4. Cuando el programa es corrido nos lleva a la ventana con el nombre de dicho programa, en este caso Estático y con tema “Módulo Para el Diseño de Ejes”. En esta comenzaremos a ingresar los datos que nos otorga el problema y los que dicho programa nos solicite. Los recuadros en blanco en los que se citan la frase “ingrese valor” son los lugares en los que los valores solicitados deberán ser ingresados, prestando atención de colocarlos adecuadamente en el espacio correspondiente.
   En la fig. 29 podemos apreciar esto de una manera más adecuada.

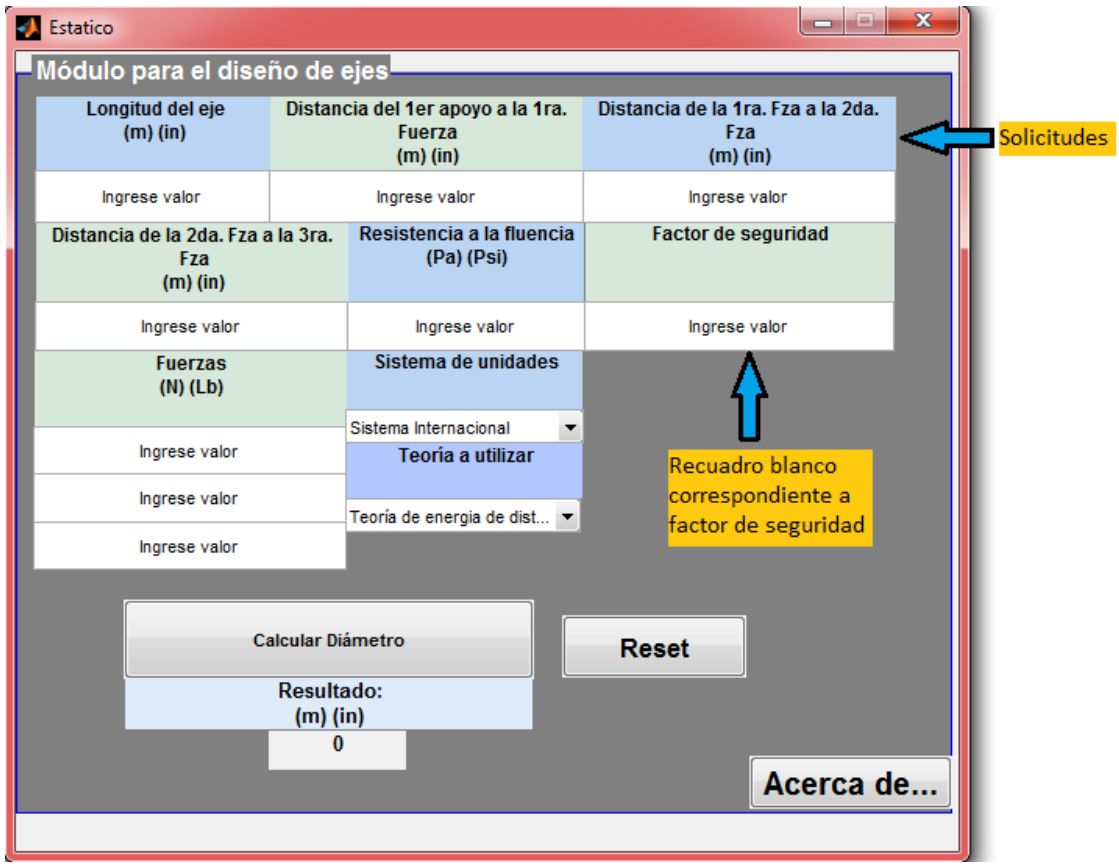


Fig. 29 Módulo para el diseño de ejes

Como se mencionó anteriormente este Módulo trabaja con 1,2 o un máximo de 3 fuerzas, es por ello que en la solicitud de fuerzas solo existen tres recuadros en línea vertical. También tiene la opción “Sistema de unidades” para elegir en que sistema se desea que realice los cálculos ya sea en SI (Sistema Internacional) o en Sistema Inglés. Mientras que la opción “Teoría a utilizar” nos despliega un menú de dos teorías; La Teoría de Energía de Distorsión y la Teoría de la Cortante Máxima. (Ver fig. 29)

5. Cabe mencionar que cuando el cálculo es para una o dos fuerzas los recuadros sobrantes en la petición de fuerzas deberán ser llenados con el valor de cero, de la misma manera que en las distancias entre fuerza y fuerza. Ver fig. 30.

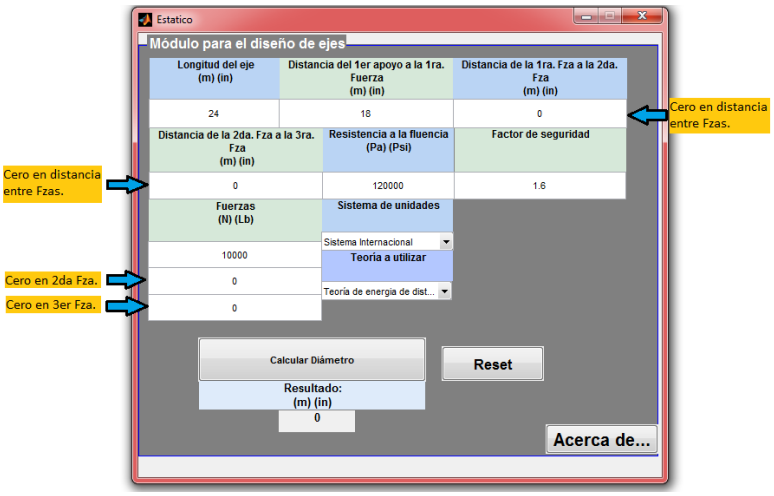


Fig. 30 Módulo de diseño con una fuerza

6. Si al momento de ingresar los datos, se comete una incoherencia, por ejemplo que se ingresen dos distancias entre fuerzas, y en la petición de estas fuerzas solo se ingrese el valor de una, el programa arrojara un mensaje de error como se muestra en la fig. 31.
   Como se mencionó los casos en los que el programa enviará el mensaje "Error en los datos" será cuando exista una incoherencia en la entrada de datos y lo que deberá realizarse en esos casos será simplemente ingresar los valores de manera adecuada.

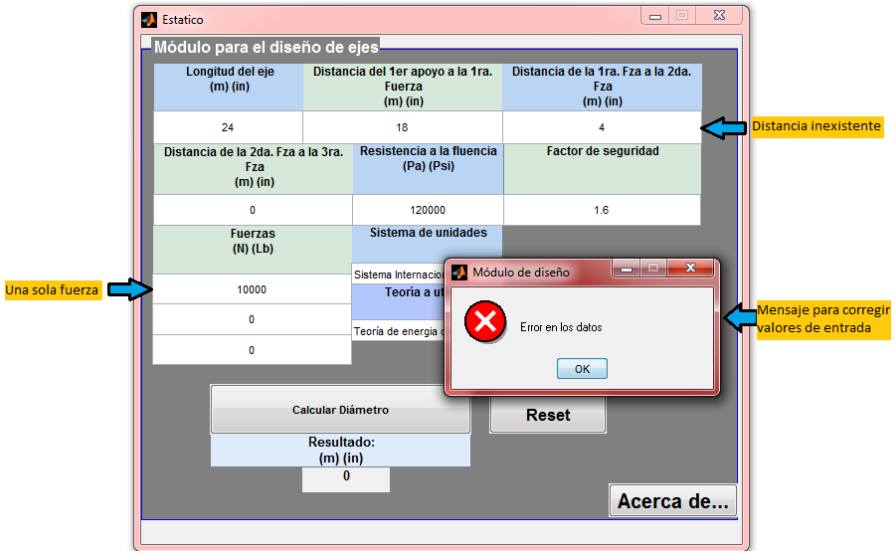


Fig. 31 Mensaje de Error

7. Una vez que los valores han sido ingresados de manera adecuada en el programa, habrá que dar al botón “Cálculo de diámetro” y esperar alrededor de 10 segundos para que el programa arroje el resultado, este botón también abrirá una ventana nueva en la cual se mostrara el eje con el diámetro obtenido y con las medidas de los tramos dados.

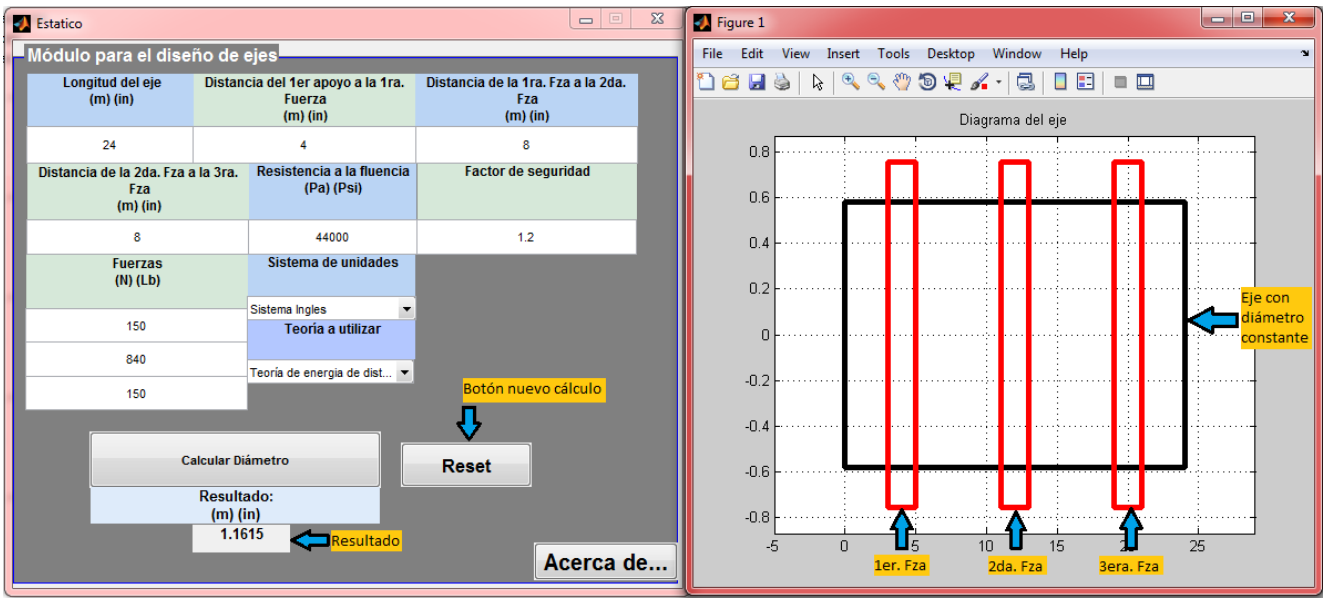


Fig. 32 Resultado con diagrama del eje

CÁLCULO DEL DIÁMETRO RECOMENDADO PARA UN EJE CON DIÁMETROS VARIABLES Y TRES CARGAS PARA CARGAS VARIABLES.

