

# TREN: una ayuda para el estudio de los engranajes

José B. RAMIRO DÍAZ, Ángel ALCÁZAR DE V. RICO y Consuelo FERNÁNDEZ JIMÉNEZ  
E.U.I.T.Aeronáutica. Pl. Cardenal Cisneros, 3. 28040, Madrid, España  
{j.ramiro, a.alcazar, consuelo.fernandez}@upm.es

## RESUMEN

El programa TREN implementa un lenguaje que permite diseñar trenes de engranajes con perfil de evolvente controlando todos los parámetros de las ruedas (módulo, addendum, dedendum, juego entre dientes, ángulo de presión), representarlos gráficamente y simular su funcionamiento manteniendo el contacto entre los dientes. Además realiza el cálculo de sus relaciones de transmisión y de contacto y detecta la aparición de interferencia entre dientes. Admite un funcionamiento interactivo o pre-programado mediante scripts. Ambos modos pueden alternarse dinámicamente. Aplicado a la asignatura de Mecanismos de la E.U.I.T.Aeronáutica ha mostrado ser una ayuda muy eficaz para comprender el contacto y la transmisión de movimiento entre ruedas dentadas así como el funcionamiento de trenes simples y compuestos, incluidos los sistemas planetarios.

**Palabras claves:** tren de engranajes, perfil de evolvente, módulo, ángulo de presión, socavación, interferencia.

## 1. INTRODUCCIÓN

La aplicación TREN está diseñada para centros docentes de carácter técnico y donde la ingeniería mecánica esté presente. Se ha desarrollado en la Escuela Universitaria de Ingeniería Técnica Aeronáutica de la Universidad Politécnica de Madrid (España) con una doble finalidad:

1. Facilitar la comprensión de los parámetros de diseño de un engranaje recto o helicoidal (entre ejes paralelos) así como su funcionamiento y el análisis del contacto entre dientes, tal y como el alumno estudia en los libros de texto de la asignatura de mecanismos [J.E.Sighley, 1998] [Robert L.Norton, 1999]
2. Servir de herramienta de trabajo para prácticas.

El desarrollo del programa obedece a la necesidad de disponer de una herramienta flexible que genere interactivamente imágenes de trenes de engranajes relativamente

complejos y perfiles de diente que acompañen explicaciones de clase. Inicialmente surgió como una herramienta para crear transparencias estáticas que mostrasen las distintas fases del contacto entre dientes y sus limitaciones, siempre en el caso de perfiles de evolvente de circunferencia. Posteriormente se añadió una librería gráfica [T.J.Pearson, 2001], lo que simplificó su uso. Finalmente se implementó un interprete de comandos y se definió una sintaxis con un conjunto de órdenes que potencian su capacidad interactiva, le dan flexibilidad y permiten usarlo como un entorno plenamente interactivo, dotado de un lenguaje propio. Todo ello sin perder la capacidad de generar imágenes preconcebidas en distintos formatos.

Existen programas para dibujar ruedas y hacer cálculos sobre engranajes [Gearpack, 1999] y otros más orientados a entornos CAD [K-prof, 2002], que trabaja en AUTOLISP (para AUTOCAD). Aparte del coste económico que puedan tener, los autores de éste trabajo estiman que el enfoque pedagógico del programa TREN supone una ventaja adicional puesto que se ha desarrollado específicamente para facilitar la comprensión del proceso de engrane entre dientes y para ayudar a relacionar la forma del diente con sus parámetros geométricos. Las primeras tentativas de desarrollar un software similar al presentado se plasmaron en rutinas LISP y se mostraron muy útiles para hacer presentaciones estáticas (transparencias) o animaciones preestablecidas. Sin embargo su empleo por parte de los alumnos era mucho más limitado, por lo que se decidió realizar un software específico.

