

ESPECIALIZACIÓN EN ESTRUCTURAS ANÁLISIS ESTRUCTURAL AVANZADO

INTERACCIÓN ENTRE SAP2000 Y MATLAB

Michel Bolaños Guerrero,

Ing. Civil, Especialista en Estructuras,
Magister en Ingeniería – Énfasis en Ingeniería Civil,
Candidato a Doctor en Ingeniería – Énfasis en Mecánica de Sólidos

2023-B

Facultad de Ingeniería - Especialización en Estructuras

<https://michel.udenar.edu.co/> - michel@udenar.edu.co

Universidad de Nariño



INTERACCIÓN SAP Y MATLAB

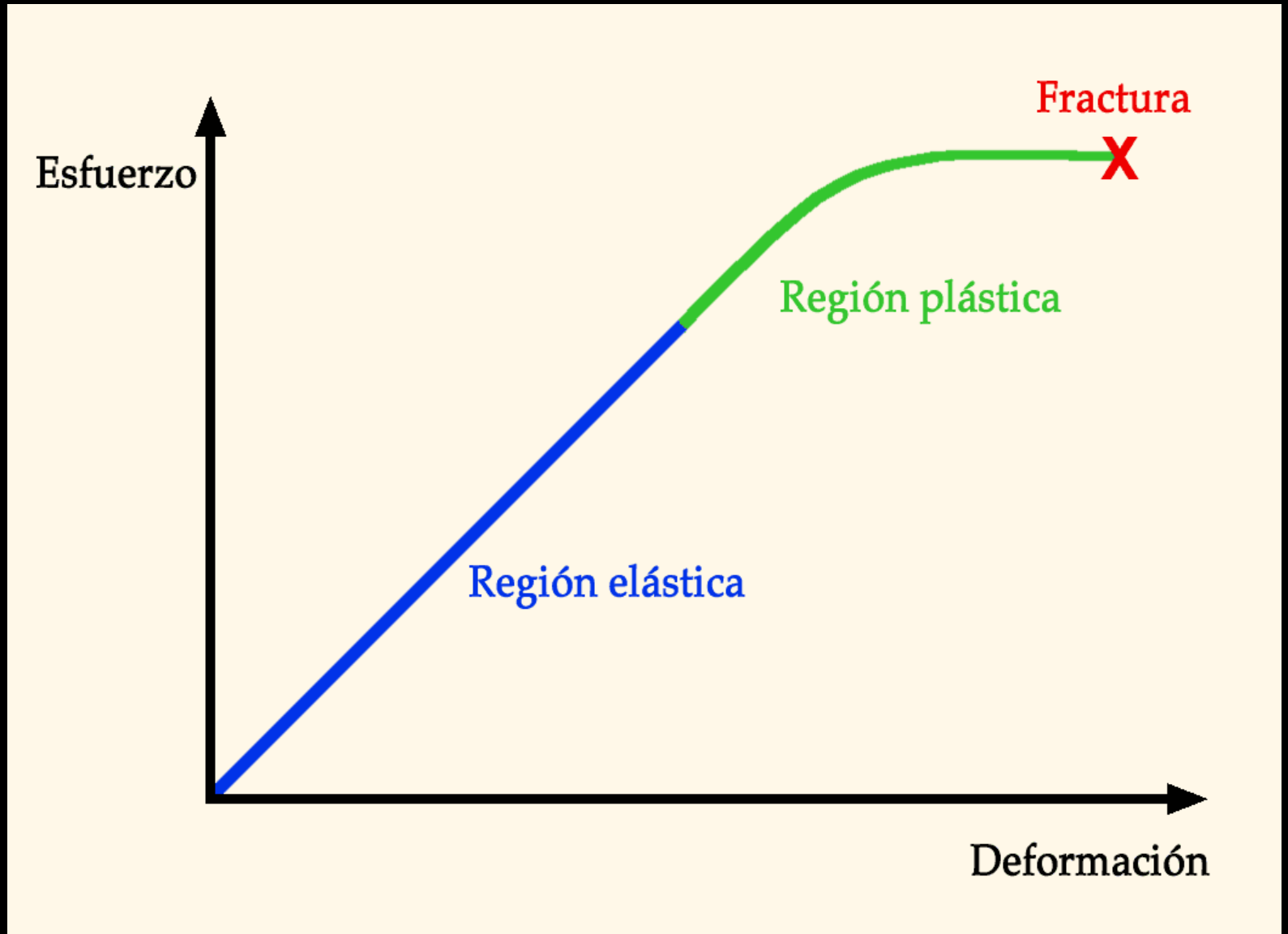
1. Generalidades.
2. OAPI.
3. Ajuste de modelos.
4. Algoritmos.
5. Maximización o minimización.



1. GENERALIDADES



1. GENERALIDADES



1. GENERALIDADES



¿Porqué?



Universidad de Nariño - Facultad de Ingeniería - Especialización en Estructuras

[ini](#) [1](#) [2](#) [3](#) [4](#) [5](#) [fin](#)

