

# Grupo de Simulación y Mecánica Computacional

## GSMC

## Proyecto CEMEF-UTN

12 de junio de 2011

## Introducción

**CEMEF-UTN (Cálculo de Estructuras con el Método de Elementos Finitos de la Universidad Tecnológica Nacional)** es un proyecto de desarrollo de 2 años de duración en su etapa inicial, impulsado por el Grupo de Simulación y Mecánica Computacional GSMC a cargo del Director Prof. Miguel A. Bavaro. Este proyecto se basa en la programación de algoritmos computacionales para conformar un código abierto (*open-source*), que permita resolver con fines académicos problemas físicos de la ingeniería a través de herramientas matemáticas de cálculo numérico, en particular el Método de los Elementos Finitos.

## Antecedentes preliminares

La idea de este plan de desarrollo surge a partir de un algoritmo desarrollado por Marcos C. Ruggeri (graduado de la carrera Ingeniería Aeronáutica en la UTN FRH) en el año 2008 para un trabajo práctico de la materia Estructuras Aeronáuticas III, en su momento dictada por el Prof. Carlos A. Carlssare y el Prof. Miguel A. Bavaro. Este programa, denominado originalmente MABS2D™ (*Matrix Analysis of Two Dimensional Bar Structures*), permitía resolver mediante el cálculo matricial de estructuras de barras cualquier sistema articulado en el plano, suponiendo válidas todas las hipótesis para el cálculo de estructuras articuladas. En su primera versión v1.0, solo era posible trabajar con estructuras hiperestáticas en el plano (ya que en el algoritmo no estaba todavía programado la funcionalidad de definir apoyos móviles), tomaba como datos de entrada plantillas de un archivo de Microsoft Excel® como interfaz gráfica y corría bajo la plataforma comercial MATLAB®.

Básicamente trabajaba discretizando el sistema continuo en elementos de barra de dos dimensiones y área constante; los efectos de flexión y corte no son considerados para la formulación. Realizaba un análisis estático lineal para calcular el campo unidimensional de tensiones y deformaciones sobre cada miembro de la estructura articulada, lo cual permitía determinar como una primera aproximación la respuesta del sistema frente a un estado de cargas y condiciones de contorno impuestas y dimensionar cada miembro de la estructura. Los datos de entrada eran importados desde un archivo de Microsoft Excel®, donde se encontraba toda la información relativa a la geometría y las propiedades físicas de la estructura: tabla de nodos, tabla de elementos, vector de cargas nodales exteriores y condiciones de contorno. Se calculaban todas las matrices de rigidez de cada elemento en coordenadas locales, y luego se transformaban a coordenadas globales para contemplar las rotaciones de cada barra. La tabla de conectividad era automáticamente generada para el proceso de ensamblaje y así obtener la matriz de rigidez completa de la estructura. Al aplicar las condiciones de vínculo se resolvía el sistema matricial reducido y se obtenía el vector de desplazamientos nodales y el vector de cargas nodales (externas y reacciones). Finalmente el programa volvía a la formulación elemental para averiguar las tensiones en cada elemento. Con estos resultados se generaba un gráfico de la estructura en la posición de equilibrio deformada con la distribución de tensiones (sigma11).

En su versión v1.1, se incorporó en agosto del 2009 la funcionalidad de trabajar con estructuras tanto hiperestáticas como isostáticas al programarse los grados de libertad prescriptos que pueden

elegirse según las condiciones de vínculo (apoyos móviles o fijos), a pedido de un estudiante de la carrera de Ingeniería Civil en la Universidad de Wisconsin-Milwaukee de Estados Unidos. Para marzo del 2010, en su nueva versión v2.1 el programa fue extendido a 3 dimensiones con la posibilidad de poder analizar estructuras reticuladas en el espacio. Entre las mejoras en esta versión MATLAB/MABS3D™, se agregó en la visualización de resultados la distribución del campo de tensiones con color y un factor de amplificación para poder ver bien el comportamiento de la estructura en la posición deformada durante la evolución del proceso de solicitación de las cargas.