## 2. DESCRIPCIÓN DEL PROGRAMA

### Formas de Ejecución

El programa TREN tiene dos modos básicos de funcionamiento: i) interactivo y ii) pre-programado. Ambos pueden mezclarse y sirven para propósitos distintos:

El modo interactivo abre una consola de texto donde introducir las órdenes y se reciben mensajes de error; y otra gráfica donde se produce la salida de los trazados. Permite ir modificando el diseño sobre la marcha o estudiar el contacto entre dientes ampliando la zona de interés, etc..

En el modo pre-programado se sustituye la entrada por teclado desde la consola por las órdenes contenidas en un fichero, que se van ejecutando secuencialmente hasta el final. Permite definir sistemas complejos y reutilizarlos, o modificarlos posteriormente sin tener que redefinirlos, o crear presentaciones programadas.

Entre los comandos existen dos que cambian de un modo de ejecución a otro. Se puede así almacenar la definición de un tren complicado en un fichero, cargarlo rápidamente y pasar a modo interactivo para trabajar con él.

### Concepción del Lenguaje.

El conjunto de comandos creados para esta aplicación, junto con el intérprete de comandos implementado, es en realidad un lenguaje de programación.

Este lenguaje se compone de unos cuarenta comandos para definir las ruedas, su geometría y sus relaciones, para controlar la ventana de trazado, para organizar animaciones y para permitir cierta interacción con el sistema operativo.

### Entrada: Tipos de Líneas.

Básicamente hay tres tipos de línea que acepta el intérprete de comandos:

**Definición o redefinición de ruedas:** permite definir la geometría de una rueda y su conexión con otras del tren. La definición de una rueda consta de cinco campos obligatorios y termina con el carácter [;] . Un ejemplo de definición sería: **1 0 e 17 0;** . Estos campos son:

**identificador de rueda** (1) es un número natural, consecutivo que identifica cada rueda que se define

**conexión** (0) es un número entero que identifica el papel y conexión de la rueda dentro del tren. Un cero (0) establece la rueda como motriz, un número positivo indica qué rueda conduce a la que estamos definiendo, y un número negativo indica que ambas ruedas son solidarias y coaxiales

**tipo de rueda** (e) indica si la rueda es dentada exterior (E) o dentada interior (I)

**número de dientes** (17) de la rueda

**giro** (0) ángulo que se debe girar la rueda antes de dibujarla, si es motriz o solidaria, o el ángulo entre los ejes para una rueda conducida. Estos valores pueden incrementarse automáticamente en las animaciones para simular el funcionamiento del tren.

A continuación del carácter [;] se pueden especificar los parámetros geométricos que definen el diente (módulo, ángulo de presión, addendum, dedendum y espesor circular

del diente) u otros como el color en el que debe dibujarse la rueda, círculos o líneas asociadas que deben aparecer, etc. Si ninguno de estos modificadores aparece en la definición o redefinición de rueda, se toma su valor por defecto, que puede establecerse independientemente.

**Modificación de variables:** cambia el valor de las variables que controlan el comportamiento del programa, como el valor por defecto del módulo de las ruedas, el color del fondo, el número de imágenes de una animación, o los límites de la ventana gráfica.

**Comandos:** son órdenes que se ejecutan directamente, son de dos tipos fundamentales:

1. pensadas para el trazado gráfico (dibujar el tren actual, ampliar la ventana, desplazar el centro, iniciar animación, etc, ...)
2. de interacción con el sistema operativo (salvar el trazado a un fichero, salvar el estado del programa, cargar un fichero de órdenes, pasar a modo interactivo, etc...)

En una sola línea se pueden incluir varias ordenes o asignaciones de variables (con ciertas restricciones) que son ejecutadas sucesivamente.