1. GENERALIDADES



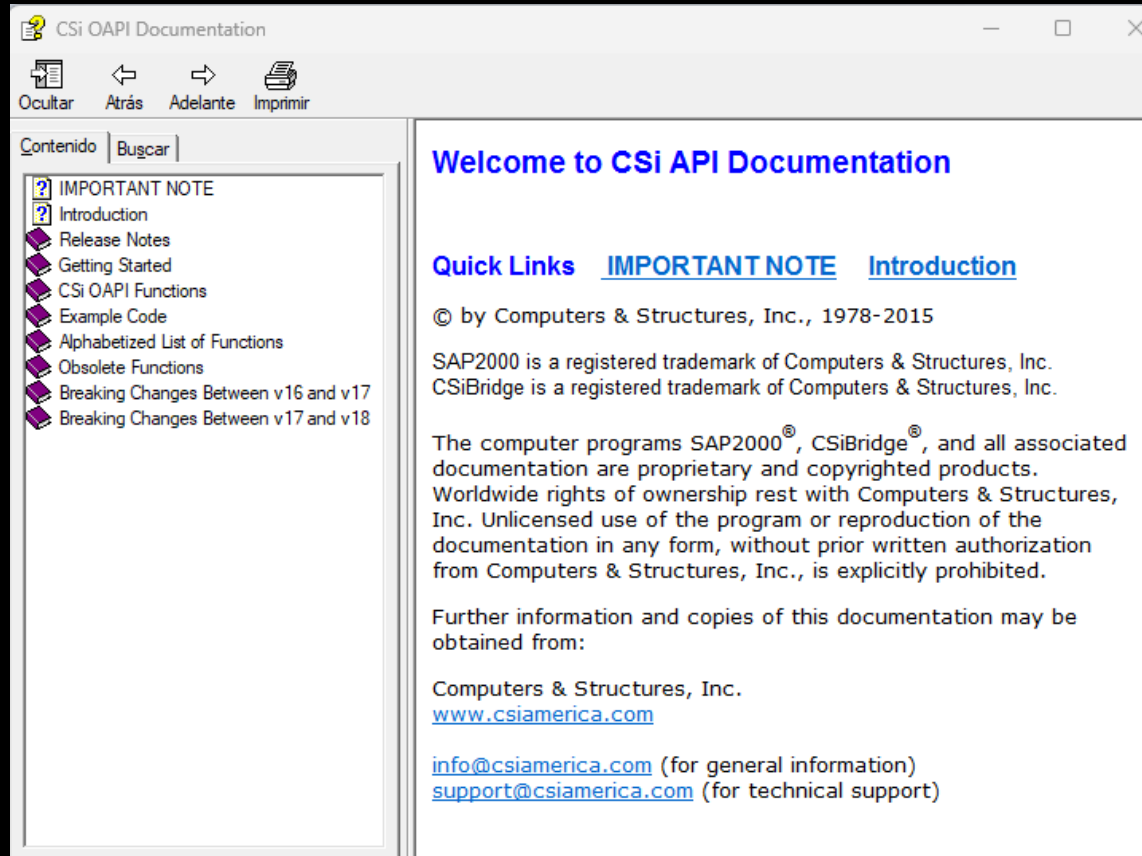
Minimización se secciones



2. OAPI

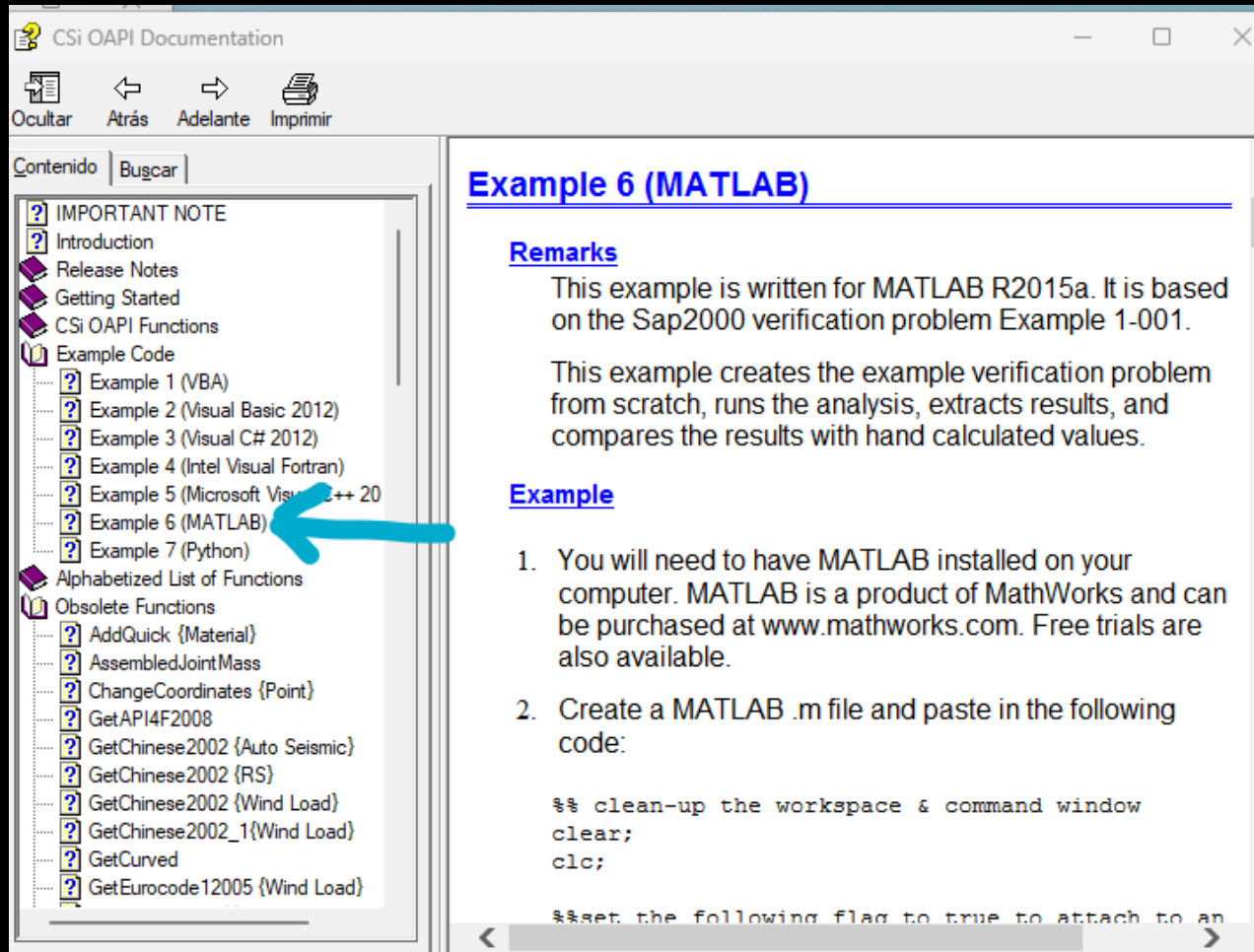
CSI_OAPI_Documentation.chm

Open Application Programming Interface (OAPI)



2. OAPI

CSI_OAPI_Documentation.chm



The screenshot shows a window titled "CSI OAPI Documentation" with a navigation pane on the left and a main content area on the right. The navigation pane lists various sections, with "Example 6 (MATLAB)" highlighted by a blue arrow. The main content area displays the text for Example 6, including a "Remarks" section and an "Example" section with a list of steps and code snippets.

Example 6 (MATLAB)

Remarks

This example is written for MATLAB R2015a. It is based on the Sap2000 verification problem Example 1-001.

This example creates the example verification problem from scratch, runs the analysis, extracts results, and compares the results with hand calculated values.

Example

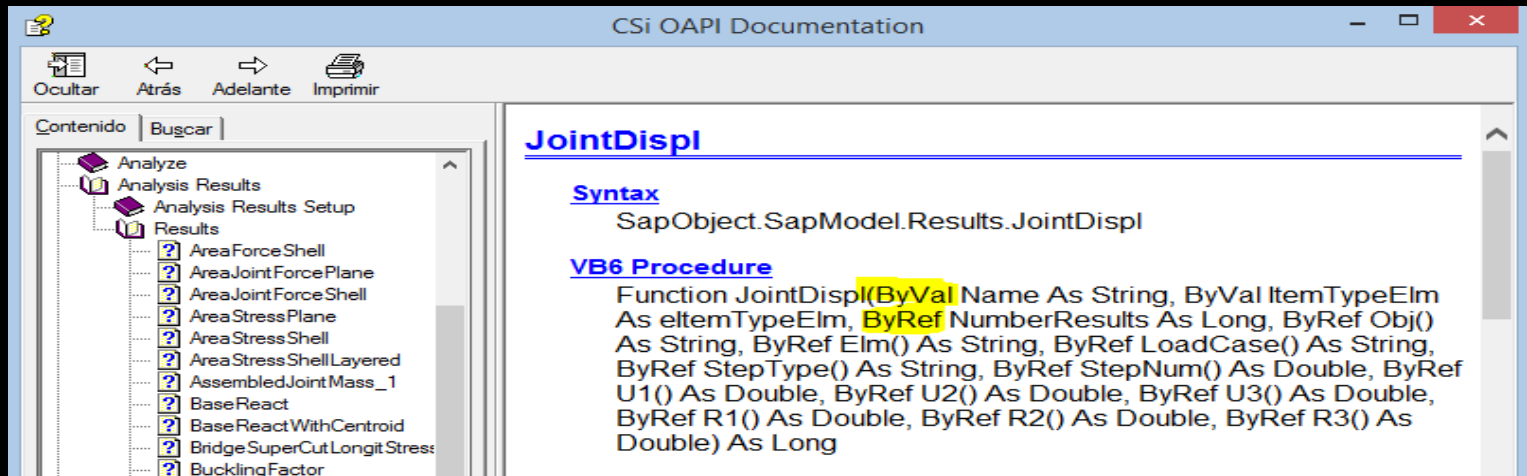
1. You will need to have MATLAB installed on your computer. MATLAB is a product of MathWorks and can be purchased at www.mathworks.com. Free trials are also available.
2. Create a MATLAB .m file and paste in the following code:

```
%% clean-up the workspace & command window
clear;
clc;

%%set the following flag to true to attach to an
```