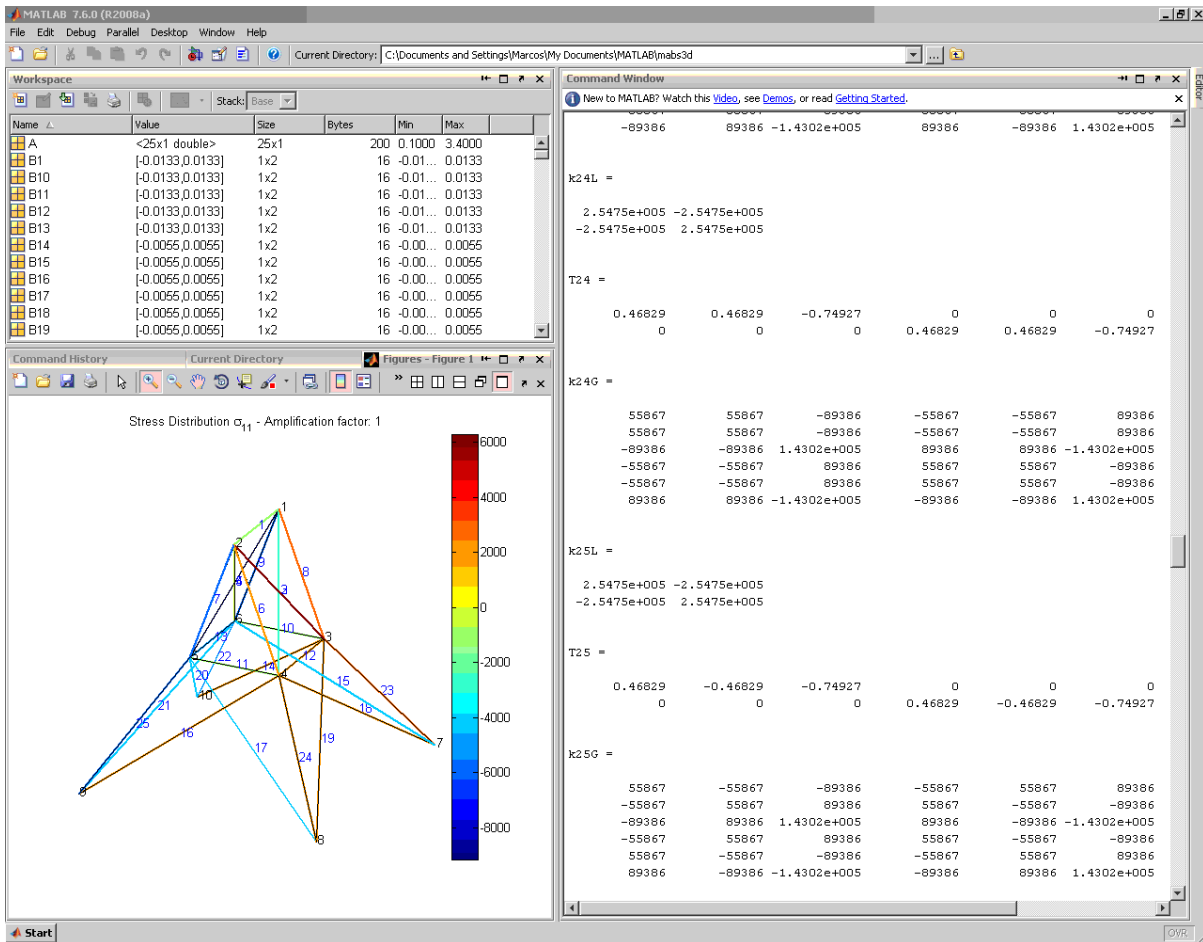
Paralelamente, en 2009 Sergio D. Lingeri (estudiante de la carrera Ingeniería Aeronáutica en la UTN FRH) realizó un programa para un trabajo práctico de la materia Estructuras Aeronáuticas III, el cual logra obtener la solución de un ejercicio en particular de cálculo dinámico a través del Método de Superposición Modal. Este programa resuelve el clásico problema de una barra en caída libre, la cual al impactar con un blanco rígido se excita y amplifica sus frecuencias naturales. Utilizando inicialmente una discretización de la estructura con tres elementos de barra y cuatro nodos, se logró obtener del análisis las cuatro frecuencias naturales y sus respectivos modos naturales de vibración del sistema.

Aplicando para este ejercicio en particular el Método de Superposición Modal (incorporado en el programa) fue posible obtener la respuesta temporal de cada uno de los nodos en un pequeño instante posterior al impacto, observando como la energía utilizada por el sistema para vibrar reduce la altura final del sistema luego de cada rebote.

Junto a los demás integrantes participa también el estudiante Andrés Vennera de Ingeniería Aeronáutica en la UTN FRH. Fue becario del programa UTN-DAAD 2008 en la Ruhr-Universität Bochum de Alemania, donde cursó materias relacionadas a Turbo-máquinas y Propulsión, a la vez que realizó una experiencia en simulación con métodos numéricos de capas límites laminares y turbulentas y su comparación con resultados teóricos. Posee experiencia en diversos lenguajes de programación en el área de bases de datos y en la implementación de microcontroladores PIC.

Más adelante se incorporó al grupo de Simulación y Mecánica Computacional Lucas Benitez Demmler estudiante de la carrera Ingeniería Aeronáutica en la UTN FRH. Posee conocimientos en lenguaje de programación C++ como así también cursos realizados en MATLAB.

Figura 1. MABS3D™ aplicado a un soporte de alta tensión corriendo bajo la plataforma comercial MATLAB® desarrollado por Marcos C. Ruggeri.