### Salida: Tipos de Salidas

La salida del programa, en condiciones normales, es dirigida a un terminal gráfico, si bien se pueden exportar gráficos y animaciones en formato postscript. Alternativamente el programa puede funcionar sin entorno gráfico. En ese caso la salida es un fichero ASCII que contiene las coordenadas cartesianas que definen cada una de las ruedas definidas, las líneas entre centros, y los centros de las ruedas. Estos ficheros pueden emplearse como entrada en otro programa gráfico cualquiera. Lógicamente estos conjuntos de puntos definen polilíneas que al cerrarse generan una aproximación a cada rueda. Cada flanco de diente se perfila inicialmente con 10 puntos. Este valor puede modificarse fácilmente en el programa fuente para aumentar la precisión, a costa de sacrificar la velocidad. Se ha observado que la precisión de trazado de diente es suficiente incluso ampliando el dibujo tanto como sea necesario para estudiar detenidamente la aparición de interferencia entre dientes.

### Documentación y Ayuda.

El programa es fácil de utilizar y dispone de dos fuentes de ayuda. La primera es un pequeño documento de cuatro páginas en el que se presenta el programa y se describen los comandos y su sintaxis. La segunda ayuda se produce como respuesta a un comando especial (*ayuda* o *?* en la sintaxis del lenguaje) tras el cual aparece un listado y breve descripción de las órdenes reconocidas.

### 3. ASPECTOS DE LA PROGRAMACIÓN

#### Entorno de Desarrollo, Limitaciones y Requisitos.

La aplicación se ha desarrollado en el entorno de sistemas operativos Linux. Esta elección se ha hecho por distintas razones, entre ellas la filosofía del propio sistema. Está programado en FORTRAN77 haciendo uso de la librería gráfica PGPLOT. Se trata de un lenguaje ampliamente extendido en la comunidad científica y resulta portable a otros sistemas operativos. La aplicación tiene algunas limitaciones, pequeñas dado el bajo nivel de requisitos que impone:

- El uso (mínimo) de las peculiaridades del compilador empleado (g77) [Manual g77, 2004]
- El empleo de una librería gráfica (pgplot [T.J.Pearson, 2001]), para darle flexibilidad y capacidad interactiva, pero que no forma parte estándar de ninguna distribución Linux, hasta donde conocen los autores (se ha instalado en Debian, Ubuntu y Fedora ).
- Necesidad de entorno gráfico para explotar su capacidad interactiva. Aún así, puede funcionar en consola de texto de forma no interactiva y generar ficheros ASCII con la información de una imagen que puede procesarse con cualquier programa gráfico. (Inicialmente, el programa utilizaba GNUPLOT [Thomas Williams 2004] para la representación gráfica).
- Aunque los trazados estáticos o avances paso a paso se realizan ágilmente en cualquier sistema, las animaciones pueden resultar lentas en ordenadores poco potentes (por debajo de 200MHz). Sin embargo ésta no es una limitación achacable al programa, teniendo en cuenta los cálculos requeridos.

#### Partes del Programa.

El programa está desarrollado de forma compacta; salvo la librería PGPLOT, el resto de las funciones se definen internamente para ajustarse a las necesidades del proyecto. El núcleo fundamental del programa está formado por las siguientes subrutinas y/o funciones, en las que se indica la información de entrada y la que produce de salida :

**Generación de un diente:** genera una sucesión de puntos que define el exterior de un diente y su hueco.

**entrada:** addendum, dedendum, ángulo de presión, juego entre dientes, tipo de rueda, módulo, número de dientes.

**salida** vector de puntos que define un diente y su hueco.

**Generación de una rueda:** genera una sucesión de puntos que define el exterior de una rueda.

**entrada:** la salida de la rutina anterior, número de dientes, posición del primer diente y del centro.

**salida:** vector de puntos que define el exterior de la rueda.

**Ubicación de una rueda conducida:** determina la posición del centro de una rueda conducida.

**entrada:** número de dientes, tipo de rueda, addendum, dedendum, juego entre dientes y de la línea entre centros.

**salida:** centro, giro (para asegurar el contacto), módulo y ángulo de presión (heredados de la motriz).

**Definición de ruedas:** incrementa (o modifica si es una redefinición de rueda) la base de datos de las ruedas definidas con sus parámetros de diseño (módulo, tipo de rueda, ángulo de presión, número de dientes, addendum, dedendum, juego entre dientes), y otros internos como la conexión (si es motriz o conducida), color, velocidad si es motriz o velocidad de giro de la línea entre ejes si es conducida (para trenes planetarios).