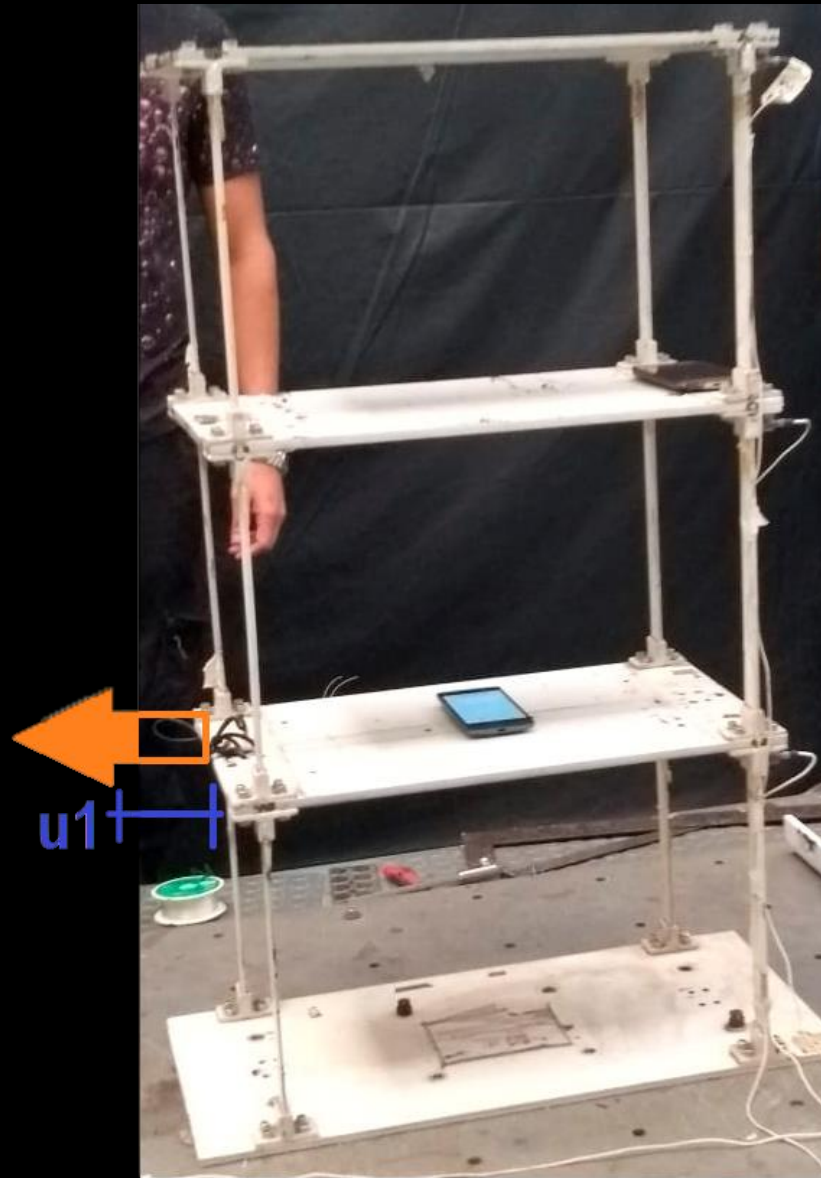

2. OAPI

CSI_OAPI_Documentation.chm

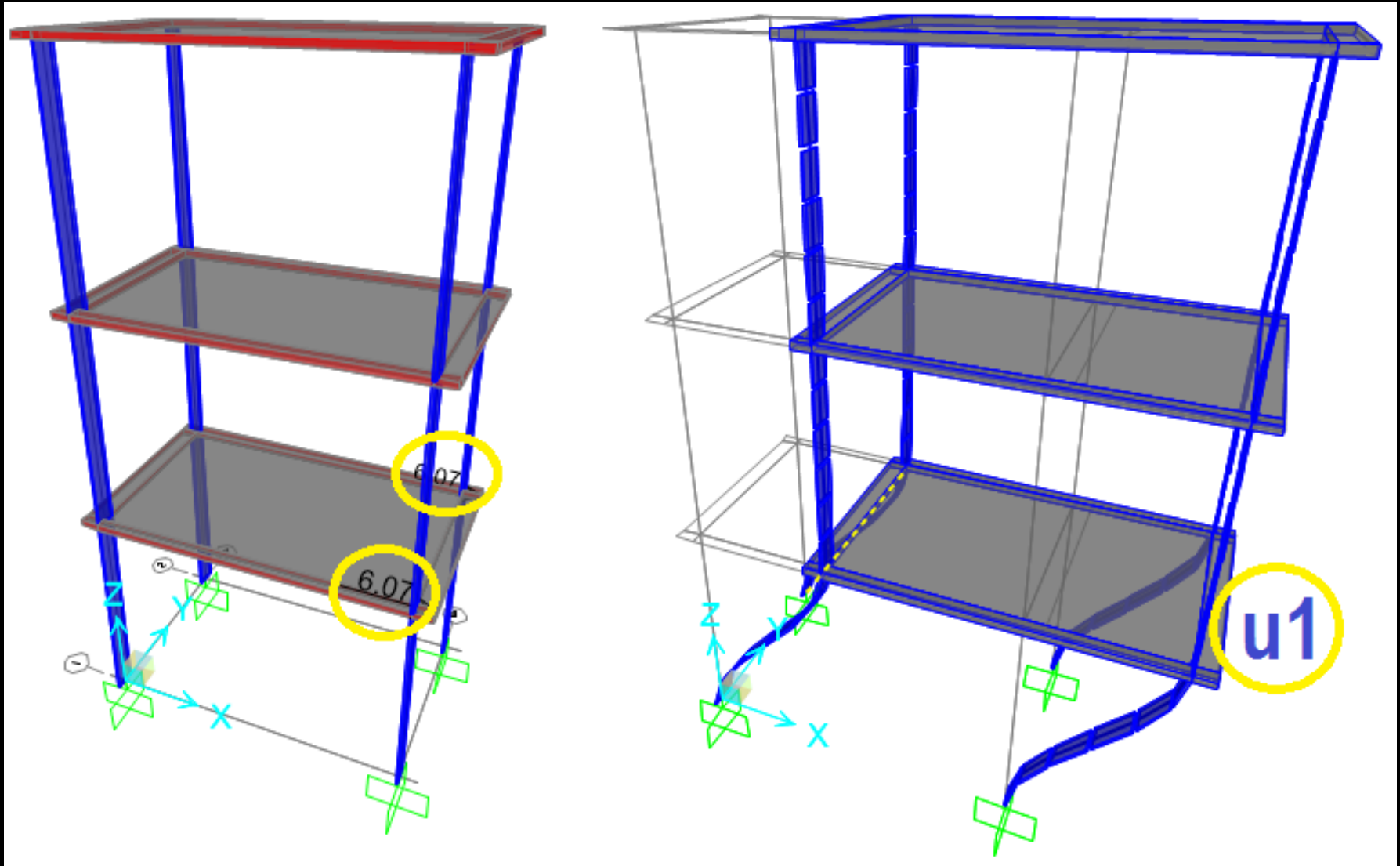


```
[ret, NumberResults, Obj, Elm, ACase, StepType, StepNum, ...  
U1, U2, U3, R1, R2, R3] = ...  
Results.JointDispl(nud, ...  
    GroupElm, ...  
    NumberResults, ...  
    Obj, ...  
    Elm, ...  
    LoadCase, ...  
    StepType, ...  
    StepNum, ...  
    U1, U2, U3, R1, R2, R3);
```

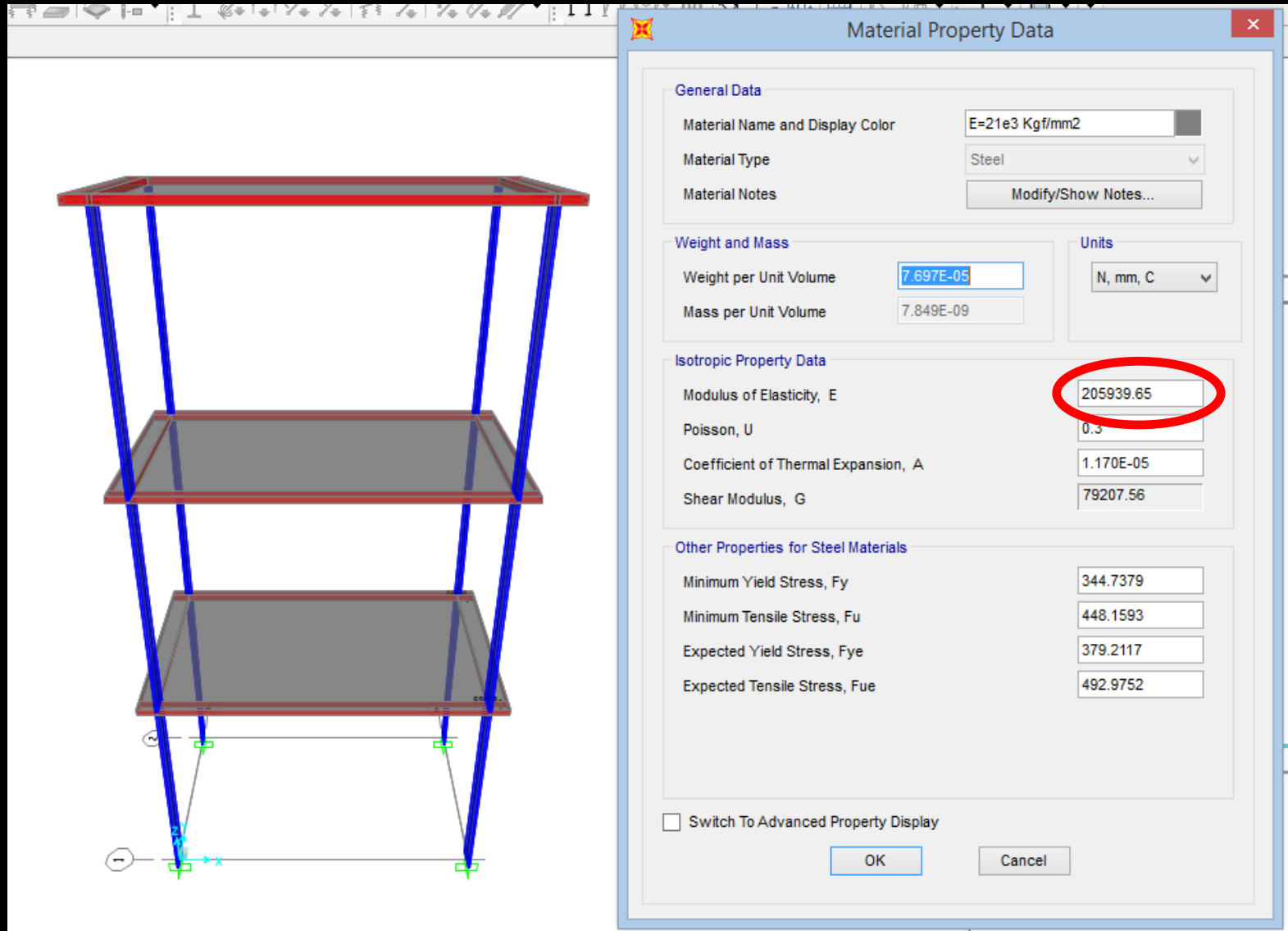
3. AJUSTE DE MODELOS



3. AJUSTE DE MODELOS



3. AJUSTE DE MODELOS



The image displays a 3D model of a three-tiered shelf structure with blue legs and red shelves. A 'Material Property Data' dialog box is open on the right side of the screen. The dialog box contains the following information:

- General Data:**
 - Material Name and Display Color: E=21e3 Kgf/mm2
 - Material Type: Steel
 - Material Notes: Modify/Show Notes...
- Weight and Mass:**
 - Weight per Unit Volume: 7.697E-05
 - Mass per Unit Volume: 7.849E-09
 - Units: N, mm, C
- Isotropic Property Data:**
 - Modulus of Elasticity, E: 205939.65 (highlighted with a red circle)
 - Poisson, U: 0.3
 - Coefficient of Thermal Expansion, A: 1.170E-05
 - Shear Modulus, G: 79207.56
- Other Properties for Steel Materials:**
 - Minimum Yield Stress, Fy: 344.7379
 - Minimum Tensile Stress, Fu: 448.1593
 - Expected Yield Stress, Fye: 379.2117
 - Expected Tensile Stress, Fue: 492.9752

At the bottom of the dialog box, there is a checkbox labeled 'Switch To Advanced Property Display' which is currently unchecked. The 'OK' and 'Cancel' buttons are also visible.

