Eurotherm Winter School, METTI 2005, Aussois (France), 16-21 january 2005

METTI 2005 Thermal Measurements and Inverse Techniques

> January 16 – 21, 2005 Centre Paul Langevin Aussois – France

A tool for the characterization of multiphysical phenomen

Workshop :

« How to solve the inverse problems with Femlab and Matlab environments »

By Sebastien ROUQUETTE*, Philippe LE MASSON°, Jialin GUO° & Tahar LOULOU°

*Laboratoire des Systèmes Mécaniques et Ingénierie Simultanée (LASMIS), Université de Technologie de Troyes, FRE CNRS 2719, 12 Rue Marie Curie, 10010 TROYES Cedex.

°Laboratoire d'Etudes thermophysiques, Energétiques et Environnement. Equipe Etude Thermophysique des Matériaux. Centre de recherche de l'Université de Bretagne Sud, Rue saint Maudé, 56321 LORIENT Cédex. sebastien.rouquette@utt.fr; philippe.le-masson@univ-ubs.fr; jialin.guo@univ-ubs.fr; tahar.loulou@univ-ubs.fr

Goals

- We use Levenberg Marquardt method for parameter estimation
- We use Femlab for the direct and sensitivity problem definitions
- We save these problems in a matlab file (« *.m »)
- We introduce the algorithm in this file
- At last the resolution of the inverse problem is realised with Matlab.

CONTENTS

- Resolution of the direct quenching problem with femlab
- The Levenberg-Marquardt algorithm
- Resolution of the sensitivity problem with femlab
- Resolution of the inverse boundary problem with matlab
- Conclusions





The direct quenching problem

• The governing equations [1] :

$$\rho C_{p} \frac{\partial T}{\partial t} - \nabla (k \cdot \nabla T) = 0 \in \Omega x T$$
$$-k \frac{\partial T}{\partial n} = 0 \text{ on } \Gamma_{2} x T$$
$$-k \frac{\partial T}{\partial n} = \varepsilon \sigma (T^{4} - T_{room}^{4}) \text{ on } \Gamma_{2} x T$$
$$-k \frac{\partial T}{\partial n} = Hcv (T - T_{room}) \text{ on } \Gamma_{3} x T$$
$$T (r, z; t = 0s) = 1173 K \in \Omega$$



[1] Essai de trempabilité jominy. Norme NF A 04-303 (juin 1979).

[2] P. Le Masson, P. Rogeon, D. Carron and J.J. Quémener. Simulation numérique thermométallurgique : influence des paramètres d'entrée sur la modélisation d'un essai de trempabilité Jominy. La revue de métallurgie-CIT/ Science et génie des matériaux, p.1055-1064, septembre 2000.

The direct quenching problem

- the resolution with femlab -

- Open FEMLAB3.1 with matlab
- In the model navigator window, choose « this configuration»
- And click OK

	Axial symmetry (2D)	_
Application Modes	cs and Conduction state analysis	Heat Transfer
	ent analysis nanics ng Module odule	Heat transfer through conduction with he flux, convective, and temperature bound conditions.
Electromagnetics M		Transient analysis in 2D axial symmetry.

- To draw a rectangle of 100mm x 12,5mm
- Click on the « // rectangle icon »
- Draw a rectangle in the draw window



Click two times on the rectangle and the object properties window appears Correct the dimensions-APPLY and OK Redimension the rectangle with the « Zoom extents icon »





In bar menu, choose «Physics » then « boundary settings » to define the boundary conditions.