**Trazado y animación:** procesa la información actual (incluyendo el giro previo) de cada rueda motriz en función de su velocidad angular e instante y calcula la correspondiente de cada rueda conducida para dibujar después todas las ruedas. En el caso de una animación procesa consecutivamente todos los fotogramas pedidos haciendo incrementar el tiempo.

**Procesado de ordenes:** compara la orden recibida con la lista de comandos conocidos, comprueba su sintaxis y ejecuta la orden.

**Programa principal:** dispone el entorno para la ejecución solicitada (interactiva o pre-programada), recibe los comandos de entrada e interactúa con el sistema operativo abriendo, creando o cerrando ficheros y con la rutina que determina e informa al usuario sobre la existencia de interferencia entre dientes y de la relación de contacto de cada engranaje.

#### Plan de Futuro

En esta sección se indican algunos de los puntos débiles del programa en los que se está trabajando

- El intérprete de comandos sólo devuelve mensajes de error indicando que el comando introducido es incorrecto y la ayuda en línea es excesivamente escueta para un usuario nuevo.
- El programa no es tolerante a ciertos fallos y en ocasiones se termina la ejecución repentinamente. Este problema no es excesivamente serio pues en un uso normal es bastante robusto y en sucesivas versiones se ha ido reduciendo sustancialmente el número de fallos. Para evitar pérdidas de trabajo el programa almacena en un fichero (en el directorio /temp/tren\_v.v) con todas las órdenes introducidas. Por otra parte se puede minimizar el riesgo de este problema definiendo el sistema en un fichero externo que se cargue en una sesión interactiva.

- Se puede implementar un control mayor de errores.
- Posiblemente fuese deseable mayor independencia de la aplicación con la librería gráfica PGPLOT.

#### 4. APLICACIÓN: EJEMPLO PRÁCTICO

En este apartado se muestra un ejemplo de práctica para el trabajo individual de cada alumno. Alguno de los ejercicios están ilustrados con las figuras de la 1 a la 4, con imágenes sacadas del programa con el siguiente script:

```

1 0 e 50 0; # definición de rueda
ventana=-15 15 40 59 # define ventana visible
1; alfa=20 # ángulo de presión rueda 1
dibuja # muestra el dibujo
psd=alfa20.ps m # exporta a postscript
# monocromo
1; alfa=10 # cambia ángulo de presión
d # dibuja (forma corta)
psd=alfa10.ps m
1; alfa=30 # nuevo ángulo de presión
d # muestra el tren actual
psd=alfa30.ps m
alfa=20 # áng. presión por defecto
1 0 e 5 0; alfa=20 # define la rueda motriz
2 1 e 25 0 ; # ... y otra conducida
ventana=-8 10 -10 9
titulo='alfa=20dg, Z=6,25'
d # muestra el tren actual
psd=evol20.ps m
1 ; alfa=25
d
psd=evol25.ps m # modifica dedendum rueda 2
2; ka=.7
d
psd=evol25_cab_corta.ps m # genera tren compuesto
1 0 e 10 0 ;
2 1 e 20 0 ; ka=1
3 -2 e 12 0 ; color=red
4 3 e 22 15 ; v=-1
titulo='tren compuesto' # establece el título
e # encuadra la imagen
psd=tren_c m
# define el tren planetario
1 0 e 12 0 ; v=1 # sol
2 1 e 18 0 ; v=1 vl=c # satélites
3 1 e 18 120; v=1 vl=c
4 1 e 18 240; v=1 vl=c
5 2 i 48 0 ; # dentada interior
titulo='Engran. planetario' # establece el título
encuadra
psd=planetario.ps m
interactivo # control al teclado