3. AJUSTE DE MODELOS

```
%% object SAP
```

```
pathDir='C:\Program Files\Computers and Structures\SAP2000 19\';  
ProgramSap=strcat(pathDir, 'SAP2000.exe');  
APIDLL=strcat(pathDir, 'SAP2000v19.dll');
```

```
a = NET.addAssembly(APIDLL); % Microsoft® .NET Framework  
helper = SAP2000v19.Helper;  
helper = NET.explicitCast(helper, 'SAP2000v19.cHelper');
```

```
%creating an instance of the Sap2000 object
```

```
SapObject = helper.CreateObject(ProgramSap);  
SapObject = NET.explicitCast(SapObject, 'SAP2000v19.cOAPI');  
helper = 0;
```

```
SapModel = NET.explicitCast(SapObject.SapModel, 'SAP2000v19.cSapModel');  
File = NET.explicitCast(SapModel.File, 'SAP2000v19.cFile');  
PropFrame = NET.explicitCast(SapModel.PropFrame, 'SAP2000v19.cPropFrame');  
PropMaterial =  
NET.explicitCast(SapModel.PropMaterial, 'SAP2000v19.cPropMaterial');  
Analyze = NET.explicitCast(SapModel.Analyze, 'SAP2000v19.cAnalyze');  
Results = NET.explicitCast(SapModel.Results, 'SAP2000v19.cAnalysisResults');  
ResultsSetup =  
NET.explicitCast(Results.Setup, 'SAP2000v19.cAnalysisResultsSetup');
```



3. AJUSTE DE MODELOS

```
%% start of application
SapObject.ApplicationStart;
ret = SapModel.InitializeNewModel;

%% opening data file

% Example of task 6
pathModel='D:\Google
Drive\DOCTORADO\Asignaturas\2\Dinámica Estructural\Ajuste
de modelos\Modelo SAP2000\';
modelName=strcat(pathModel, 'Tarea6.sdb');

ret = File.OpenFile(modelName);
ret =
SapModel.SetPresentUnits(SAP2000v19.eUnits.Ton_mm_C); %
Unidades
```



3. AJUSTE DE MODELOS

```
%% Ciclo en Sap2000

Ec = 9.5; %Tnf/mm2
U1_obj=2.4; %measured in laboratory 2.4mm. It's the target in task 6
Dif=+inf;
maxSteps=50;
Dif_obj=U1_obj*.01; %1% of objective-value
step=1;

mod=Ec;
while Dif>Dif_obj && step<=maxSteps
    % Desbloquea el modelo
    ret = SapModel.SetModelIsLocked(false);
    ret=PropMaterial.SetMPIsotropic('E=21e3 Kgf/mm2', mod, 0.3, 9.900E-06);

    % Corre modelo
    ret=Analyze.RunAnalysis;
    % Selecciona el CASE del cual desea obtener los datos
    ret = ResultsSetup.DeselectAllCasesAndCombosForOutput;
    ret = ResultsSetup.SetCaseSelectedForOutput('F-rigidez');
    nud='248';
    NumberResults = 0;
    GroupElm=SAP2000v19.eItemTypeElm.ObjectElm;
    Obj = {' '}; Elm = {' '};
    LoadCase = {' '};
    StepType = {' '};
    StepNum = 0;
    U1 = 0;    U2 = 0;    U3 = 0;    R1 = 0;    R2 = 0;    R3 = 0;

    [ret, NumberResults, Obj, Elm, ACase, StepType, StepNum, ...
    U1, U2, U3, R1, R2, R3] = ...
    Results.JointDispl(nud, GroupElm, NumberResults, Obj, Elm, LoadCase, StepType, StepNum, U1, U2, U3, R1, R2, R3);
    Dif=U1(1)-U1_obj;
    if abs(Dif)>Dif_obj
        mod=mod+mod*.5*Dif/U1_obj;
    end
    disp(strcat('step ', num2str(step), ' u1=', num2str(U1(1)), ' dif=', num2str(Dif)))
    step=step+1;
end
```



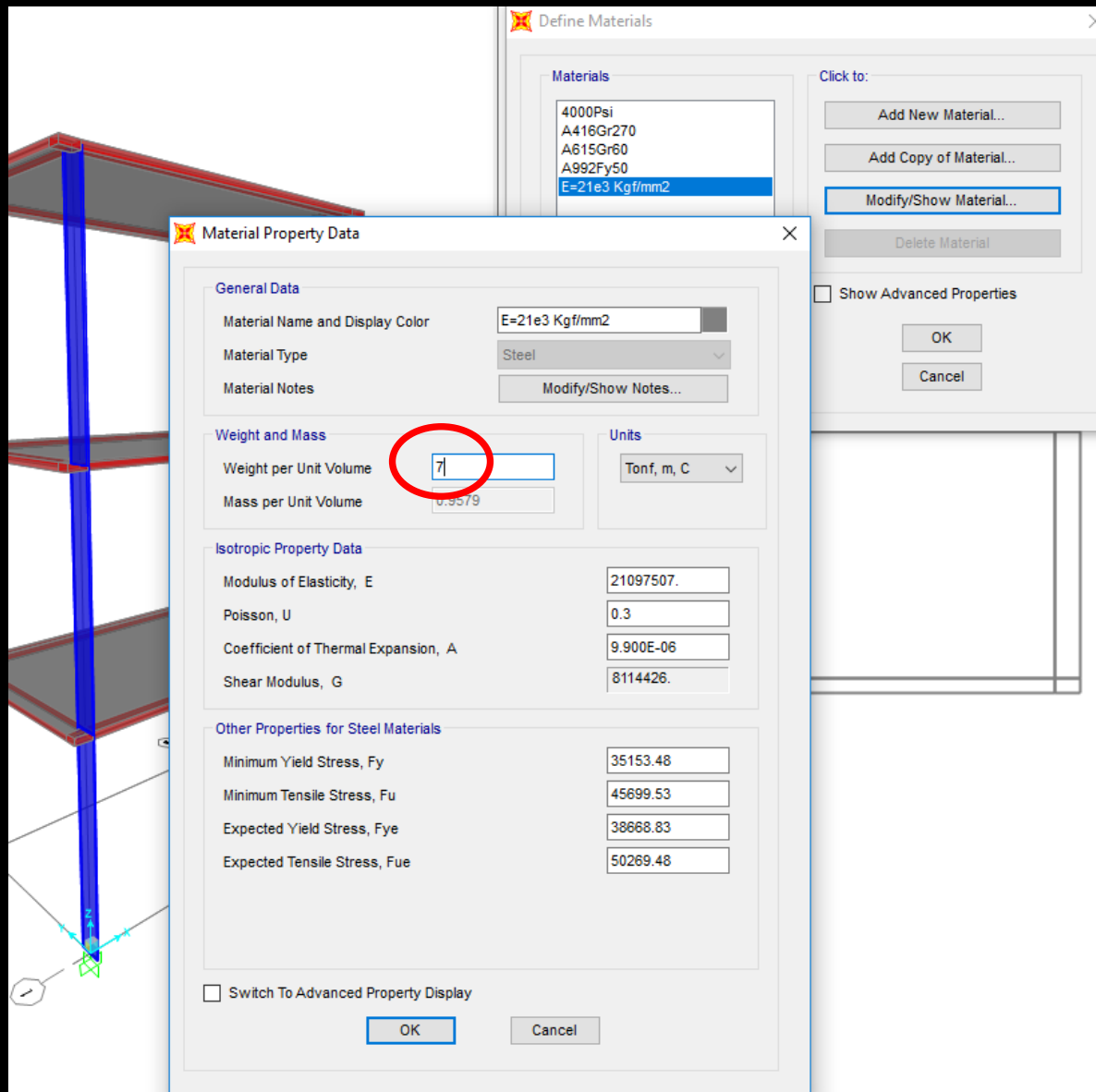
3. AJUSTE DE MODELOS

```
step:1 E=15.3444 u1=5.353 dif=2.953
step:2 E=18.2666 u1=3.3141 dif=0.91412
step:3 E=19.7277 u1=2.7839 dif=0.38394
step:4 E=20.4583 u1=2.5778 dif=0.17775
step:5 E=20.8236 u1=2.4857 dif=0.085702
step:6 E=21.0062 u1=2.4421 dif=0.0421
step:7 E=21.0975 u1=2.4209 dif=0.020867
step:8 E=21.0975 u1=2.4104 dif=0.010388
Ec (Tonf/mm2) = 21.0975
```

```
%% Best way to close SAP2000
%set the objects to Nothing
%at the end of your program ALWAYS terminate
the objects in this manner
ret = SapObject.ApplicationExit(false());
SapModel = 0;
SapObject = 0;
```



3. AJUSTE DE MODELOS



3. AJUSTE DE MODELOS

```
RoW=0;RoM=0;
[ret,RoW,RoM]=PropMaterial.GetWeightAndMass('E=21e3 Kgf/mm2',RoW,RoM);
%frequency 3
f1_obj=14.5; %measured in laboratory Hz. It's the target in task 6

Dif=+inf;
maxSteps=50;
Dif_obj=max(f1_obj)*.01/100; %0.1% of objective-value
step=1;

mod=RoW;
while Dif>Dif_obj && step<=maxSteps
    % Desbloquea el modelo
    ret = SapModel.SetModelIsLocked(false);

    ret=PropMaterial.SetWeightAndMass('E=21e3 Kgf/mm2',... %name
                                      1,... %option weight (2=mass)
                                      mod); %value of weight

    % Corre modelo
    ret=Analyze.RunAnalysis;
    % Selecciona el CASE del cual desea obtener los datos
    ret = ResultsSetup.DeselectAllCasesAndCombosForOutput;
```