EJECUCIÓN DEL PROGRAMA

1. Para poder ejecutar el programa de MATLAB nombrado Dinámico se selecciona en la barra de herramientas la carpeta “abrir archivo” (Open file) como se indica en la Fig. 33.

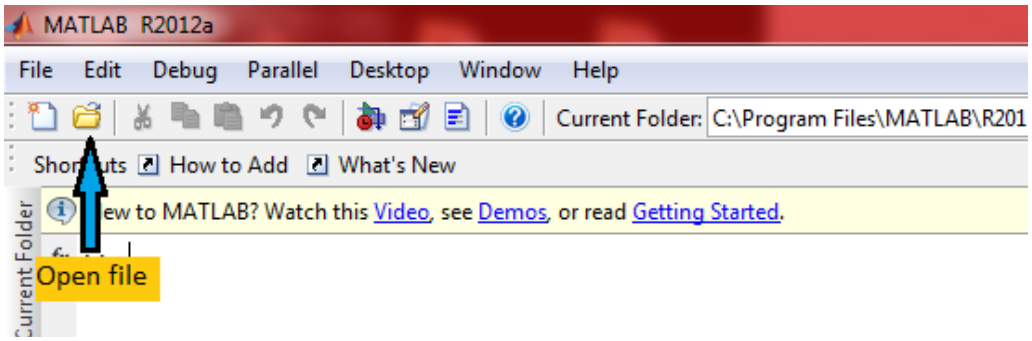


Fig. 33 Carpeta de búsqueda de programas

2. La carpeta “abrir archivo” nos abrirá una ventana nombrada simplemente “abrir” (Open) en la cual buscaremos la ubicación del archivo (ya sea que este almacenada en la memoria de la PC o en una memoria externa) para seleccionarlo y abrirlo.

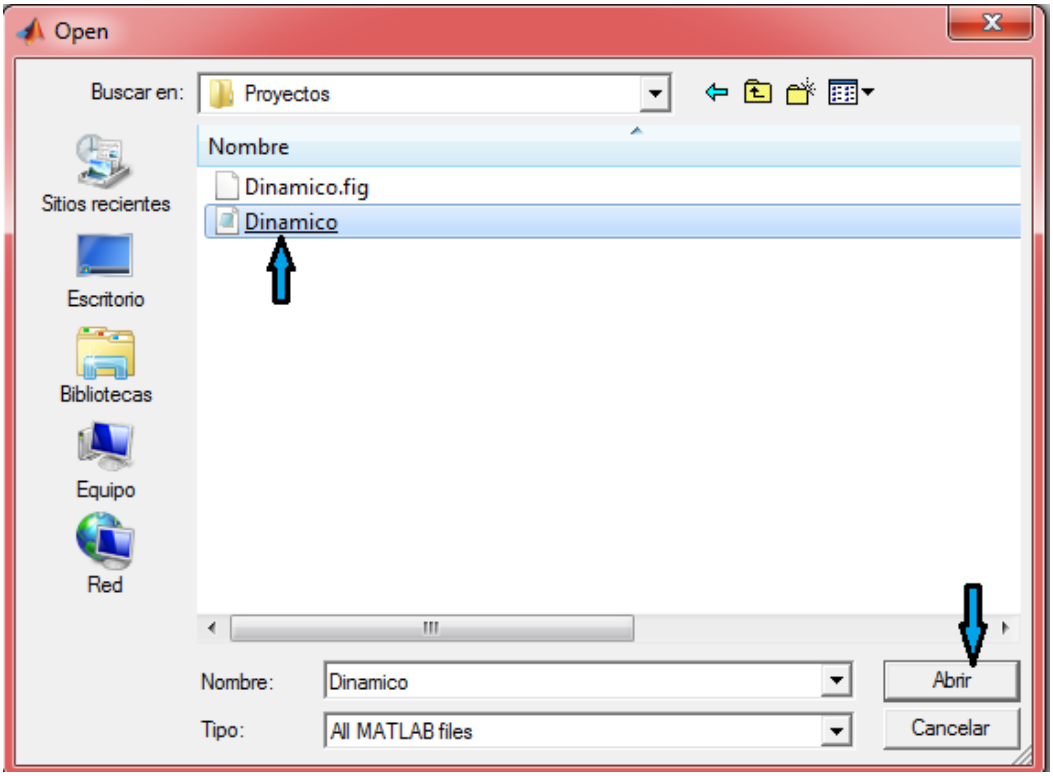


Fig. 34 Ventana Open

3. Una vez que damos abrir la ventana que nos aparecerá de manera inmediata será la ventana de programación (Editor) la cual no deberá sufrir modificaciones para que el programa siga funcionando adecuadamente. En esta ventana encontraremos en la barra de herramientas el botón “correr o ejecutar” (Run) el cual seleccionaremos ya que es el encargado de correr el programa como se muestra a continuación en la Fig. 35.
   En el caso de que de manera accidental se altere o modifique dicha programación, bastará con cerrar la ventana antes de correrla y no guardar las modificaciones.

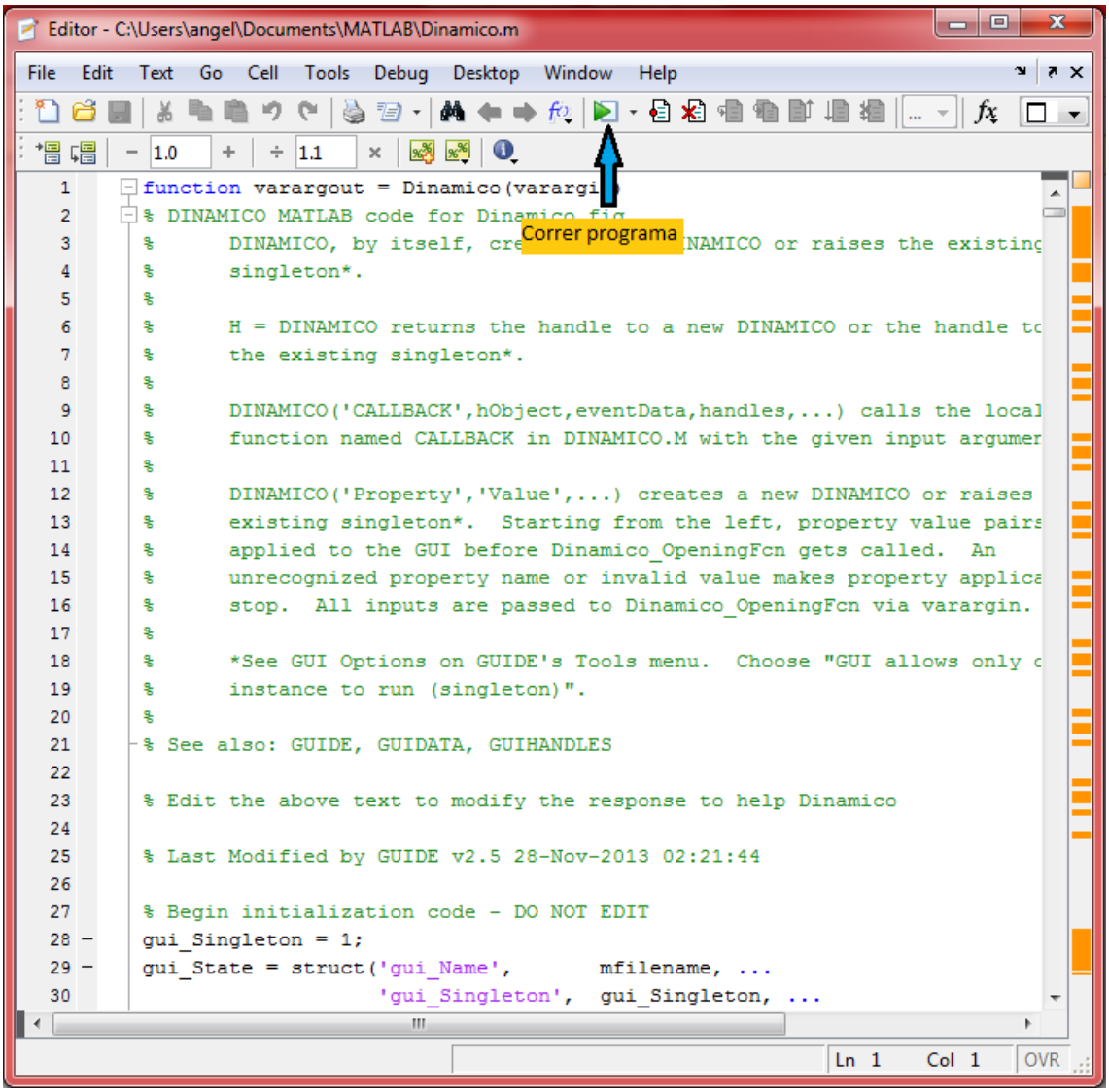


Fig. 25 Ventana de programación

4. Cuando el archivo es corrido nos lleva a la ventana con el nombre del programa, en este caso Dinámico y notaremos que dicho programa se encuentra dividido en dos partes, la parte principal con el tema de "Módulo de diseño de ejes" y una parte secundaria con el tema de "Factores que modifican la resistencia a la fatiga". (Ver Fig. 36)
Debido a que los factores que modifican la resistencia a la fatiga se solicitan en la parte principal del módulo y que estos dependen de condiciones diferentes del eje, se creó la parte secundaria como complemento del principal.