```

#### Objetivo de la práctica

Esta práctica tiene como objetivo familiarizarse con distintos conceptos de la geometría del diente desarrollados en clase así como de la relación de transmisión, visualización del fenómeno de la interferencia, y la creación de sistemas simples de engranajes

#### Geometría del Diente

Se recuerda que muchos parámetros de la geometría del diente vienen dados por defecto, pero pueden redefinirse: (módulo=2, ángulo de presión=20°, addendum=m ( $k_a = 1$ ), dedendum=1.25m ( $k_d = 1,25$ ), espesor circular =  $0,95P_c/2$ )

#### Forma del diente / ángulo de presión:

1. Crear una rueda con bastantes dientes (50) y ampliar la ventana para ver claramente uno de ellos. Mostrar los círculos de base, primitivo exterior y de fondo.
2. Modificar el ángulo de presión ( $\alpha(^{\circ}) = 5, 14, 20, 25, 30, \dots$ ) incrementarlo hasta que ambas caras del diente se corten (aparezca una cola de pescado. Comentar que ha ocurrido y relacionarlo con el espesor circular del diente.

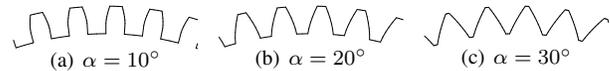


Figura 1: Efecto del ángulo de presión sobre la forma del diente para una rueda de 50 dientes de altura normal.

#### Círculos de fondo y de base / ángulo de presión:

1. Volver a establecer el ángulo de presión en  $\alpha = 20^{\circ}$
2. Crear una rueda de pocos dientes (5) y observar si parte del perfil no es de evolvente.
3. Aumentar el número de dientes y observar el perfil.
4. Prever (calcular) el mínimo número de dientes para que todo el perfil sea de evolvente y probarlo.
5. Modificar el ángulo de presión (establecerlo primero en  $\alpha = 10^{\circ}$  y después en  $\alpha = 30^{\circ}$ ) y volver a calcular y comprobar el perfil.

#### Perfil del diente: altura de la cabeza y del pie:

1. Volver a establecer el ángulo de presión en  $\alpha = 20^{\circ}$  y crear una rueda de 25 dientes.
2. Modificar los valores del addendum y dedendum y comentar las repercusiones que tiene sobre la forma del diente (si es de evolvente o no en el pie, y si puede o no desarrollarse toda la cabeza).

#### Engrane entre Dientes: Relación de Contacto

Volver a establecer el ángulo de presión en  $\alpha = 20^{\circ}$  y crear una rueda de 25 dientes (motriz) y otra de proporciones estándar y 60 dientes conducida de la anterior. Mostrar la línea de acción y la normal. Solicitar la información sobre la relación de contacto en cada modificación que se haga.

**Altura de la cabeza y del pie:** Modificar los valores del addendum y dedendum de la motriz y comentar las repercusiones que tiene sobre el engrane con la otra rueda.

**Ángulo de presión:** Sobre el engranaje original modificar el ángulo de presión de la rueda motriz (observar que automáticamente se modifica el del conducido para satisfacer las condiciones de engrane). Comentar la repercusión que tiene sobre el engrane con la otra rueda (estimación de la variación de la relación de contacto).

### Engrane entre Dientes: Socavación

Volver a establecer el ángulo de presión en  $\alpha = 20^\circ$  y crear una rueda de 6 dientes (motriz) y otra de proporciones estándar y 60 dientes conducida de la anterior, también de proporciones estándar.