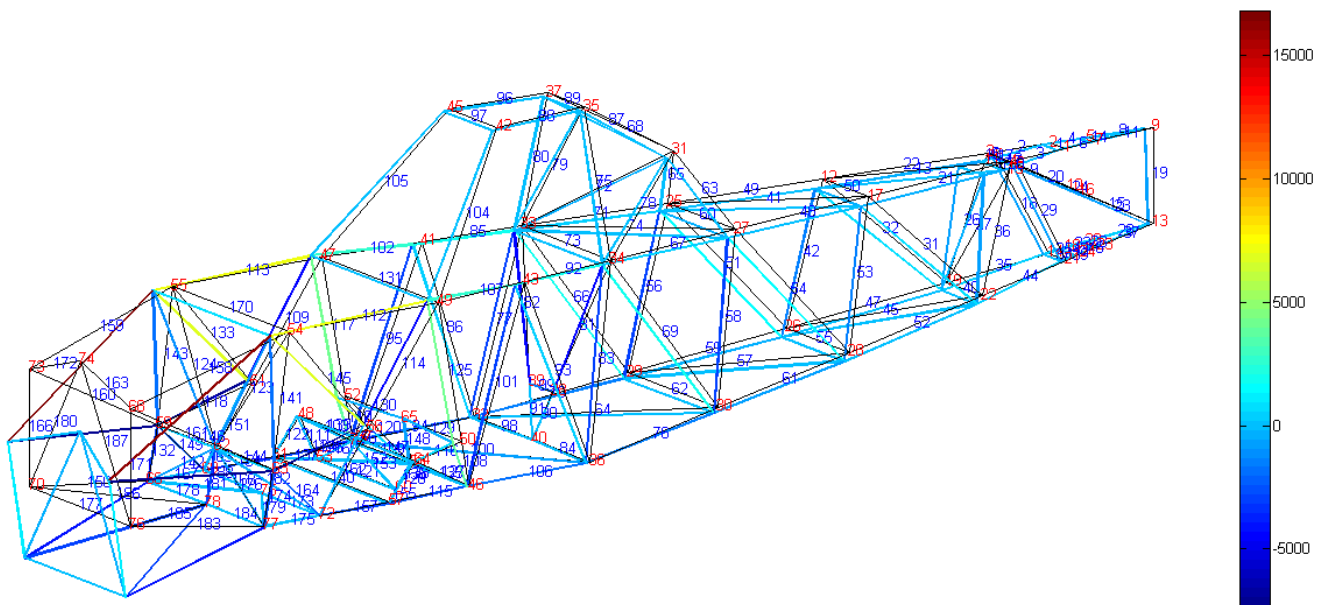
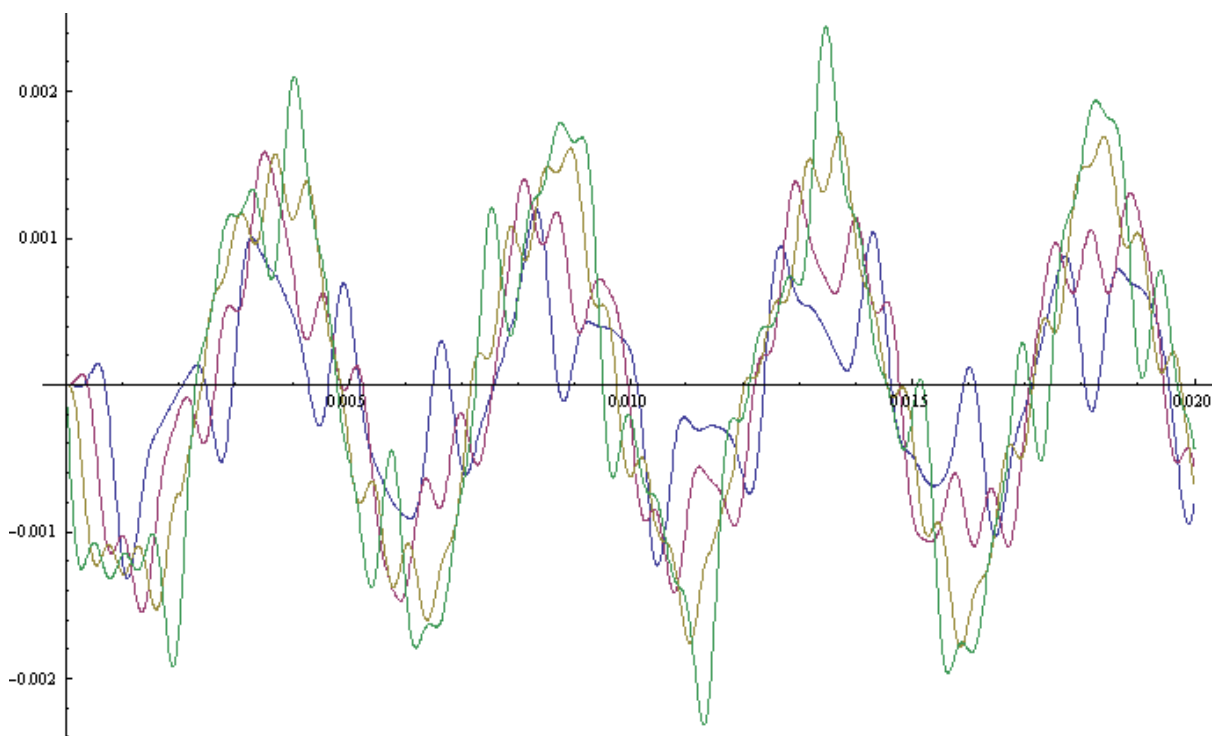