3. AJUSTE DE MODELOS

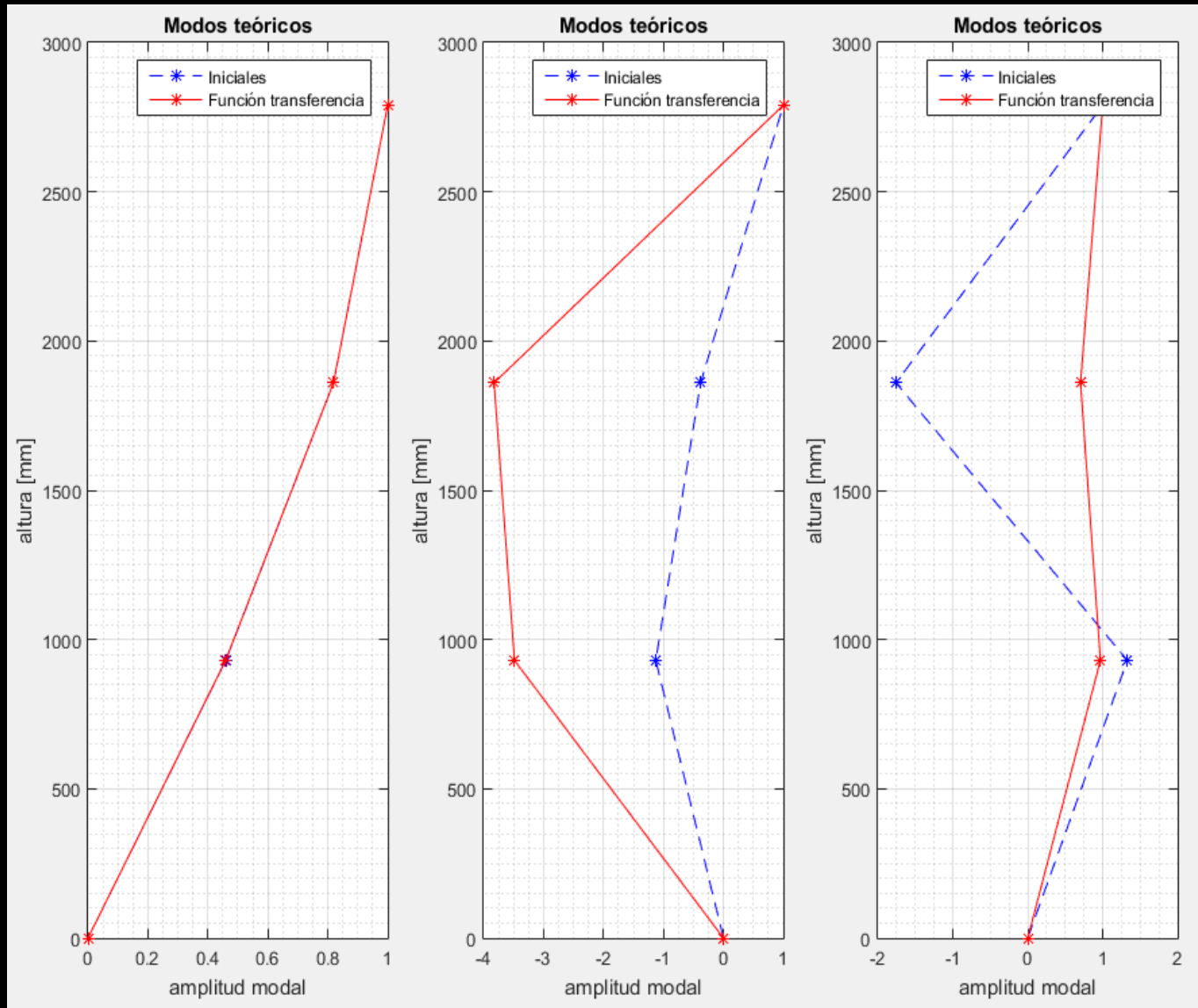
```
ret = ResultsSetup.SetCaseSelectedForOutput('MODAL');
NumberResults = 0;
GroupElm=SAP2000v19.eItemTypeElm.ObjectElm;
Obj = {' '}; Elm = {' '}; LoadCase = {' '}; StepType = {' '}; StepNum = 0;
Period=0; Frequency=0; CircFreq=0; EigenValue=0;

[ret, NumberResults, LoadCase, StepType, StepNum, ...
    Period, Frequency, CircFreq, EigenValue] = ...
Results.ModalPeriod(NumberResults, LoadCase, StepType, StepNum, ...
    Period, Frequency, CircFreq, EigenValue);
F=[Frequency(1) Frequency(2) Frequency(3)];
Dif=max(F(3)-f1_obj);
disp(strcat('step:', num2str(step), ' W=', num2str(mod), ' F=', num2str(F), ...
    ' dif=', num2str(Dif)))
if abs(Dif)>Dif_obj
    mod=mod+mod*.5*Dif/max(f1_obj);
end
step=step+1;
end
```

3. AJUSTE DE MODELOS

step:1	W=7	F=4.2327	11.6297	16.2975	dif=1.7975
step:2	W=7.4339	F=4.12829	11.3489	15.9218	dif=1.4218
step:3	W=7.7983	F=4.04631	11.1283	15.6257	dif=1.1257
step:4	W=8.101	F=3.9818	10.9545	15.3921	dif=0.89208
step:5	W=8.3502	F=3.93095	10.8175	15.2075	dif=0.7075
step:30	W=9.3923	F=3.73758	10.2958	14.5023	dif=0.0023211
step:31	W=9.3931	F=3.73745	10.2955	14.5018	dif=0.0018475
step:32	W=9.3937	F=3.73735	10.2952	14.5015	dif=0.0014706
step:33	W=9.3942	F=3.73727	10.295	14.5012	dif=0.0011705