1. Comprobar si hay socavación y dónde se produce. Solicitar la información sobre la interferencia en cada modificación que se haga.
2. Modificar sucesivamente los siguientes parámetros y anotar su efecto sobre la socavación.
  - a) Ángulo de presión y anotar su efecto sobre la socavación.
  - b) Altura de la cabeza de cada una de las ruedas.
  - c) Altura del pie de cada rueda.
  - d) Número de dientes de cada rueda

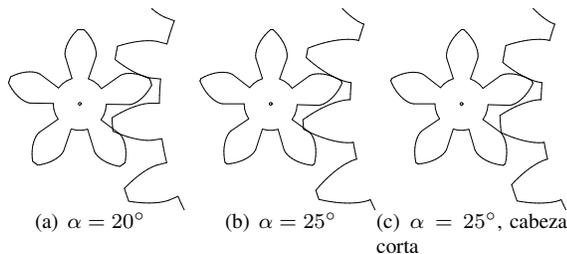


Figura 2: Efecto del ángulo de presión sobre la interferencia entre dientes. Obsérvese como la rueda grande en (b) reduce la socavación sobre el piñón al incrementar el ángulo de presión de 20 a 25 grados. En la imagen (c) además se ha recortado la cabeza del diente de la rueda, con lo que desaparece la interferencia, a costa de reducir la relación de contacto. Las ruedas trazadas tienen 5 y 25 dientes.

### Trenes de engranajes. Relación de Transmisión

#### Trenes ordinarios simples:

1. Construir un tren con tres ruedas dentadas y determinar la relación de transmisión conseguida.
2. Modificar el número de dientes de la rueda intermedia y comprobar nuevamente.
3. Modificar el número de dientes de las otras dos ruedas y comprobar de nuevo.

#### Trenes ordinarios compuestos:

1. Construir un tren con tres ejes y cuatro ruedas dentadas y determinar la relación final conseguida (el eje intermedio soporta dos ruedas, la segunda de las cuales es solidaria a la primera).
2. Modificar el número de dientes de cada rueda y comprobar el efecto sobre la relación final.

TREN-2.0 tren compuesto 2005 Mecánica, EUITA (UPM)

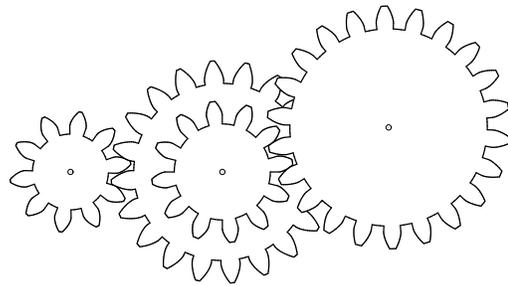


Figura 3: Tren compuesto con eje intermedio de dos ruedas. Esta imagen es un fichero exportado desde el programa.

#### Trenes planetarios:

1. Construir un tren planetario de un brazo, calculando previamente el número de dientes necesario para que el sol y la dentada interior sean coaxiales.
2. Conseguir una animación manteniendo la rueda interior fija.
3. Calcular la relación de transmisión brazo/sol y comprobarla.
4. Añadir otros dos brazos y comprobar qué reparto de dientes permite construir el mecanismo.

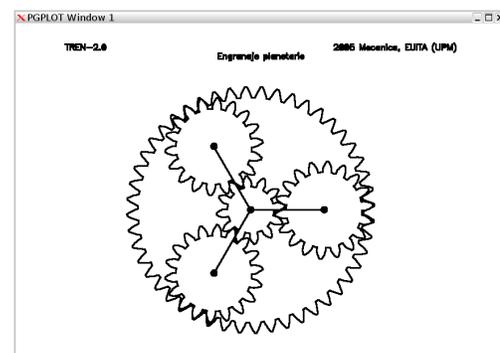


Figura 4: Tren planetario. La imagen es una captura directa de la ventana gráfica de la aplicación.