Figura 2. MABS3D™ aplicado al fuselaje reticular del Piper PA-25 “Pawnee”. Visualización de tensiones sigma11 con factor de amplificación x200.



**Figura 3.** Respuesta adimensional temporal de los grados de libertad activos en la modelo de elementos de barra con el algoritmo de cálculo dinámico desarrollado por Sergio D. Lingeri.

## Organización del GSMC



## Proyecto CEMEF-UTN

### 1. Equipo de trabajo



- Contactos

1. Marcos C. Ruggeri [mruggeri@frh.utn.edu.ar](mailto:mruggeri@frh.utn.edu.ar)
2. Lucas Benítez Demmler [luga\\_2003@hotmail.com](mailto:luga_2003@hotmail.com)
3. Andrés Vennera [andresvennera@gmail.com](mailto:andresvennera@gmail.com)
4. Sergio D. Lingeri [sdlingeri@hotmail.com](mailto:sdlingeri@hotmail.com)

## 2. Metodología de la investigación

El proyecto CEMEF-UTN tiene como fundamento teórico varias disciplinas de la Ingeniería, en particular la Ingeniería Aeronáutica, carrera de grado estudiada por todos los integrantes del proyecto. Entre las asignaturas relacionadas al proyecto se pueden mencionar Análisis Matemático I y II, Física I y II, Computación, Álgebra y Geometría Analítica, Estructuras Aeronáuticas I, II y III, Mecánica del Continuo, entre otras. Todas estas herramientas son útiles para el desarrollo de algoritmos computacionales, lo que requiere de conocimientos previos no sólo en programación, sino también en la física y matemática de lo que se quiere programar en forma sistemática, para así generalizarlo a la resolución de cualquier problema real, simple o complejo.

Otro pilar teórico fundamental serán los métodos de cálculo numérico, en especial el Método de los Elementos Finitos o el Método de las Diferencias Finitas para la discretización del sistema continuo en otro sistema discreto, con puntos designados adecuadamente para obtener la información de las variables de campo.

La metodología de la investigación se realizará programando algoritmos bajo la plataforma de cálculo numérico de libre distribución scilab® (*open-source*), que procesen la información de acuerdo a los conceptos teóricos estudiados. Esta plataforma a su vez contiene su propio lenguaje de programación de alto nivel orientada a cálculo numérico con sintaxis similar a MATLAB®. scilab® fue desarrollado a comienzos de 1990 por el Instituto Nacional de Investigación en Informática y Automatización de Francia (INRIA) y es distribuido actualmente bajo licencias GPL y CeCILL.

Los códigos deberán ser revisados, probados y verificados reiteradas veces a través de un procedimiento conocido como *debugging*. Se usarán varios problemas de ejemplo como aplicación directa para visualizar rápidamente los posibles errores en la programación.

Toda la información será contenida en archivos ejecutables desde cualquier computadora con un sistema operativo funcionado adecuadamente. Además, los algoritmos del CEMEF serán desarrollados con el fin de proveer un ambiente amigable y fácil de usar para estudiantes, docentes y profesionales de la ciencia e ingeniería que deseen experimentar sus conocimientos con fines académicos.