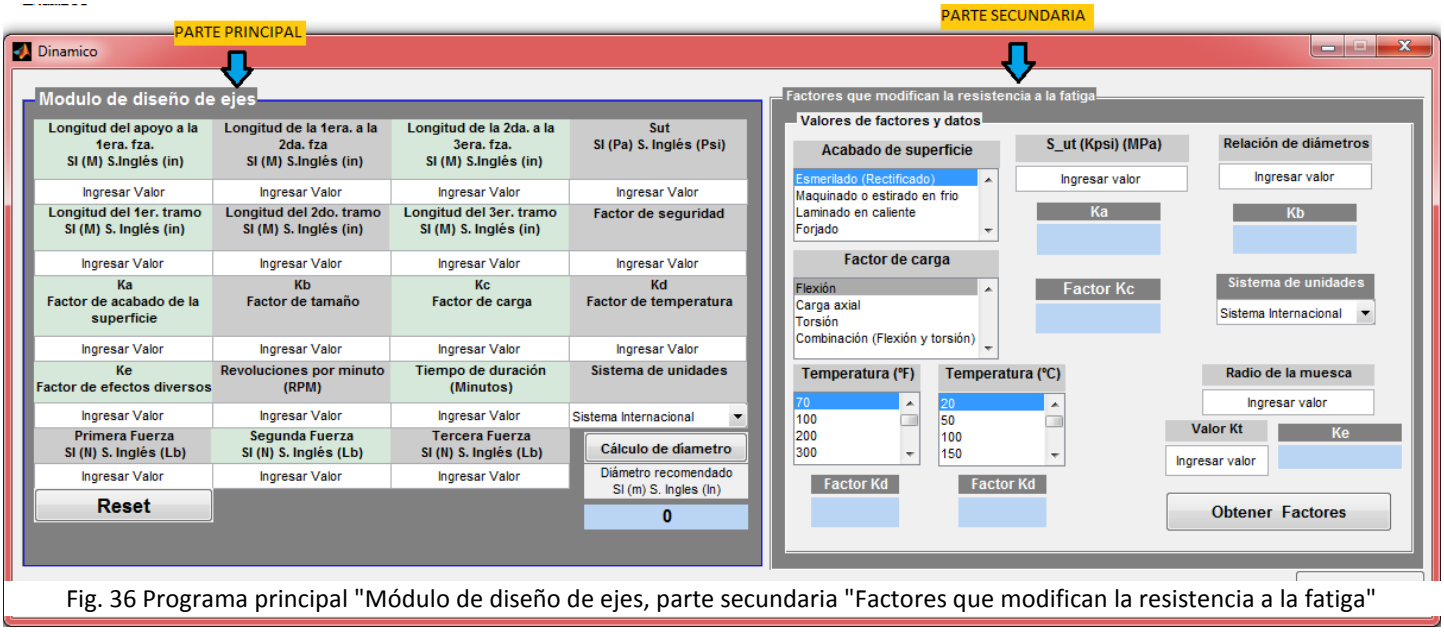
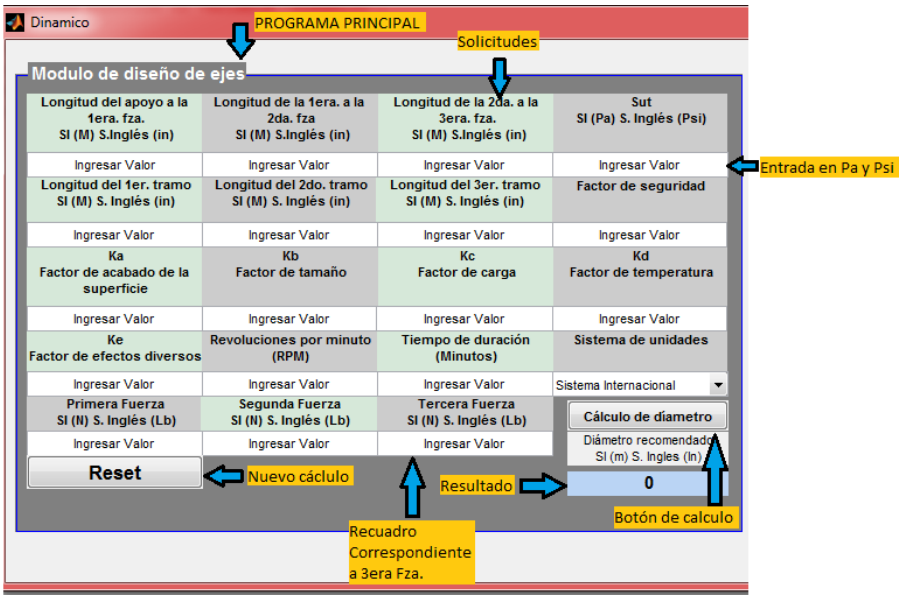


Fig. 36 Programa principal "Módulo de diseño de ejes, parte secundaria "Factores que modifican la resistencia a la fatiga"



5. Los datos que nos otorga el problema y los que dicho programa nos solicite serán ingresados en los recuadros en blanco en los que se citan la frase "ingresar valor", para el

programa principal, prestando atención de colocarlos adecuadamente en el espacio correspondiente, es decir, donde se solicite.

6. El programa realiza cálculos para duración infinita y para duración finita; para el primer caso los valores de entrada en “Revoluciones por minuto” y “Tiempo de duración” deberán ser igual a cero; ver Fig. 37 (a). Para el segundo caso estos mismos valores de entrada deberán ser diferentes de cero, para que el cálculo llevado a cabo sea para una duración determinada; ver Fig. 37 (b).

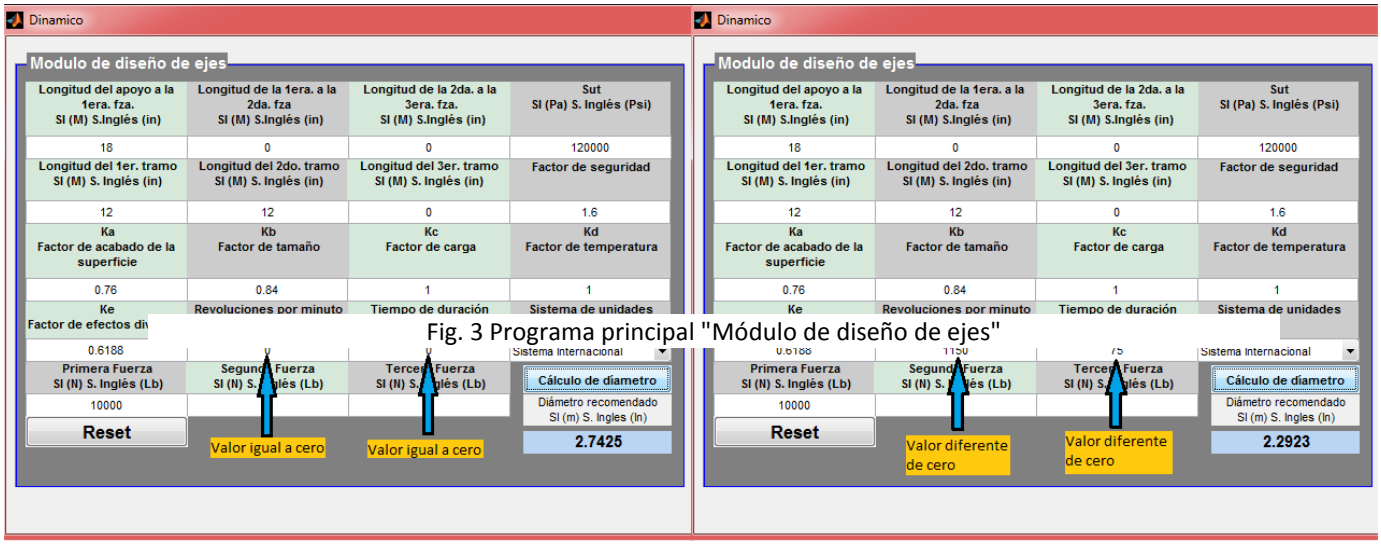


Fig. 3 Programa principal "Módulo de diseño de ejes"

(a) (b)

Fig. 37 Módulo de diseño de ejes. Cálculo duración infinita (a); Cálculo duración finita(b)

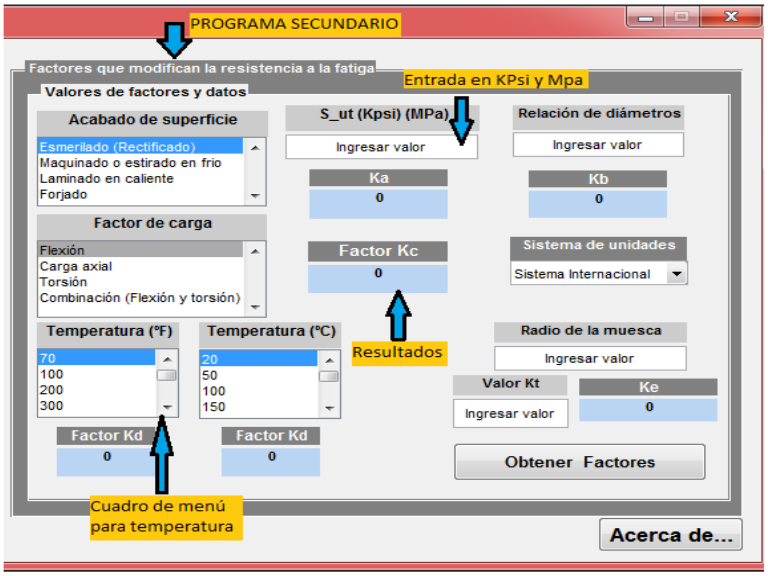


Fig. 38 Programa secundaria "Factores que modifican la resistencia a la fatiga"

7. Del mismo modo en el programa secundario se realizan solicitudes de datos en recuadros citando la frase “ingresar valor” o seleccionando una opción de un menú otorgado. Es importante mencionar que el programa secundario trabaja con un rango de 50-240 (Kpsi) en el Sistema inglés y con un rango de 200-1655 (MPa) en Sistema Internacional. Además de que la entrada en esta, es en Kpsi y MPa a diferencia del principal que sus entradas son en Psi y Pa.