## 5. CONCLUSIONES

El programa se halla en evolución y está abierto a sugerencias de modificación. La versión actual se ha empleado en los tres últimos cursos académicos, dentro de un plan piloto de aplicación de metodologías activas. En el curso 2005/2006 se ha realizado, simultáneamente, un estudio del impacto de dichas metodologías activas en el aprendizaje [J.B.Ramiro, 2006]. De dicho estudio, en la parte que respecta a la utilización del programa TREN, pueden deducirse las siguientes conclusiones:

- Esta herramienta puesta a disposición del alumno en aulas de informática facilita un trabajo personalizado adaptado a cada alumno a la vez que fomenta el trabajo cooperativo puesto que unos alumnos se convierten de forma natural en tutores de otros menos iniciados.
- El trabajo personal incrementa el rendimiento del alumno al facilitar la visualización y manipulación de expresiones y conceptos relativamente complicados relacionados con la función evolvente.
- Potencia la motivación del alumno al comprobar que el tren por él diseñado cumple las condiciones impuestas inicialmente (como por ejemplo relaciones de transmisión, coaxialidad de ejes o viabilidad de un engranaje planetario). En caso contrario corregir el diseño inicial es un proceso de alto valor autodidacta.
- Así mismo hemos comprobado que ayuda al alumno a asimilar y visualizar conceptos que de otra forma serían meramente matemáticos, como son el ángulo de presión o el fenómeno de la interferencia. También permite al alumno profundizar y reflexionar sobre aspectos que de otro modo pasarían fácilmente desapercibidos. Por ejemplo la cabeza y el pie del diente son conceptos que el alumno, en primera instancia, tiende a concebir como simples partes de la altura del diente, sin comprender su diferente papel en el engrane.
- La valoración sobre los resultados académicos se ha publicado en [A.Alcázar, 2006], y arroja un resultado muy positivo. En el último curso se ha pasado una encuesta específica a los alumnos sobre este programa. Los resultados mostrados a continuación no pretenden ser estadísticamente rigurosos, sino dar una idea de su repercusión. La encuesta tenía las siguientes preguntas, y resultados, valorados sobre 10:
  - facilidad de uso del programa: 5.6
  - ayuda a asimilar conceptos geométricos: 8.0
  - ayuda a comprender el mecanismo: 7.7
  - la relación esfuerzo resultado es: 7.1
  - Recomendarías su uso: 8.0

- Como ejemplo concreto el programa ha ayudado a gran número de alumnos a comprender el funcionamiento de una caja de cambios tradicional de un automóvil.
- Es además una herramienta simple y flexible para el docente a la hora de generar imágenes y animaciones sencillas o de proponer trabajos y prácticas.

## 6. REFERENCIAS

1. J. B. Ramiro Díaz, Á. Alcázar de Velasco Rico, C. Fernández Jiménez. "Impacto de metodologías activas en el perfil académico y profesional del ingeniero técnico aeronáutico". Jornadas Nacionales de Intercambio de Experiencias Piloto de Implantación de Metodologías ECTS, Badajoz, 2006.
2. Gearpack. <http://sofengsvc.com> (1999)
3. J.E.Sighley y J.J,Uicker Jr. "Teoría de máquinas y mecanismos". *Ed. Mc. Graw Hill*. México (1998).
4. K-prof ([www.k-prof.com.ru](http://www.k-prof.com.ru)), 2002
5. Manual de g77 en la versión gcc-3.4.4 (y otras anteriores) de la distribución Debian 3.1r. (2004)
6. Robert L. Norton "Diseño de máquinas" *Ed. Prentice Hall*, México (1999)
7. T. J. Pearson, Librería y Manual de la librería PG-*PLOT* versión 5.2.2. (2001). [tjp@astro.caltech.edu](mailto:tjp@astro.caltech.edu), [www.astro.caltech.edu/~tjp/pgplot/](http://www.astro.caltech.edu/~tjp/pgplot/)
8. Thomas Williams, Collin Kelley, Manual de GNU-*PLOT* 4.0 (<http://gnuplot.sourceforge.net>), (2004)