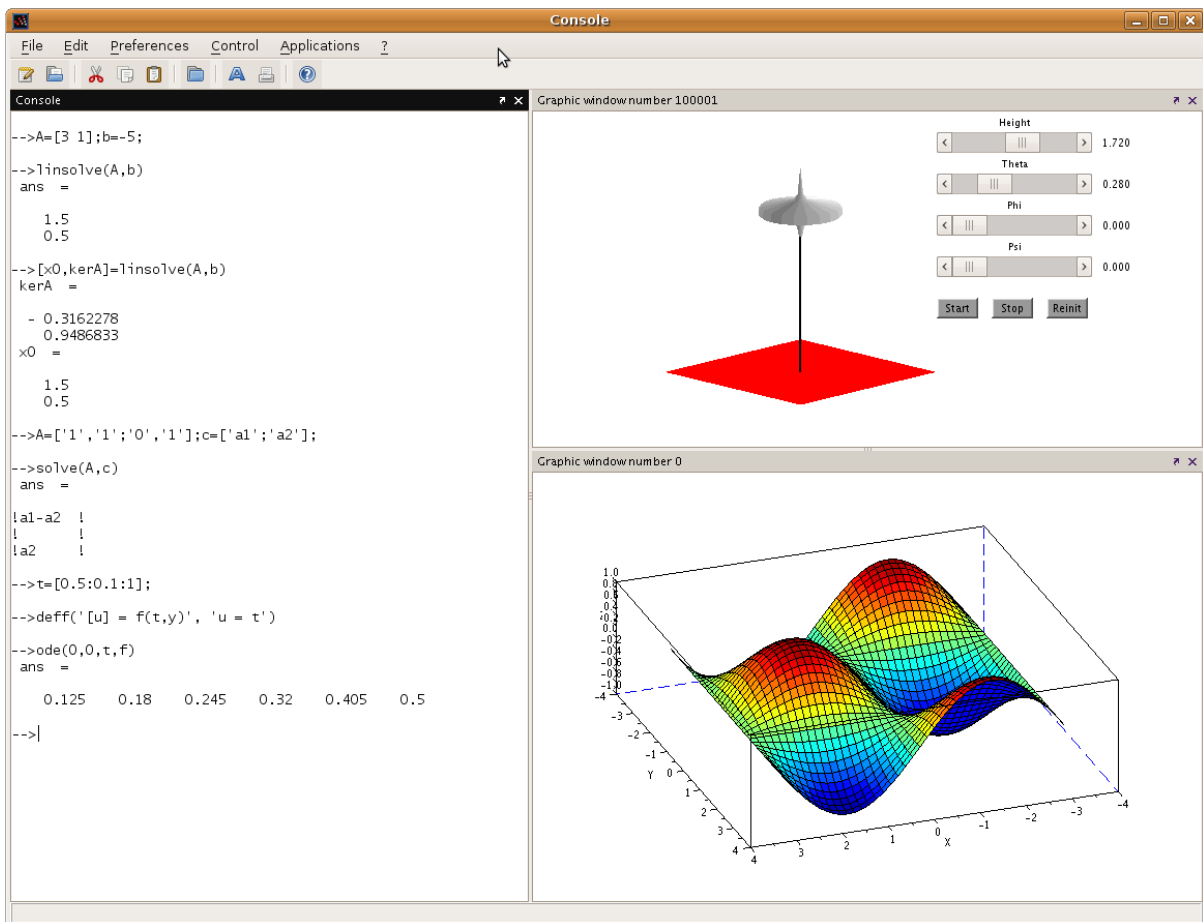


Figura 4. Interfaz gráfica de scilab® ejecutando líneas de comando.

### 3. Plan de trabajo

La etapa inicial del proyecto CEMEF-UTN tendrá una duración aproximada de 2 años, dividida brevemente en las siguientes fases de desarrollo:

- 1 – Familiarización de scilab®
  1. Interfaz
  2. Sintaxis
  
- 2 – Migración de plataforma para MABS3D™ de MATLAB® a scilab®
  - Traductor M2SCI
  - Interfaz
  - Sintaxis
  
- 3 – Programación de una interfaz gráfica de usuario GUI
  
- 4 – Optimización del algoritmo.
  3. *Debugging*
  4. Optimización, robustez y programación de errores
  
- 5 – Cálculo estático lineal: Formulación de otros elementos
  5. Vigas 3D
  6. Vigas-pórtico 3D
  
- 6 – Mayor funcionalidad en formulación de elementos
  7. Funciones de interpolación de segundo orden
  8. Area variable
  
- 7 – Optimización del algoritmo.
  9. *Debugging*
  10. Optimización, robustez y programación de errores
  
- 8 – Mayor funcionalidad en entradas de preprocesamiento
  11. Cargas distribuida de línea
  
- 9 – Mayor funcionalidad en resultados de postprocesamiento
  12. Visualización de condiciones de contorno *BC*
  13. Visualización de cargas
  
- 10 – Cálculo dinámico lineal: Formulación de elementos y método de resolución
  14. Barras 3D
  15. Vigas 3D
  16. Vigas-pórtico 3D
  17. Extracción de frecuencias naturales
  18. Método de Superposición Modal
  
- 11 – Optimización del algoritmo.
  19. *Debugging*
  20. Optimización, robustez y programación de errores
  
- 12 – Documentación

## Conclusiones preliminares

Los resultados esperados permitirán no solo aumentar el nivel de conocimiento académico científico y tecnológico de los miembros del proyecto, sino además poder volcar esos contenidos a la resolución de numerosas aplicaciones concretas para problemas de ingeniería de la vida real, como por ejemplo análisis de soportes de antenas, fuselajes reticulares tubulares de aviones ligeros, bancadas de motor, nacelas de motor, cohetes espaciales en el lanzamiento, entre otros.