8. Cuando algunos de los valores en el programa principal sea igual a cero esta tendrá que ser escrita en el lugar que le corresponda y no dejar el recuadro con la leyenda “ingresar valor”, ya que de no ser así el programa enviará un mensaje de error, tal como se muestra a continuación en la Fig. 39

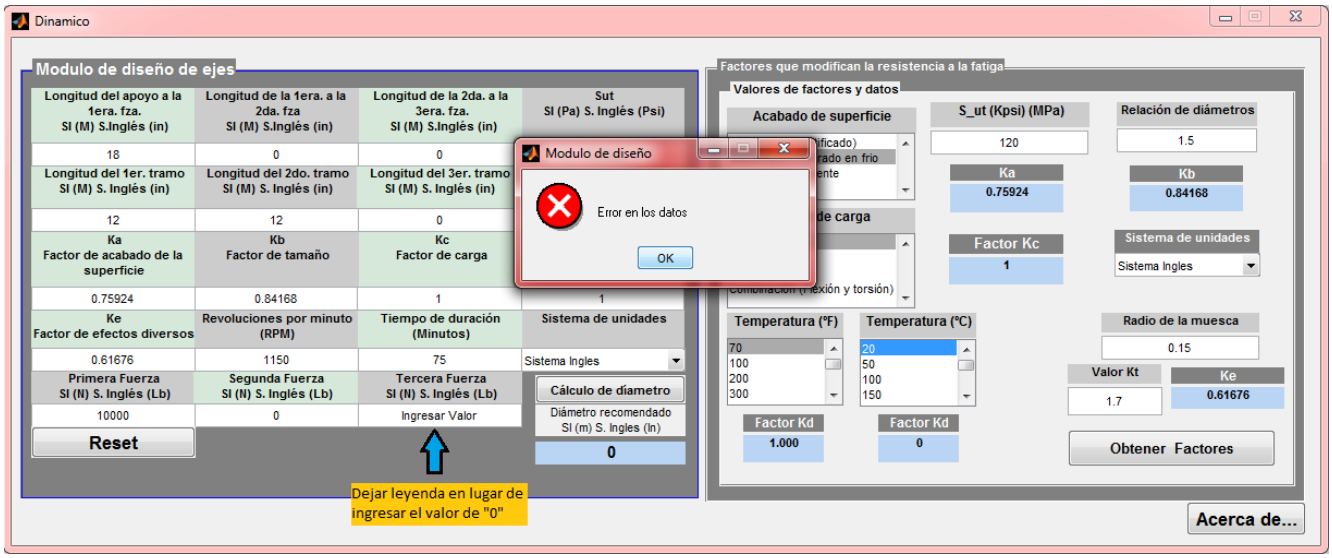


Fig. 39 Mensaje de error

Los casos en los que el programa enviará el mensaje “Error en los datos” será cuando exista una incoherencia en la entrada de datos y lo que deberá realizarse en esos casos será simplemente ingresar los valores de manera adecuada.

9. Una vez que los valores han sido ingresados de manera adecuada en el programa principal, habrá que dar al botón “Cálculo de diámetro” y esperar alrededor de 10 segundos para que el programa arroje el resultado, este botón también abrirá una ventana nueva en la cual se mostrara el eje con el diámetro obtenido y con las medidas de los tramos dados.

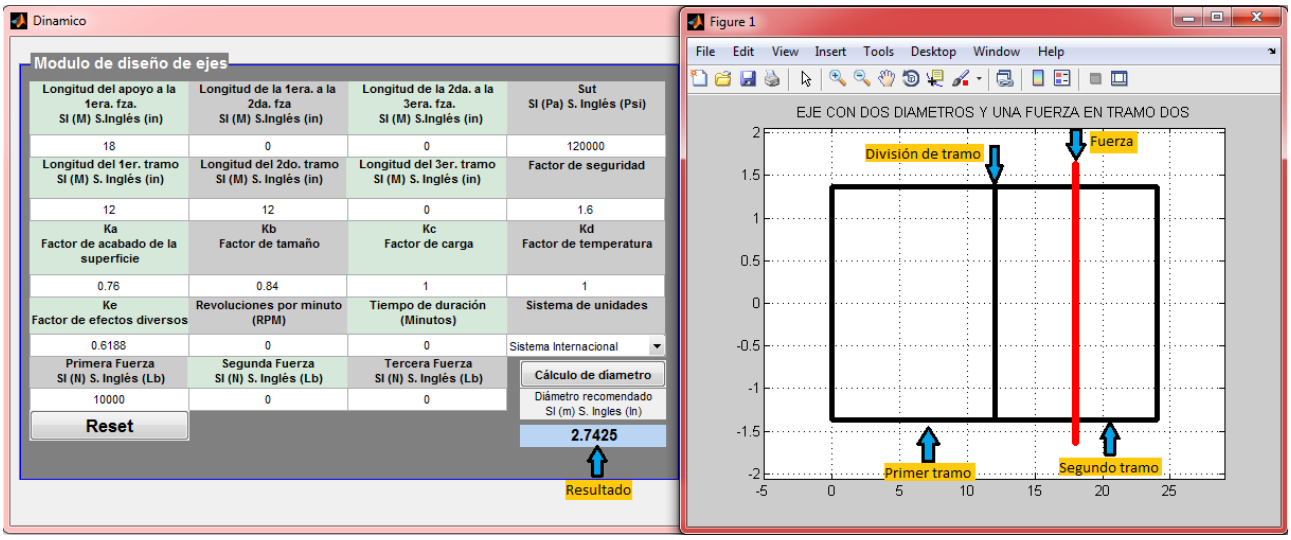


Fig. 40 Grafica de resultado

Generalidades del programa

En esta parte les mostraremos parte del programa ya que es muy extenso y usa los mismos comando solo que aplicado a diferentes ecuaciones, también explicaremos brevemente para que sirven los comandos que hemos utilizado en este programa.

Al ir al editor de MatLab para ver la configuración del programa podemos observar lo siguiente:

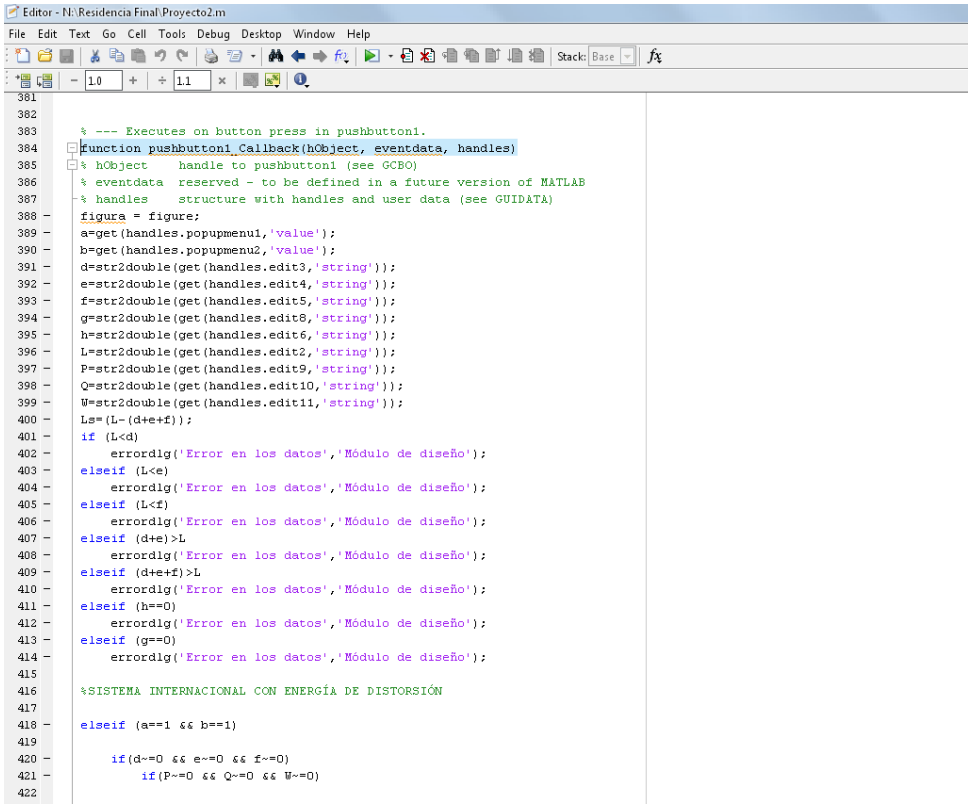


Figura 41. Vista del programa desde el editor de MatLab

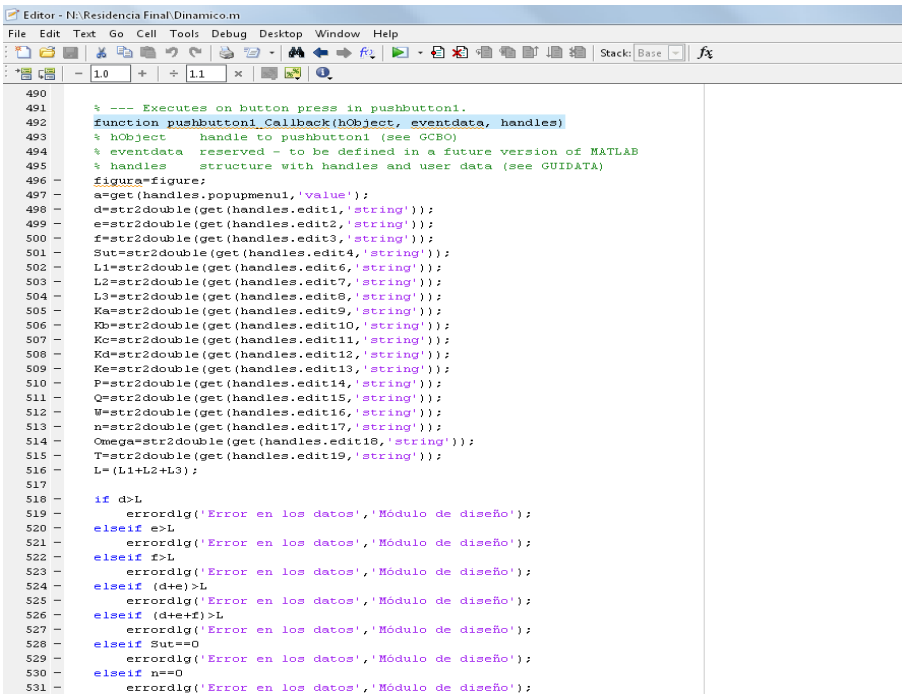


Figura 42. Vista del programa desde el editor de MatLab

Los comandos que se pueden apreciar en la figura son los siguientes:

- If: Este comando se utiliza como una sentencia para crear bucles y de esta manera poder separar las diferentes condiciones del programa.
- elseif: funciona como una segunda sentencia en la cual adquiere o realiza dichas operaciones dependiendo de ciertas condiciones que le demos.
- else: funciona como una última opción en la cual se le dará una condicional para que realice cierta función.
- figure: comando para realizar en una figura aparte la grafica de nuestro eje.
- switch: Es una sentencia que se usa para crear una estructura de control como la que se muestra en la figura siguiente:

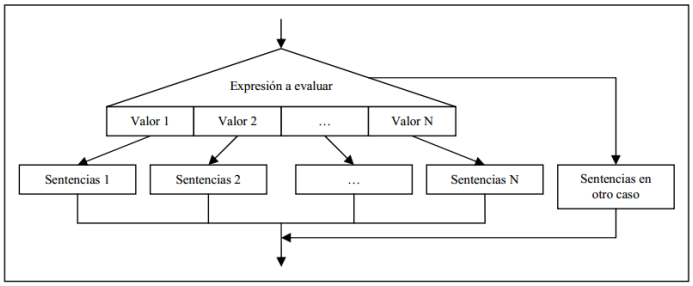


Figura 43. Estructura de la sentencia switch.

La estructura de esta sentencia es la siguiente:

```
switch <Expresión a evaluar>
   case <Valor 1>
      <Sentencias 1>
   case <Valor 2>
      <Sentencias 2>
   case <…>
      <…>
   case <Valor N>
      <Sentencias N>
   otherwise
      <Sentencias en otro caso>
end
```

Figura 44. Estructura de como aparece en MatLab

A continuación les presentaremos otra figura donde les diremos para que sirven cada comando utilizado como lo hicimos anteriormente:

```
%EJE CON UN DIÁMETRO Y UNA FUERZA

 if (P~=0 && Q==0 && W==0 && d~=0 && e==0 && f==0 && L1~=0 && L2==0 && L3==0)
    %REACCIONES EN LOS APOYOS
                RA=(P-P*(d/L));
    %MOMENTO FLECTOR DE CADA CORTE
                M1=(RA*d);
    %CALCULO DE DIAMETRO
                D1=nthroot(((32*n*M1)/(pi*Se)),3);
                set(handles.text17,'string',num2str(D1));
                X=linspace(0,d-1,500);y=linspace(-D1/2,D1/2,500);x1=0;x2=d-1;y1=-D1/2;y2=D1/2;
                plot(X,y1,'k.',X,y2,'k.',x1,y,'k.')
                hold on;grid on
                x6=linspace(d-1,d-1,500);x7=d-1;
                plot(x6,y1,'k.',x6,y2,'k.')
                x11=linspace(d-1,d-1,500);x12=d-1;x13=d-1;
                plot(x11,y1,'k.',x11,y2,'k.')
                x17=linspace(d-1,L,500);x18=d-1;x19=L;
                plot(x17,y1,'k.',x17,y2,'k.',x19,y,'k.')
                x3=linspace(d-0.1,d+0.1,500);y3=linspace (-D1/2-(.2*D1/2),D1/2+(.2*D1/2),500);x4=d-0.1;x5=d+0.1;y4=-D1/2-(.2*D1/2);y5=D1/2+(.
                plot(x3,y4,'r.',x3,y5,'r.',x4,y3,'r.',x5,y3,'r.')
                x8=linspace(d-0.1,d+0.1,500);x9=d-0.1;x10=d+0.1;
                plot(x8,y4,'r.',x8,y5,'r.',x9,y3,'r.',x10,y3,'r.')
                x14=linspace(d-0.1,d+0.1,500);x15=d-0.1;x16=d+0.1;
                plot(x14,y4,'r.',x14,y5,'r.',x15,y3,'r.',x16,y3,'r.')
                axis([-5 (L+5)  -(D1/2+0.5*D1/2) (D1/2+0.5*D1/2)])
                title('EJE CON UN DIAMETRO Y UNA FUERZA');

```

Figura 45. Algunos comando usados

- %: Este signo nos sirve para que indiquemos que esto no forma parte de la programación y así a la hora de correr el programa no lo tome como línea a ejecutar y simplemente ignore dicha línea donde se encuentra el signo %texto, sino solo es para dar una breve explicación de lo que se anda realizando
- linpace: Este comando sirve para decir que haga una línea recta desde un punto determinado hasta otro punto, en la imagen podemos apreciar que dice X=linspace(0,a+1,500), esto quiere decir que se hará una recta desde 0 hasta a+1 y que dicha recta esta dividida en 500 partes.
- axis: sirve para decir al momento de graficar desde que unto hasta que punto mostrara la figura a ala que le hallamos aplicado plot.
- hold on: Sirve para que todos las graficas aparescan en una sola figura.

- grid on: sirve para hacer un enmallado en el plano coordenado.
- plot: este comando sirve para graficar, en la imagen podemos ver claramente que dice: plot(X,y1,'k.',X,y2,'k.',x1,y,'k.'), acá queremos decir que graficara el valor de X y el y1, dándole el color negro a cada parte a la que hemos dividido a la recta de linspace, es por eso que se le pone 'k.' el punto que lleva nos indica que le dará color a cada parte dedicha recta que salga, para este caso graficara una línea horizontal que tendrá las coordenadas (0,y1) y (a+1,y1), el siguiente que es (X,y2,'k.') nos dice que hará una línea horizontal de color negro con las coordenadas (0,y2) y (a+1,y2) y por tenemos la expresión (x1,y,'k.') ultimo nos hará una línea horizontal de color negro que tendrá las coordenadas (0,-d/2) y (0,d/2). En la siguente figura les mostraremos un ejemplo con valores ya asignados de como quedaria plot(X,y1,'k.',X,y2,'k.',x1,y,'k.'):

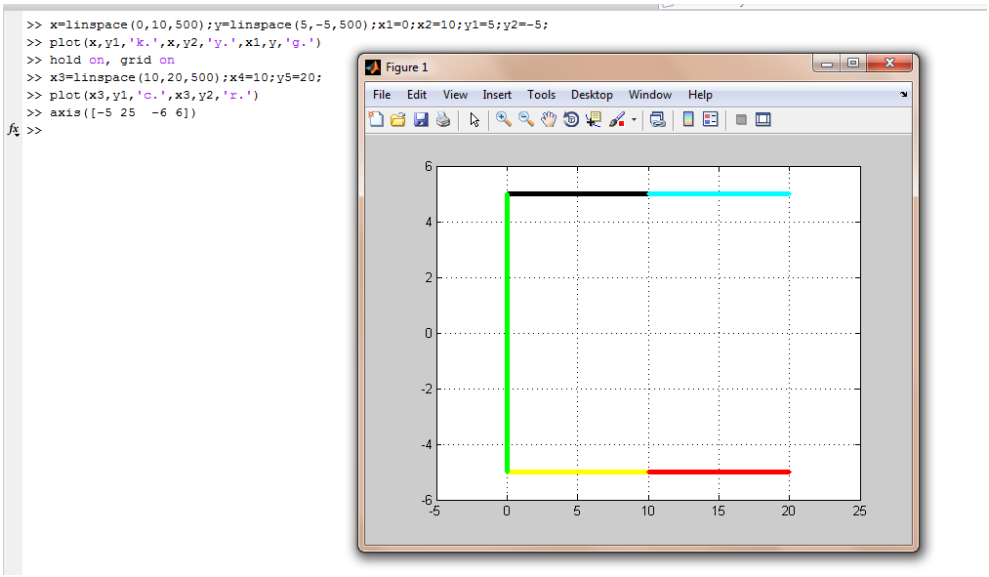


Figura 46. Ejemplo de cómo usar plot.

En la figura 48 podemos ver claramente el uso de los comandos linspace, plot, gridon, holdon y axis, los colores que deseemos que contengan nuestras graficas debe ir siempre entre comillas simples 'color'.

Los comandos antes mencionados son todos los que se utilizaron durante el diseño de este programa, solo que en expresiones diferentes pero son todos los comandos utilizados en dicho programa.

CONCLUSIONES

Nuestro proyecto cumplió con las expectativas requeridas, estamos seguros que nuestro software será de útil ayuda para complementar los trabajos diario que realiza un Ing. Mecánico, el cual ayudara tanto en el ámbito industrial como en el académico, ya que dicho programa será utilizando también en las aulas de diferentes instituciones, con lo cual, los alumnos tendrán una herramienta sencilla y potente para poder diseñar ejes que estarán sometidos a cargas estáticas o cargas variables.