3. AJUSTE DE MODELOS

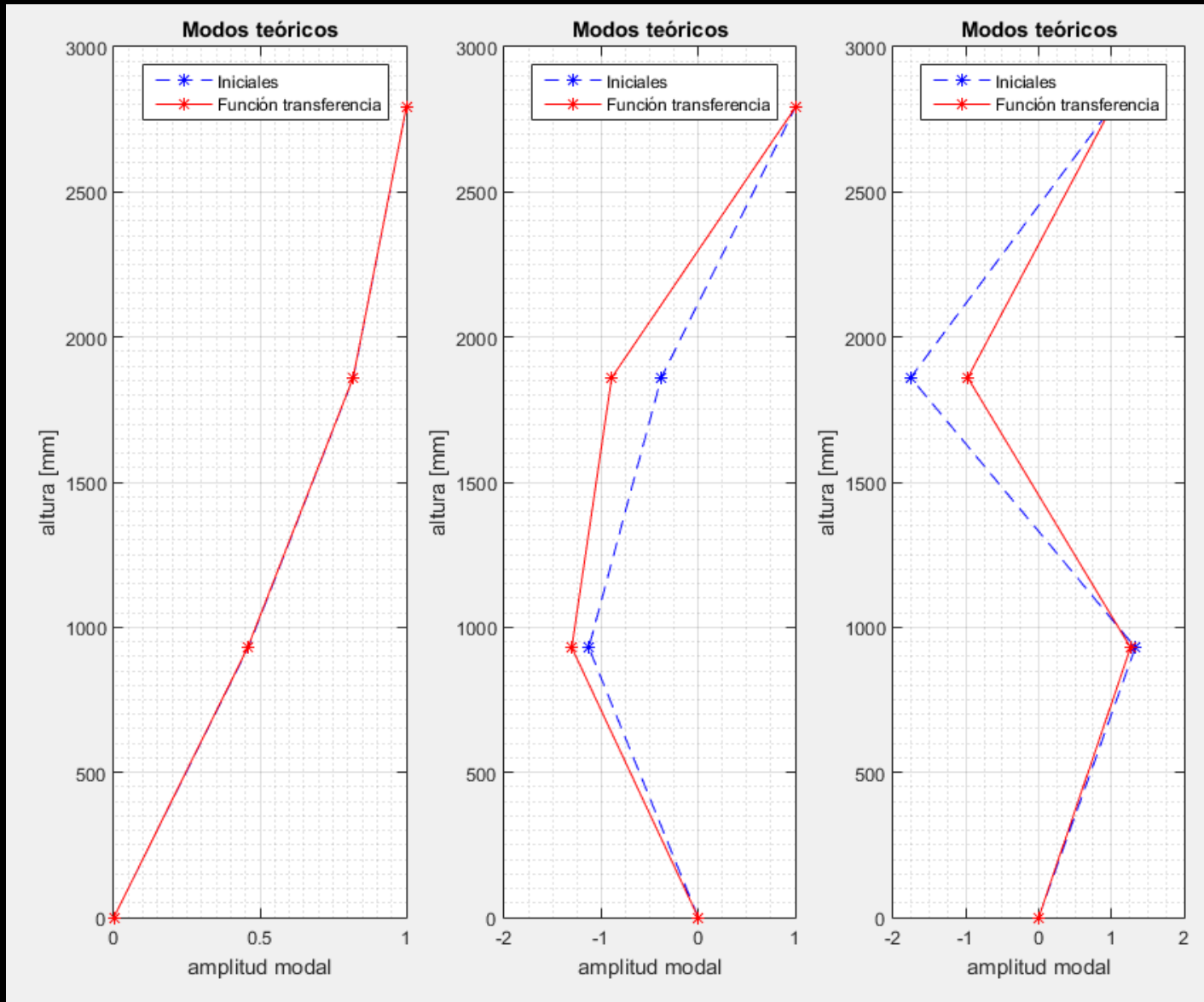


3. AJUSTE DE MODELOS

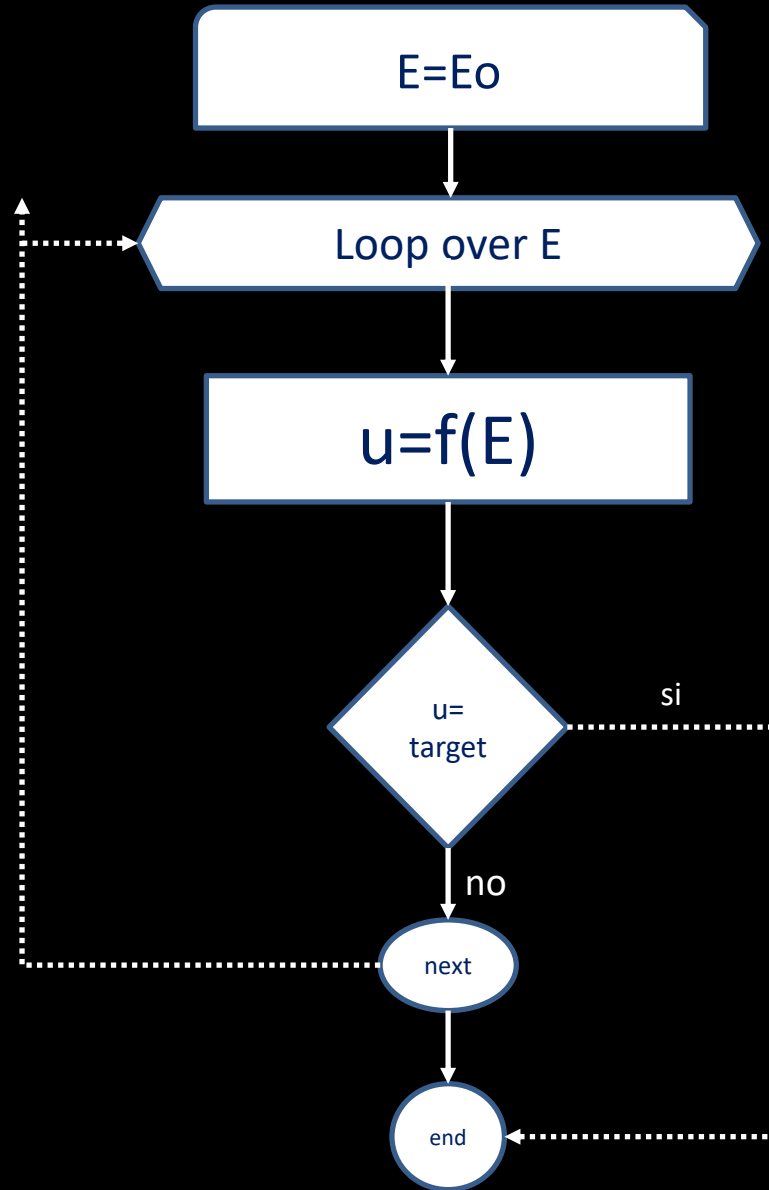
```
%ajuste de la inercia para acercarse a la 1a frecuencia de 3.5 Hz
dif_obj=0.00001*3.5;
dif=+inf;
I=I_ini;
iteracion=1;
while ((dif>dif_obj) && (iteracion<200))
    K=(E*I/h^3)*[24    -12    0;
                -12    24   -12;
                 0    -12   12];
    Kmed=mean(abs(K(find(reshape(K,size(K,1)*size(K,2),1)~=0))));
    [phi,wn2]=eig(K,M); % Vectores y valores propios
    f=wn/(2*pi); % Frecuencias (Hertz)
    dif=3.5-f(1,1);
    if abs(dif)>abs(dif_obj)
        I=I+.1*dif*I;
    end
    disp(strcat(num2str(iteracion),';',num2str(f(1,1)),';',num2str(I)));
    iteracion=iteracion+1;
end
```



3. AJUSTE DE MODELOS

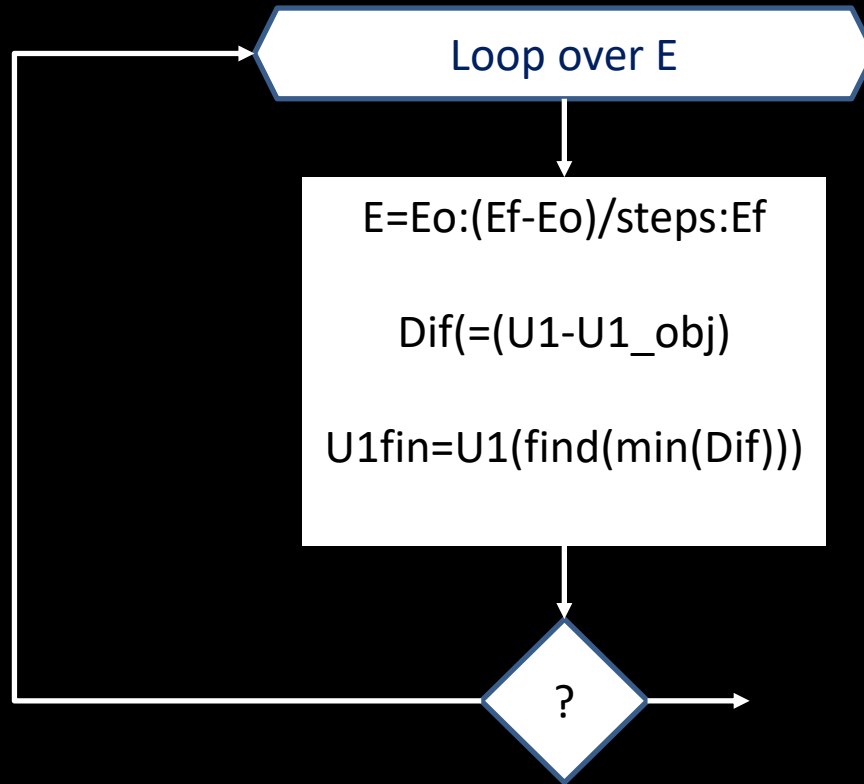


4. ALGORITMOS



4. ALGORITMOS

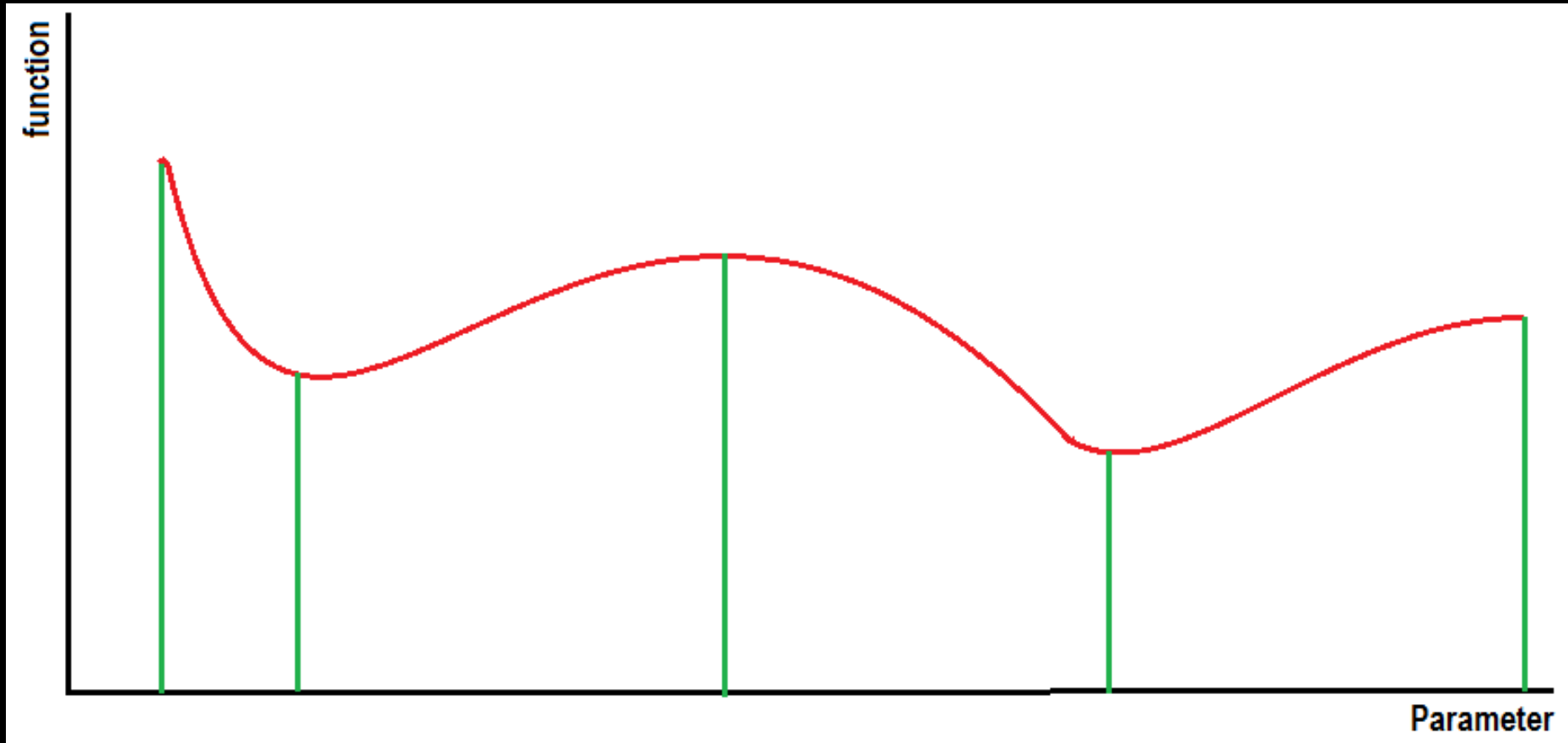
Most simple loop:



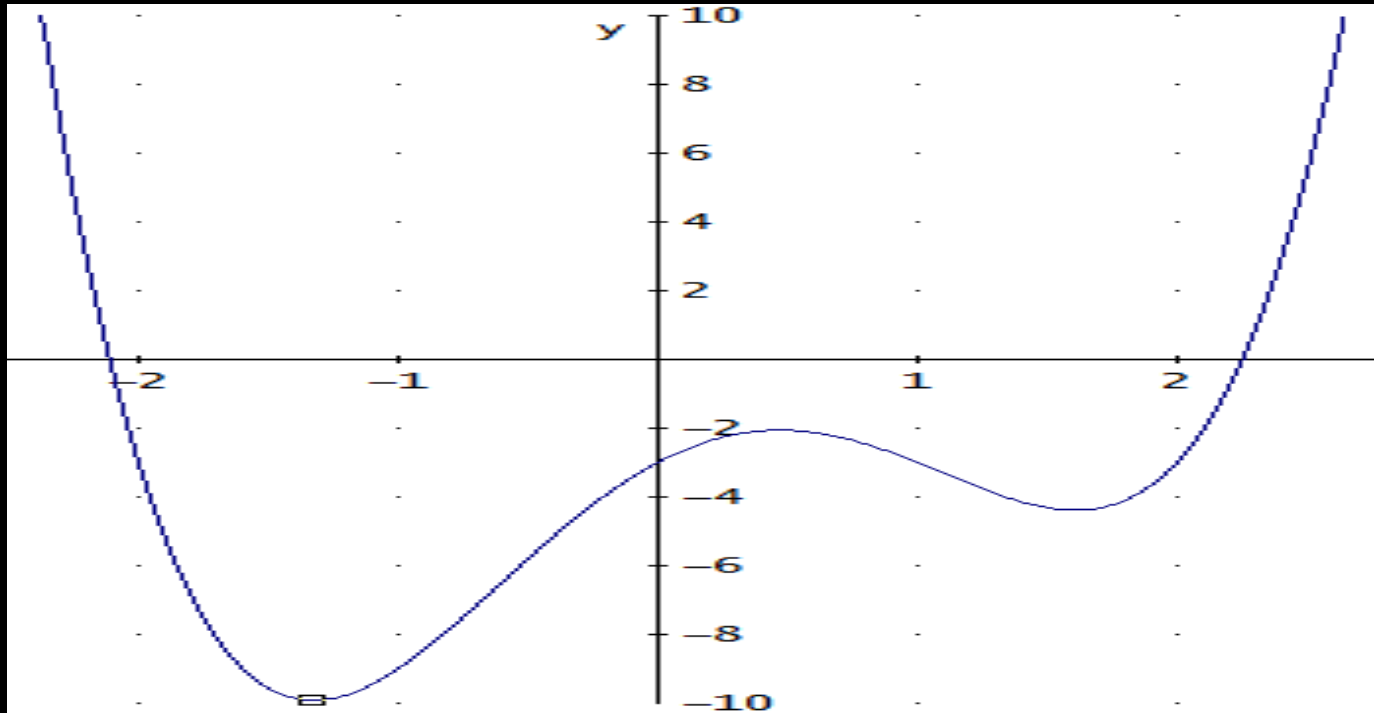
Slower,

But not necessarily most precise

5. MAXIMIZACIÓN O MINIMIZACIÓN



5. MAXIMIZACIÓN O MINIMIZACIÓN

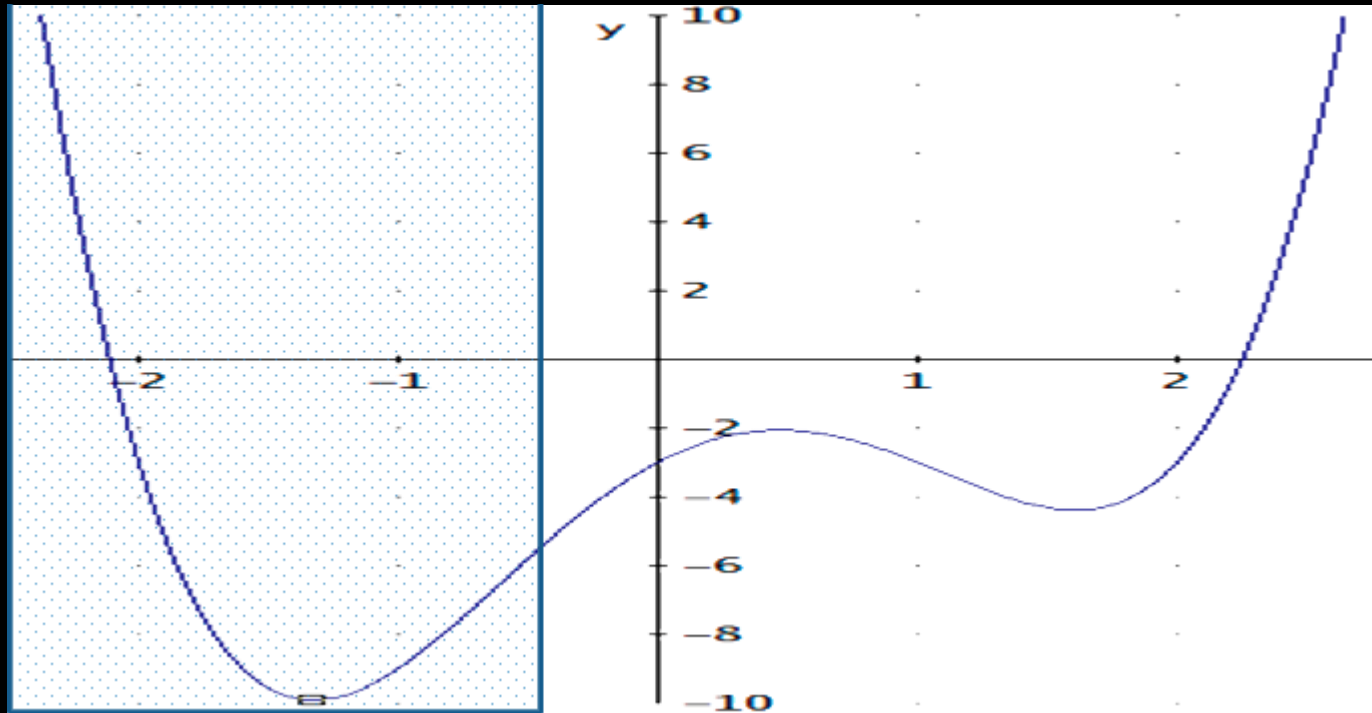


`fun='x^4-x^3-4*x^2+4*x-3'`

`x = fmincon(fun,x0,A,b)`

`A*x ≤ b.x0`

5. MAXIMIZACIÓN O MINIMIZACIÓN



`fun='x^4-x^3-4*x^2+4*x-3'`

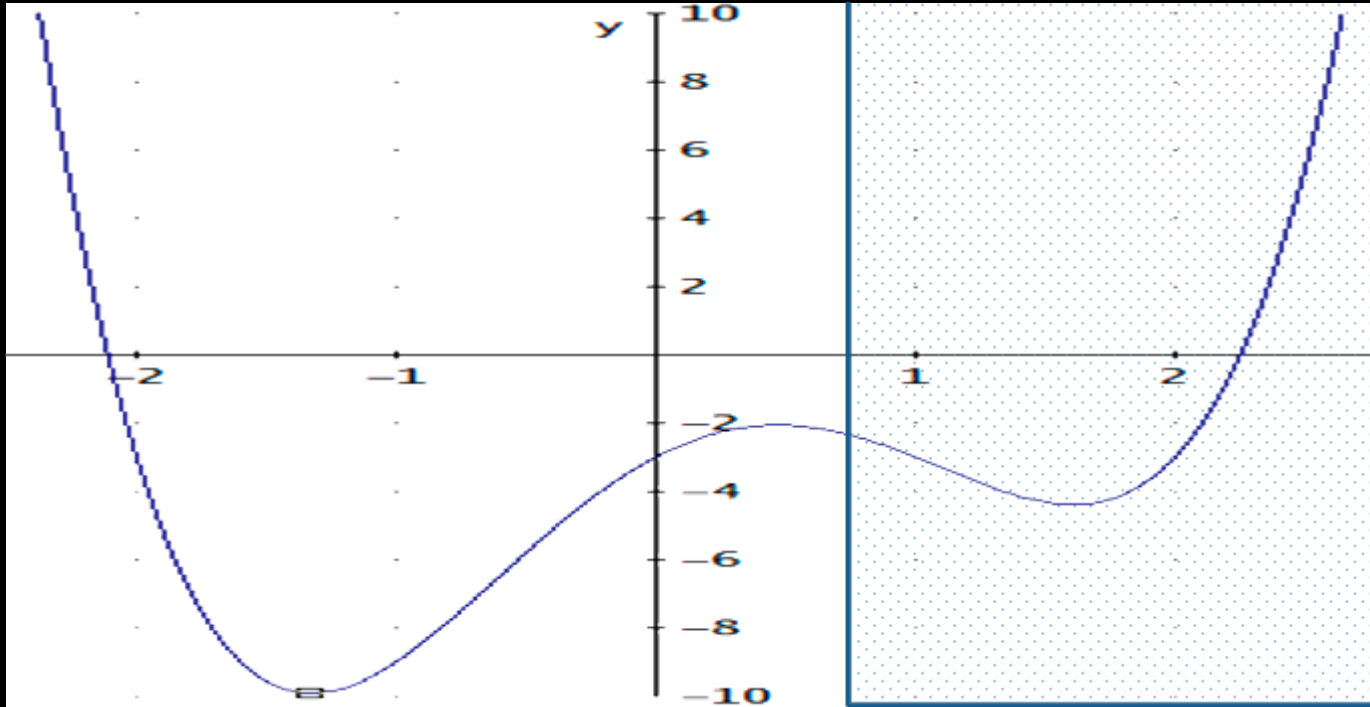
`x0=-10; A=1; b=1/20; %A*x ≤ b . x0`

`x = fmincon(fun,x0,A,b)`

`x = -1.3263`



5. MAXIMIZACIÓN O MINIMIZACIÓN



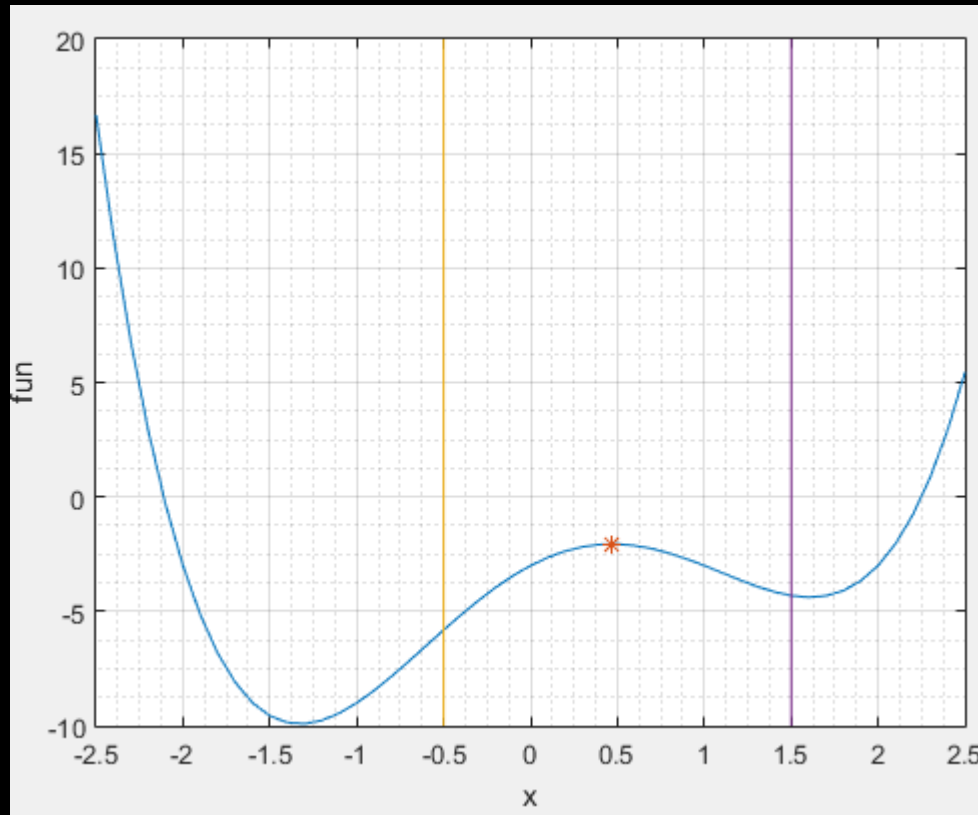
```
fun='x^4-x^3-4*x^2+4*x-3'
```

```
x0=0.8;A=-1; b=1; %A*x≤b.x0
```

```
x = fmincon(fun,x0,A,b)
```

```
x = 1.6073
```

5. MAXIMIZACIÓN O MINIMIZACIÓN



```
fun='x^4-x^3-4*x^2+4*x-3'
```

```
nfun=['-(',fun,')'];
```

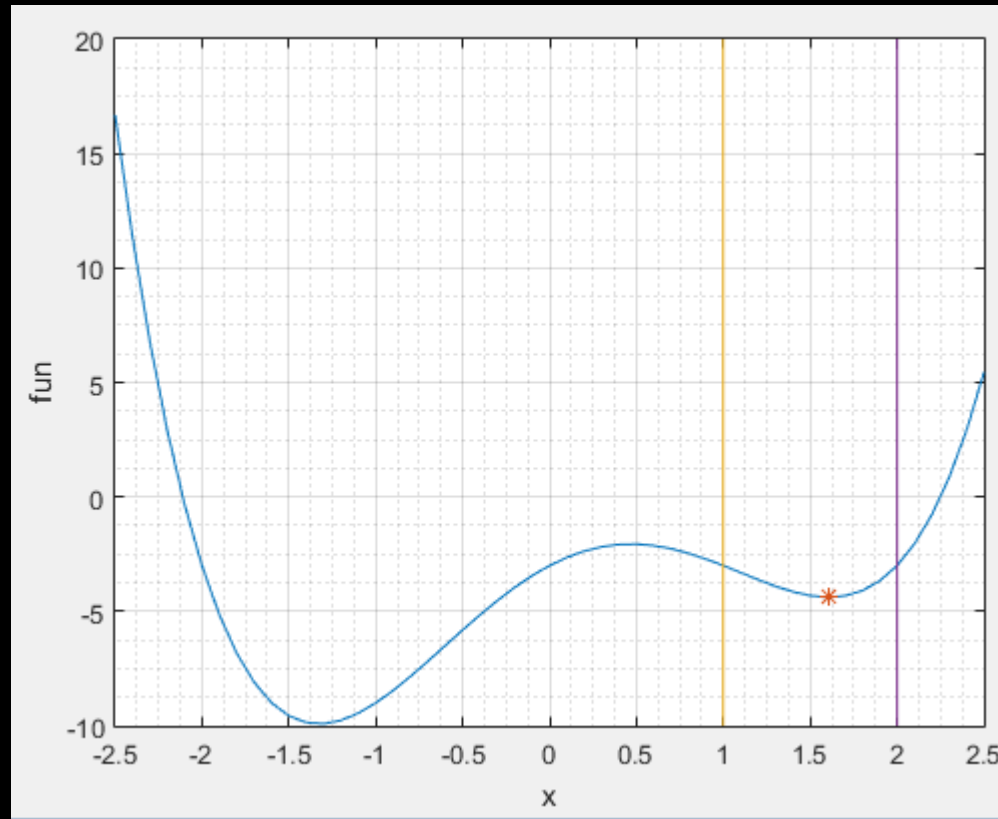
```
x = fminmax(nfun,x0,A,b,Aeq,beq,lb,ub)
```

```
A=[]; b=[]; Aeq=[]; beq=[]; lb=-0.5; ub=1.5;