Una vez desarrollada la etapa inicial de CEMEF-UTN, el programa será comparado con otros paquetes comerciales actuales (NASTRAN®, ABAQUS®, CATIA/ELFINI®, MATLAB/CALFEM®) así como también códigos libres (openFEM™, TrussAnalysis™), para verificar la validez de los resultados en el método de cálculo para la resolución de la ecuación matricial de equilibrio estático/dinámico de la estructura.

Creemos que este proyecto incentivará enormemente el uso masivo de códigos *open-source* en el ambiente universitario y la programación de algoritmos con fines netamente académicos, con el objetivo de fomentar el aprendizaje de los métodos numéricos mediante el uso de algoritmos sistemáticos que permitan pasar de un modelo físico complejo a uno matemático simplificado de una manera moderadamente sencilla, amigable y fácil de usar.

*Este proyecto es dedicado in memoriam al Prof. Carlos A. Carlassare, gran ejemplo de vida como profesional y como persona, quien como Director del GSMC impulsó desde el primer momento el desarrollo de un proyecto para fomentar el valor del conocimiento por sobre todas las cosas. A él se le agradece el haber enseñado y transmitido el espíritu de la honestidad para lograr formar una sociedad más digna, justa y solidaria.*

## Memorandum 13/12/10: Estado del Proyecto

Nivel: FASE II

Estado: Terminada

**Ejemplo: Cálculo estructural de tensiones unidimensionales aplicado al fuselaje reticular del PA-25 "Pawnee" implementando el elemento de barra lineal 3D (bar3d)**

- Líneas de comando

---

scilab-5.3.0-beta-4

Consortium Scilab (DIGITEO)

Copyright (c) 1989-2010 (INRIA)

Copyright (c) 1989-2007 (ENPC)

---

Startup execution:

```
loading initial environment
-->file="PA-25_Fuselaje_v24_scilab.xls";
Warning : redefining function: file . Use funcprot(0) to avoid this message
-->[M,nod,elem,E,A,L,dircos,nn,ne] = inpbar3d(file);
Warning : redefining function: file . Use funcprot(0) to avoid this message
---> PROBLEM SIZE:
! - NUMBER OF NODES: 78 !
! - NUMBER OF ELEMENTS: 188 !
-->[kL,B,T,kG]=stiffbar3d(E,A,L,dircos,ne);
-->[tc,K0]=assemb3d(elem,3,6,kG,nn,ne);
-->[K,P,cDOF,cDOFasc]=bc(M,K0);
-->[u0,u,P0]=SPDsolve(K0,K,P,cDOF,cDOFasc,nn);
ACTIVE STIFFNESS MATRIX OF STRUCTURE IS NOT SINGULAR. SPD VERIFICATION COMPLETED SUCCESSFULLY.
Warning :
matrix is close to singular or badly scaled. rcond = 6.0973D-19
computing least squares solution. (see lsq).
-->[ue,sigma,f]=stressbar3d(E,A,B,T,tc,u0,ne);
-->preplottruss3d(nod,elem,nn,ne,1)
```