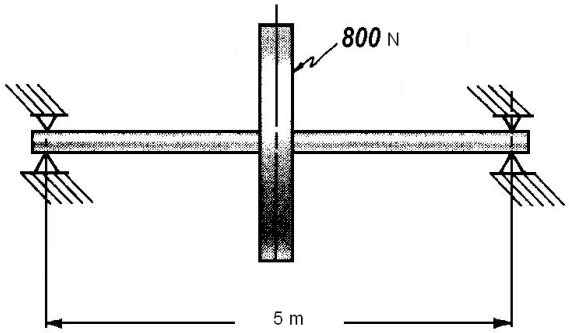
El facilitar el resultado de los cálculos requerido en materias como diseño es un gran paso para reafirmar nuestras dudas y así centrarnos en aprender cosas que verdaderamente nos puedan servir.

Para que este proyecto fuera completo, se fueron añadiendo diferentes detalles los cuales fueron haciendo al programa un paquete completo para el diseño de ejes.

ANEXOS

Validación del programa

Ejemplo 1.- Un eje con una carga en el centro de 800 N está hecho de aluminio fundido 195-T6 con una resistencia a la fluencia de 160 MPa se maquina hasta el diámetro final con un factor de seguridad de 1.10 y una longitud de 500 mm. Calcule el eje recomendado para dicho eje mediante la teoría de energía de distorsión.



P = 800 N

d = 2.5 m

L = 5 m

Sy = 160 MPa

n = 1.10

Solución:

Encontramos las reacciones:

RA = (P-P*(d/L));
RA = 400

El momento de flexión en el corte es:

$M1 = RA(d)$ = 1000 N*m

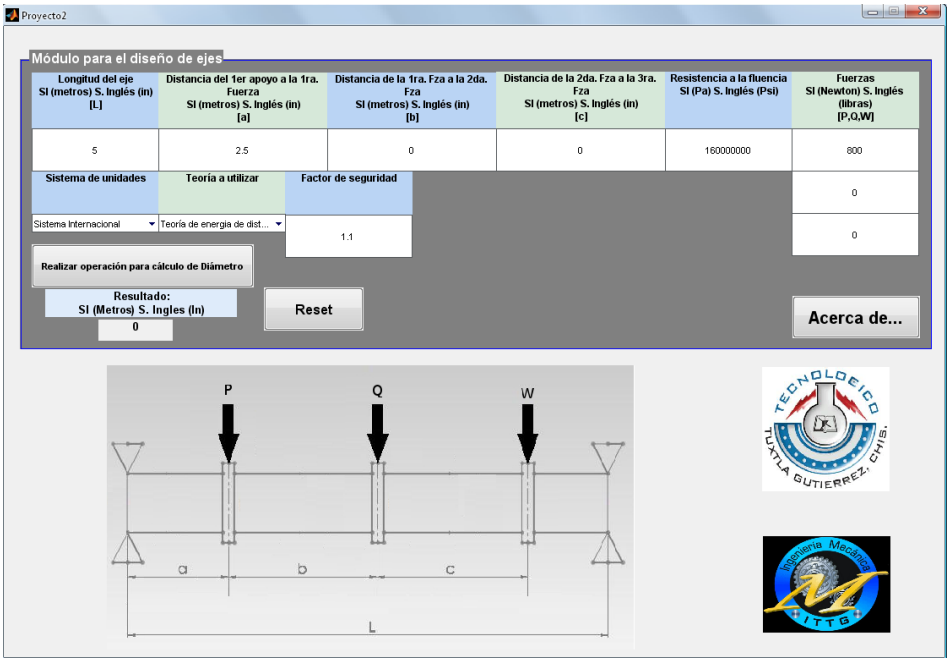
Luego obtenemos el diámetro:

$\frac{S}{n_s} = \frac{32M_p}{\pi d^3}$ O bien $\frac{160(10^6)}{1.1} = \frac{32(1000)}{\pi d^3}$

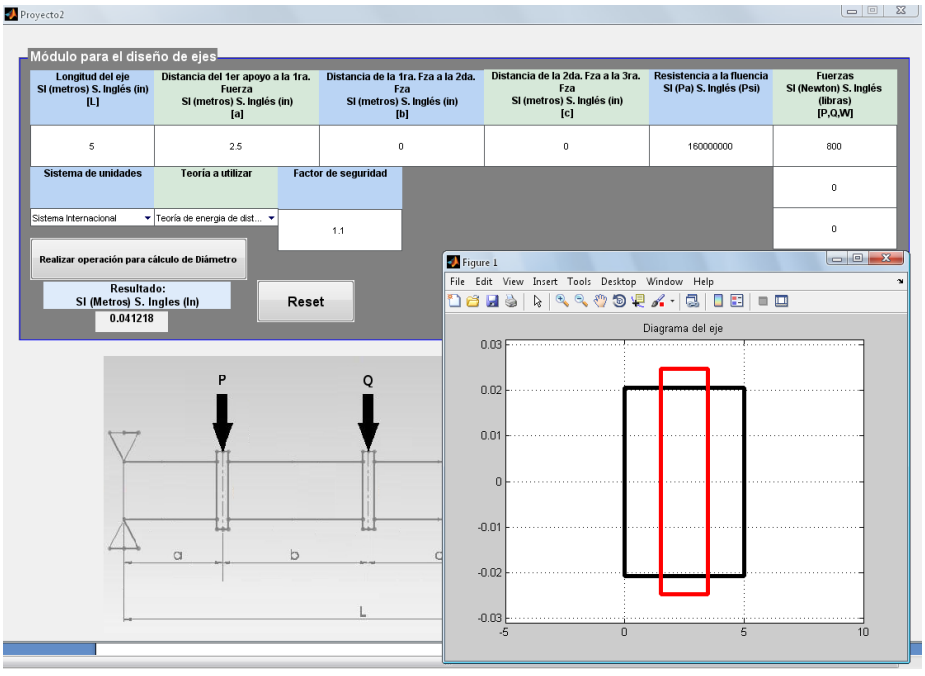
La resolución da para el diámetro, el valor de **d = 0.041218 m.**

Utilizando Software:

- Ingresamos los datos:

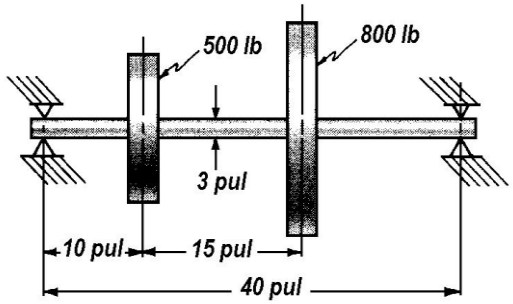


- Damos click al botón para realizar operación para cálculo de diámetro.



Y el valor arrojado por el programa es el mismo que nos da al realizarlo de la manera clásica, un diámetro recomendado para el eje de: **0.041218 m.**

Ejemplo 2.- Para el eje que se muestra en la figura, determinar: a) el diámetro recomendado para un factor de seguridad de 1.8 con la TED (S_y= 44 Kpsi).



P = 500 lb

Q = 800 lb

d = 10

e = 15

L = 40

Sy = 40 Kpsi

n = 1.8

Solución:

Encontramos las reacciones:

RA= (Q+P-Q*((d+e)/L)-P*(d/L))

RA = 675 lb

El momento de flexión en los cortes es:

$M1 = RA(d)$ = 6750 lb*in

$M2 = RA(d + e) - P(e)$ = 9375 lb*in

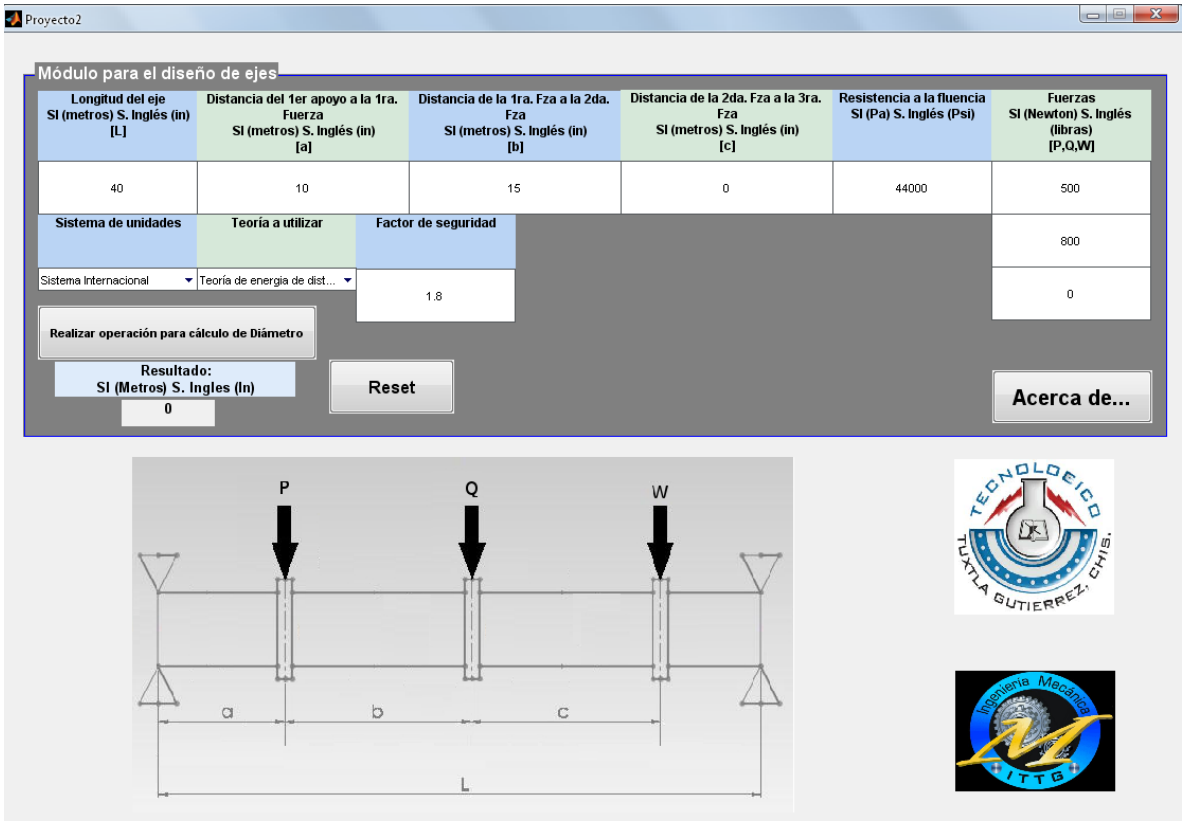
Luego obtenemos el diámetro:

$\frac{S}{n_s} = \frac{32M_p}{\pi d^3}$ O bien $\frac{44(10^3)}{1.8} = \frac{32(9375)}{\pi d^3}$

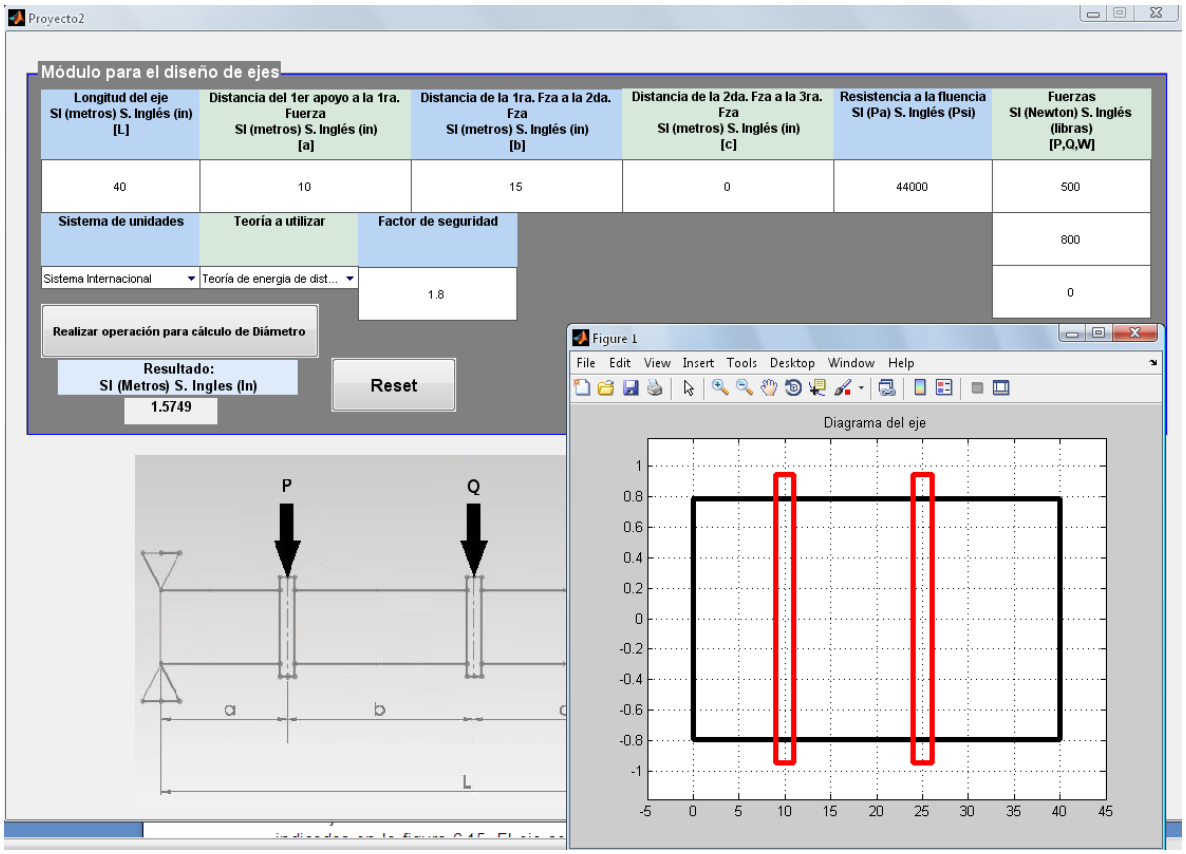
La resolución da para el diámetro, el valor de **d = 1.5749 pulg.**

Utilizando Software:

- Ingresamos los datos:



Damos click al botón para realizar operación para cálculo de diámetro.



Y el valor arrojado por el programa es el mismo que nos da al realizarlo de la manera clásica, un diámetro recomendado para el eje de: **1.5749 in.**

Ejemplo 3. Diseño en Ingeniería Mecánica Shigley 3era. Edición pág. 242

A un eje de sección circular se le carga por flexión estática aplicándole las tres fuerzas indicadas en la figura 6.15. El eje se construirá con una barra de acero estirado en frío UNS G10100 que tiene una resistencia de fluencia tabulada de 44 Kpsi. El factor de seguridad con base en la resistencia ha de ser $n_s = 1.20$.

La naturaleza de las cargas F_1, F_2 y F_3 es tal que puede esperarse que ocurran diferentes sobrecargas relativas si el eje no es utilizado apropiadamente. Debido a esto, se eligen $n_1 = n_3 = 1.25$ y $n_2 = 2.40$.

Calcúlese un diámetro adecuado para el eje por medio de n_1, n_2, n_3 y n_s

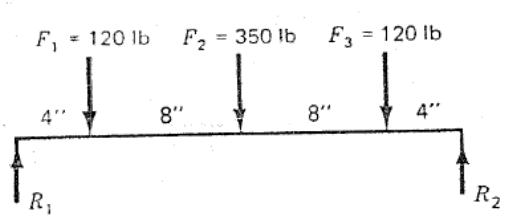


FIGURA 6-15

Solución:

Utilizando $F_p = n_i F_i$, se obtiene

$F_{1.p} = 1.25(120) = 150$

$F_{2.p} = 2.40(350) = 840$

$F_{3\ .p} = 1.25(120) = 150$

El máximo momento de flexión se presenta en el centro de la viga. Su magnitud es

$$M = 12\left(\frac{F_1 + F_2 + F_3}{2}\right) - 8F_1$$

Y así el momento permisible es

$$M_p = 12\left(\frac{150 + 840 + 150}{2}\right) - 8(150) = 5640\ lb * pulg$$

El esfuerzo permisible vale

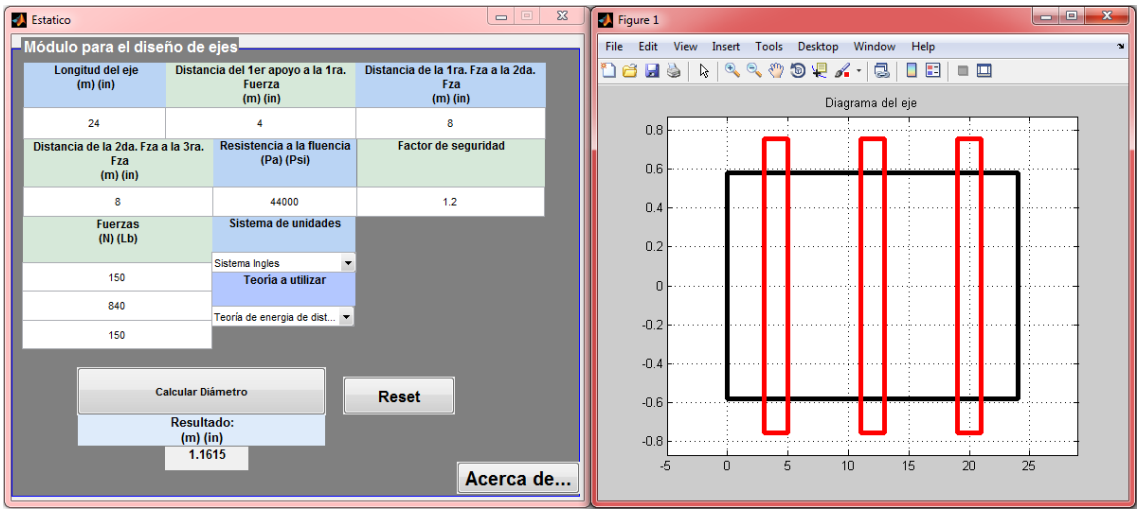
$$\sigma_p = \frac{M_p}{I/c} = \frac{32M_p}{\pi d^3}$$

Luego; se tiene que

$\frac{S}{n_s} = \frac{32M_p}{\pi d^3}$ O bien $\frac{44(10^3)}{1.20} = \frac{32(5640)}{\pi d^3}$

La resolución da para el diámetro el valor de **d = 1.161 pulg.**

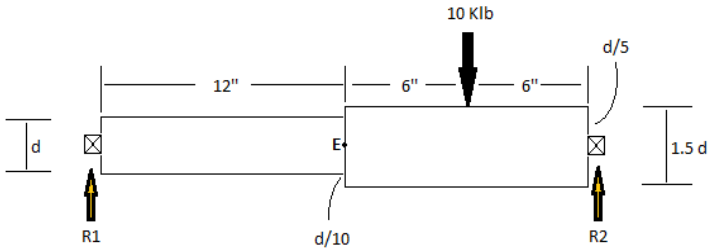
Utilizando software:



Resultado obtenido **d = 1.1615** pulg.

Problema propuesto y resuelto por M.C. Ignacio Arrioja Cárdenas

Ejemplo 4: El eje de transmisión que se muestra tiene reacciones R_1 y R_2 en sus cojinetes, gira a 1150 RPM y soporta una carga flexionante de 10 Klb. Las especificaciones son: acero dúctil con S_{ut} = 120 Kpsi y S_y = 90 Kpsi. El eje debe ser maquinado y tener una duración de 75 min. Utilice un factor de diseño de 1.6 y obtenga el diámetro apropiado.