```

```
x = 0.4691
```



5. MAXIMIZACIÓN O MINIMIZACIÓN



```
fun='x^4-x^3-4*x^2+4*x-3'
```

```
x = fminmax(fun,x0,A,b,Aeq,beq,lb,ub)
```

```
A=[]; b=[]; Aeq=[]; beq=[];
```

```
lb=1; ub=2;
```

```
x = 1.6073
```



5. MAXIMIZACIÓN O MINIMIZACIÓN

- fgoalattain
- fmincon
- fminimax
- fminunc
- fseminf
- fsolve
- intlinprog
- linprog
- lsqcurvefit
- lsqlin
- lsqnonlin
- particleswarm
- quadprog



Gracias

Créditos a:

<https://openai.com/dall-e-2> - <https://aminoapps.com/> - <https://miprofe.com/> - <https://www.youtube.com/@EASYCTE> -

Michel Bolaños Guerrero, Ing. C., Esp., Mag.
Candidato a Doctor en Ingeniería – Énfasis en Mecánica de Sólidos



Facultad de Ingeniería - Especialización en Estructuras

<https://michel.udenar.edu.co/> - michel@udenar.edu.co

Universidad de Nariño

[ini](#) [1](#) [2](#) [3](#) [4](#) [5](#) [fin](#)