13/12/10

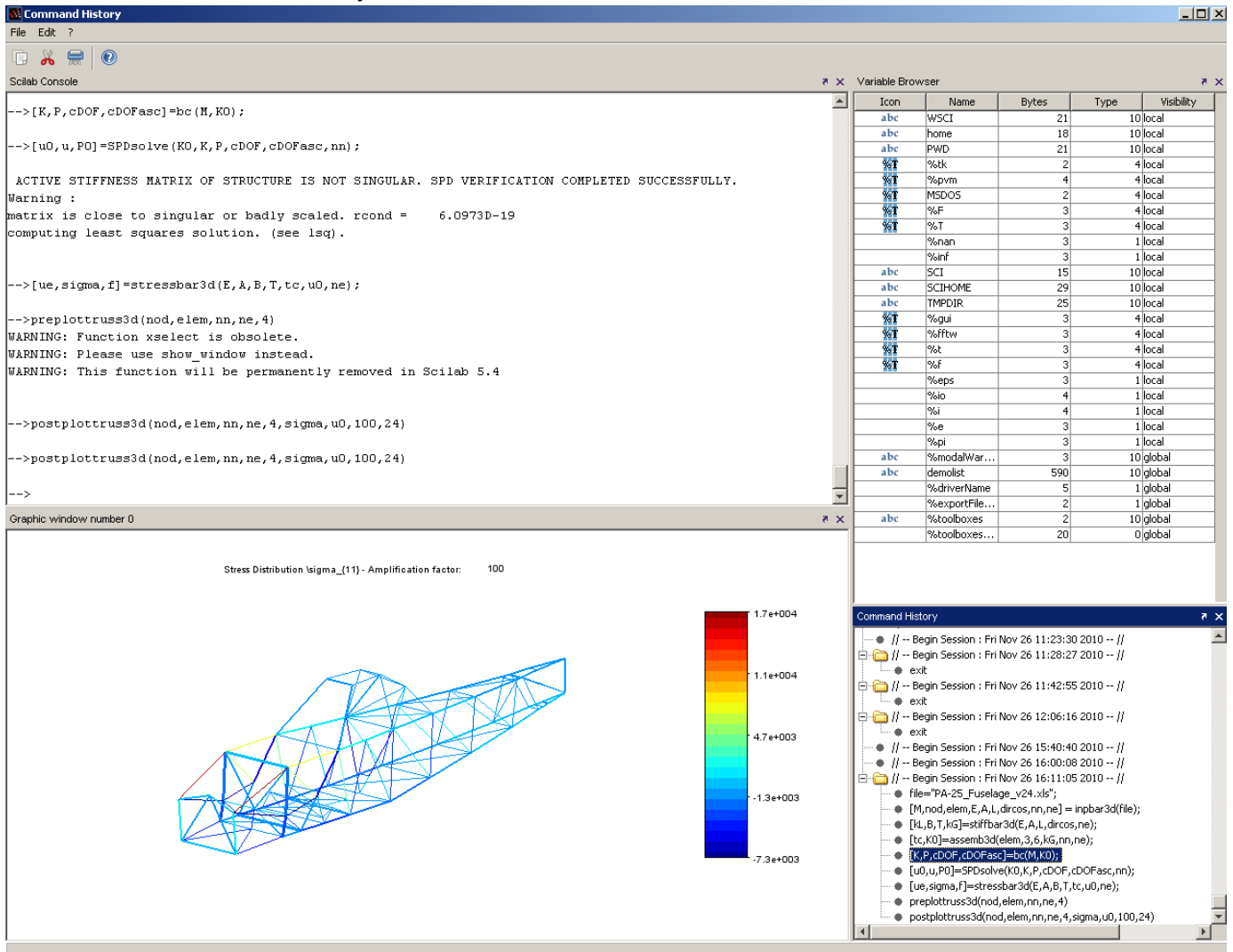
WARNING: Function xselect is obsolete.

WARNING: Please use show\_window instead.

WARNING: This function will be permanently removed in Scilab 5.4

-->postplottruss3d(nod,elem,nn,ne,1,sigma,u0,100,24)

• *Ambiente de trabajo*



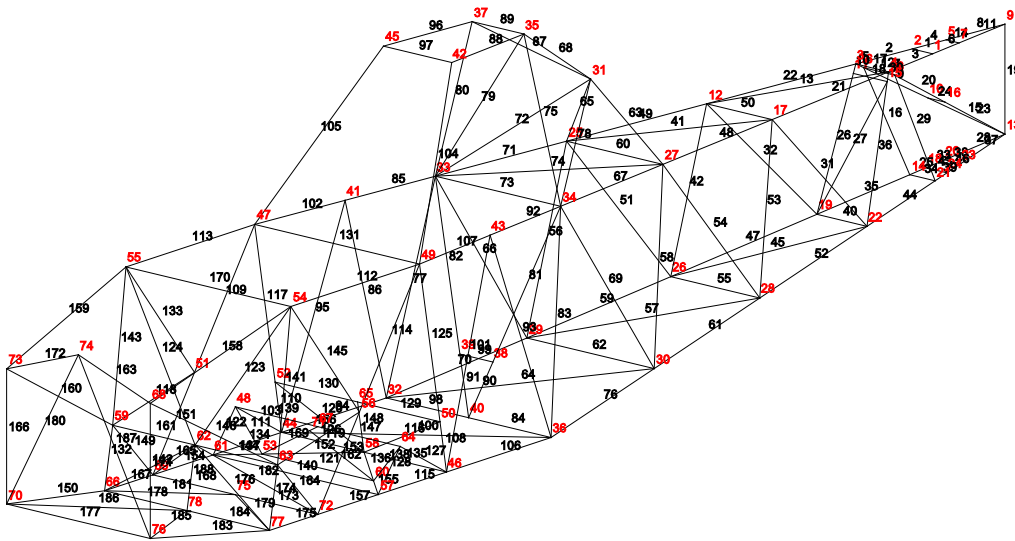
The screenshot displays the Scilab environment with the following components:

- Command History:** Shows the execution of commands:
 

```
-->[K,P,cDOF,cDOFasc]=bc(M,K0);
-->[u0,u,P0]=SPDsolve(K0,K,P,cDOF,cDOFasc,nn);
ACTIVE STIFFNESS MATRIX OF STRUCTURE IS NOT SINGULAR. SPD VERIFICATION COMPLETED SUCCESSFULLY.
Warning :
matrix is close to singular or badly scaled. rcond = 6.0973D-19
computing least squares solution. (see lsq).
-->[ue,sigma,f]=stressbar3d(E,A,B,T,tc,u0,ne);
-->preplottruss3d(nod,elem,nn,ne,4)
WARNING: Function xselect is obsolete.
WARNING: Please use show_window instead.
WARNING: This function will be permanently removed in Scilab 5.4
-->postplottruss3d(nod,elem,nn,ne,4,sigma,u0,100,24)
-->postplottruss3d(nod,elem,nn,ne,4,sigma,u0,100,24)
-->
```
- Variable Browser:** A table listing variables and their properties:
 