Solución:

$t = 75\ min$

$\omega = 1150\ RPM$

$n_s = 1.6$

$S_e' = 0.5S_{ut} = 60$ Kpsi

R_1 = 2500 lb

R_2 = 7500 lb

M_E = 30000 lb · in

$$\sigma_{max} = \frac{32M}{\pi d^3} = \frac{32(30000\ lb \cdot in)}{\pi d^3} = \frac{305.58}{d^3}\ Kpsi$$

$$\sigma_{max} = \sigma_{min} = \sigma_v = \frac{305.58}{d^3}\ Kpsi$$

$$N = \omega t = 1150(75) = 86250 \text{ ciclos}$$

Relación de diámetro = 1.5

Factores que modifican la resistencia a la fatiga

$$K_a = 2.70(120^{-0.265}) = 0.76$$

$$K_b = 0.879(1.5^{-0.107}) = 0.84$$

$$K_c = K_d = 1$$

$$K_e = \left(\frac{1}{1.616}\right) = 0.6188$$

S_e = (0.76)(0.81)(1)(1)(0.6188)(60*10^3) = 23.7 Kpsi

$$n_s\sigma_v = aN^b$$

Obteniendo valor de a y b

$$a = \frac{(0.9S_{ut})^2}{S_e} = 492.5$$

$$b = -\frac{1}{3}log\frac{(0.9S_{ut})^2}{S_e} = -0.22$$

Sustituyendo

$$\frac{1.6(305.58)}{d^3} = 492.15(86250^{-0.22})$$

Resolviendo obtenemos que **d = 2.3 in**

Para una duración infinita tenemos

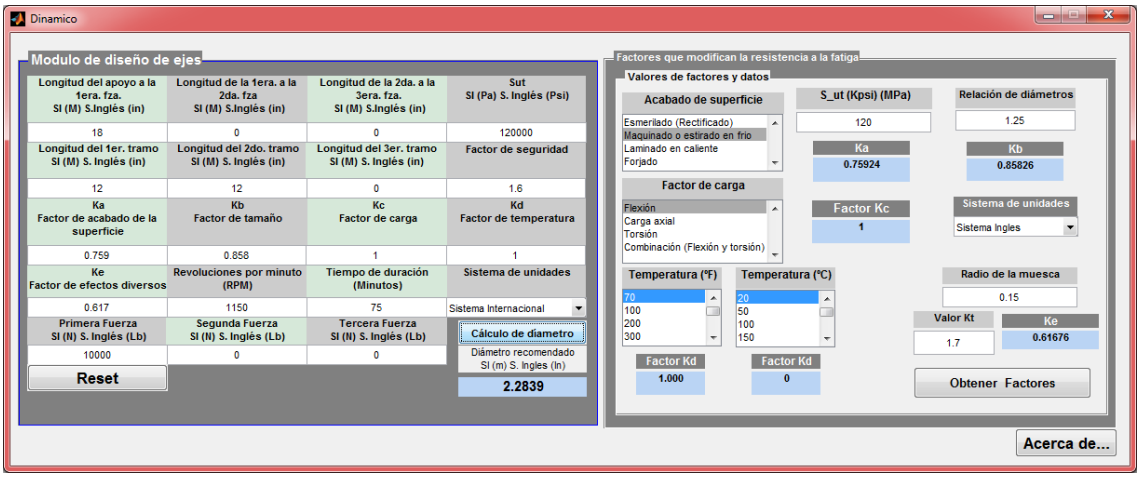
$$\frac{\sigma_v}{S_e} + \frac{\sigma_m}{S_{ut}} = \frac{1}{n_s}$$

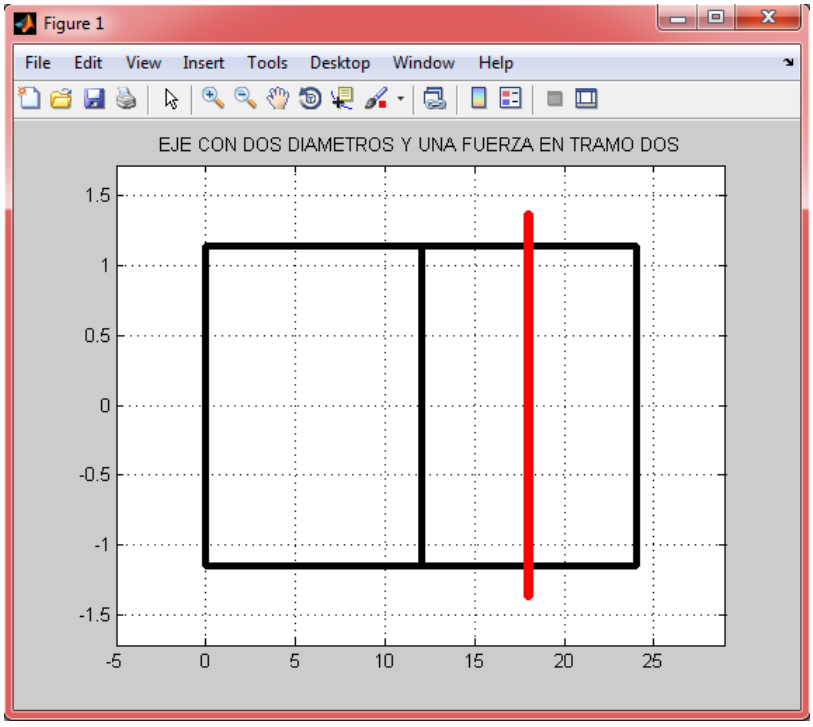
Pero $\sigma_m = 0$; por tanto

$$\frac{\frac{305.58}{d^3} Kpsi}{27.3\ Kpsi} = \frac{1}{n_s}$$

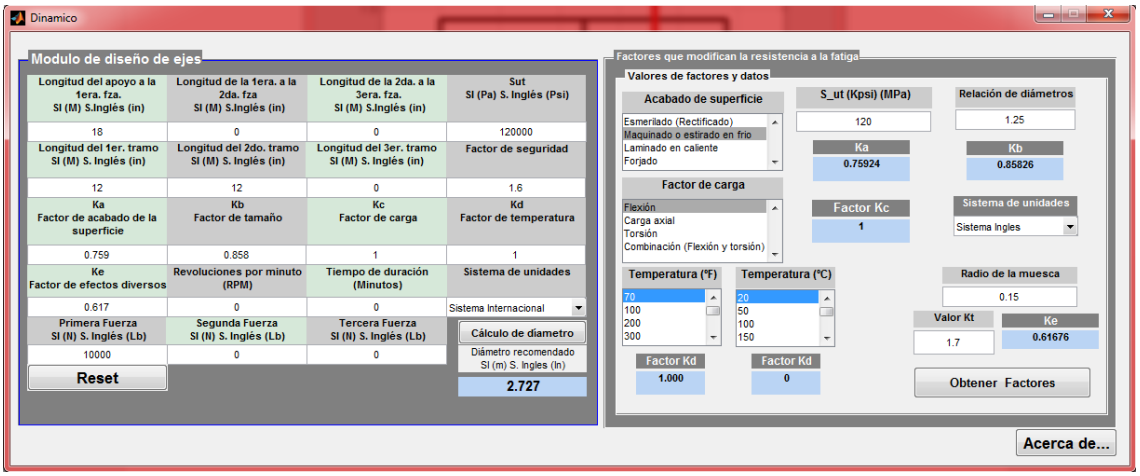
Resolviendo obtenemos para una duración infinita **d=2.743**

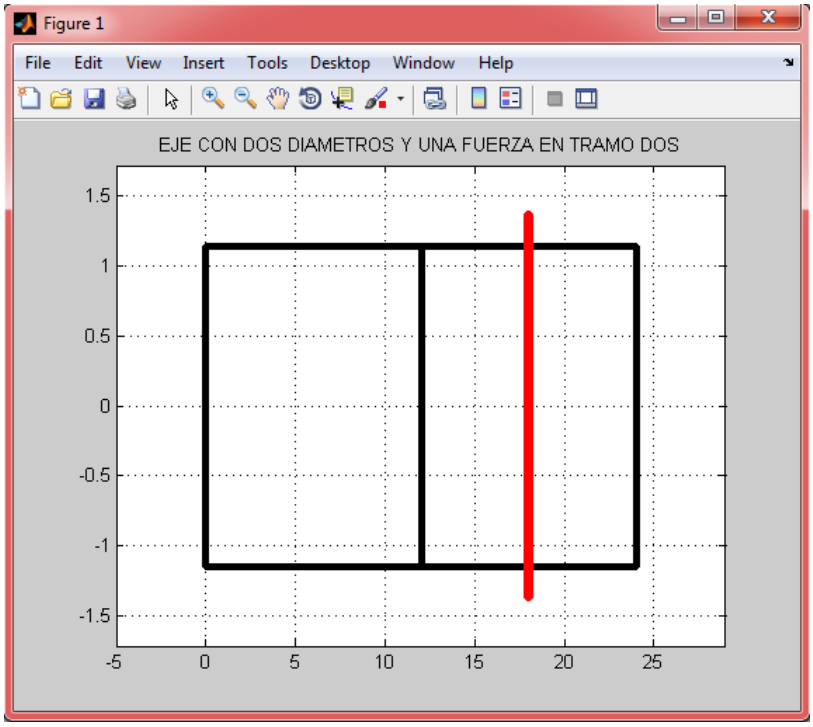
Resolviendo por medio de Software





Resultado obtenido para un duración finita **d = 2.2839 in**





Resultado obtenido para una duración infinita **d = 2.727**

BIBLIOGRAFIA

- Joseph Edward Shigley, “Diseño en Ingeniería Mecánica”, Editorial Mc Graw Hill, Octava edición.
- Amos Gilat, “Matlab, una introducción con ejemplos prácticos”, Editorial Reverte
- AngelGarcimartin, “Introducción al MatLab”, [En linea], http://fisica.unav.es/~angel/matlab/matlab0.html
- Javier Gracía d Jalón; Jóse Ignacio Rodríguez; Jesús Vidal,”AprendaMatLab 7.0 como si estuviera en primero”, [En linea], http://mat21.etsii.upm.es/ayudainf/aprendainf/Matlab70/matlab70primero.pdf
- Principe Gavidia; Lineker L., “Priemra práctica de MatLab”, [En linea], http://prinlider.onlinewebshop.net/Practica%2001-Metodos.pdf
- Neiker D.; Arévalo Roque, “Diseño por rigidez y análisis dinámico”,[En linea], http://www.wiziq.com/tutorial/364303-analisis-de-vibraciones