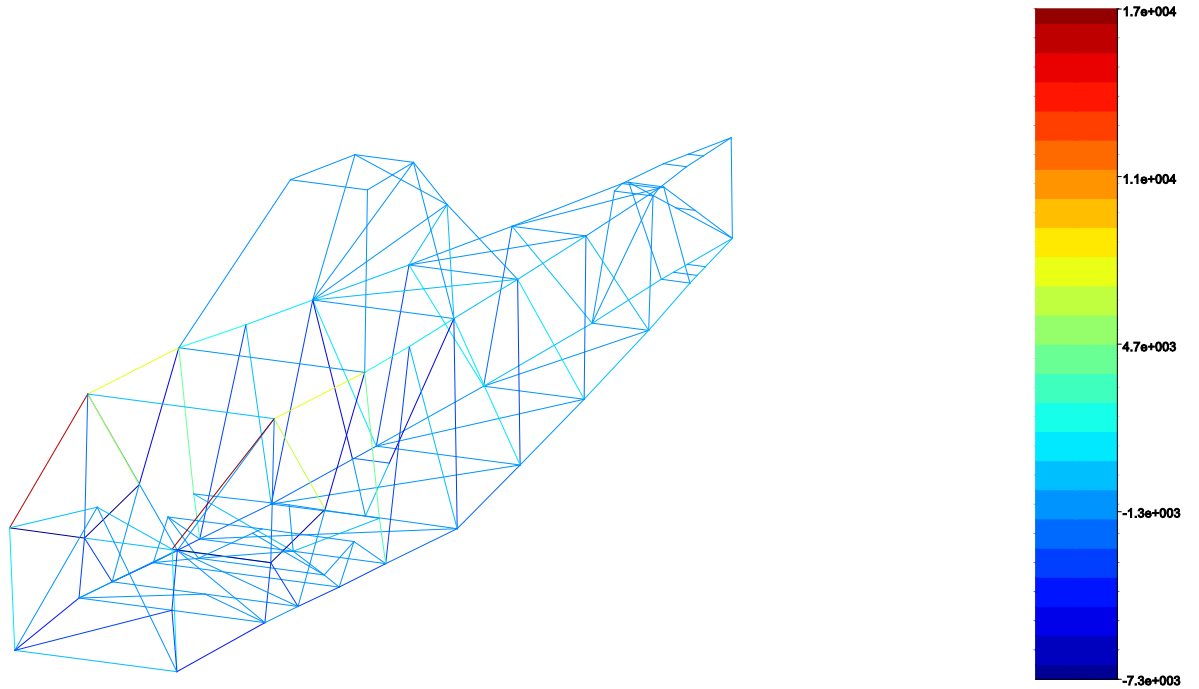
Icon	Name	Bytes	Type	Visibility
abc	WSCI	21	10	local
abc	home	18	10	local
abc	PWD	21	10	local
%T	%tk	2	4	local
%T	%pvm	4	4	local
%T	MSDOS	2	4	local
%T	%F	3	4	local
%T	%T	3	4	local
	%nan	3	1	local
	%inf	3	1	local
abc	SCI	15	10	local
abc	SCIHOME	29	10	local
abc	TMPOIR	25	10	local
%T	%gui	3	4	local
%T	%ftw	3	4	local
%T	%t	3	4	local
%T	%f	3	4	local
	%eps	3	1	local
	%io	4	1	local
	%i	4	1	local
	%e	3	1	local
	%pi	3	1	local
abc	%modalWar...	3	10	global
abc	demolist	590	10	global
	%driverName	5	1	global
	%exportFile...	2	1	global
abc	%toolboxes	2	10	global
	%toolboxes...	20	0	global
- Graphic window number 0:** Displays a 3D truss structure with a stress distribution plot. The plot is titled "Stress Distribution \sigma\_{11} - Amplification factor: 100". A color scale on the right indicates stress values ranging from -7.3e+003 (blue) to 1.7e+004 (red).
- Command History (bottom):** Shows a sequence of session starts and commands, including the execution of the stressbar3d and postplottruss3d functions.

• Malla de pre-procesamiento



- Visualización de resultados de post-procesamiento

Stress Distribution  $\sigma_{11}$  - Amplification factor: 100



13/12/10

## Flow Diagram of CEMEF-UTN v1.1-6

### Implementation of 3D Linear Bar Element (*bar3d*)