1 6	EMLA	B - Geom	1/Co	onvec	tion and t	Conduct	ion (cc)	: [Untit	led]									_1
File	Edit	Options	Dra	w Ph	rysics Me	sh Solv	e Post	processin	ig Mult	iphysics	Help							
D	ê [6	16	Subdoma	in Setting	js	4 = 1	≅ 🐗	D.		* 12	Ω6 Ωδ	Ω ①	<₽ 8			
_					Boundary	Settings												
					Point Set	ings												
-	172		4		Scalar Va	riables,,,												•
0	۰	-0.68	L		Propertie	s			53 0.4	i-	<u> </u>	83 0.*	10		58 87			-
0	1		<u> </u>		Equation	System	•											
•	T	0.00			Periodic (Condition	s 🕨											
*	~	-0.03	Γ		Identity (Iondition	s 🕨						1.5		20	1.5		
-	2		205		Far-Field.													
+	~	-0.7	F.		Model Se	ttings		92 1	22			112	10	99 1	624	12	19	
*			L		Selection	Mode	•											
	画	-0.71	₹.	-13	36	2	10	55	82			0	19	8	10		1	1
Δ	10		L															
+	10	-0.72	F.	10	1		-	1	12		18		23	10	8	10	1	- 55-
			L															
7	ALL	-0.73	2	235	Q)	13	25	2	53	R	1	8	12	(Q)	33	25	(\$)	392
Ц	NOME																	_
D	D.	-0.74	L	-22			-22								10			
D		0																0.000
2		0.75																
=		-0.75	Γ	-25			-								223			
5																		
2		-0.76	F	100		-3	100	9.V	112			224		14	673.	12	14	
2																		
		-0.77	5	28	22	2	20	21	8		2	8	88		82			32.
K																		
		-0.78	2	23	12	82	20	23	82		0 0	8	23	12	85		12	- 272
			_	Ľ.	1	1	i.		1	r=0	1	1	1	1	1	1	1	
				0.06	.0.05	0.04	-0.03	-0.02	-0.01	0	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08



- Boundary 1: Axial symmetry
- Boundary 2: select heatflux h=HCV and Tinf ='Troom',
- Boundaries 3 and 4: select heat flux « const » ='stefan*emi' and « Tamb »='Troom'
- APPLY and click OK

$\mathbf{n} \bullet (k \nabla T) = q_0 + h(T_{inf} - T) + h(T) + h(T_{inf} - T) + h(T) + h$	Const(T _{amb} ⁴ - T ⁴)		
Boundary selection	Boundary sources ar	nd constraints	
1	Boundary condition:	Heat flux	3
2	Quantity	Value/Expression	Description
4	q ₀	0	Inward heat flux
	h	0	Heat transfer coefficient
	T _{inf}	0	External temperature
×	Const	stefan*emi	Problem-dependent constant
Select by group	Tamb	Troom	Ambient temperature
Interior boundaries	To	0	Temperature



- In « Physics » menu
 bar, choose
 « subdomain settings »
 to define the material
 properties.
- The subdomain settings window appears.

	W F	EMLAB	- Geom	1/Heat	Transre	r by Loi	nauccio	on (nca)	: tremp	e.fl				
	File	Edit	Options	Draw	Physics	Mesh	Solve	Postpro	cessing	Multip	hysics H	Help		
	D	i 🔁 🔛		6 1	Subd	omain Se	ttings		= ≌		PP		🖌 δΩ	32 26
				_	Boun Point	dary Setl Settings	tings							
	2			4	Scala Prope	r Variable erties	95	-	Ţ		Ţ.		ļ	
»Y			0.1	_	Equa Perio	tion Syst dic Condi	em tions	•	28					
	• ~		0.00		Ident Far-F Mode	ity Cond ield, I Setting:	itions	•						
			0.00	2	Selec	tion Mod	e	•			1			
			0.06	_			13		a					
	ø													



- In the coefficients domain selection, enter:
- KAC for the thermal
- RHOAC for the Density, ~
- CPAC for the heat capacity, conductivity,
- 0 for the heat source;
- In the init part, give the initial temperature : Tsteel
- OK





Go to the « options »
in the menu bar,
choose « constants » to
define all the
parameters and their
values.

FEMLAB - Geom1/Heat Transfer by Conduction (ht): [Untitled] Options Draw Physics Mesh Solve Postprocessing Multiph File Edit Axes/Grid Settings... 🗅 🚅 🗲 D Update Symbols R Constants... Expressions 0 Integration Coupling Variables **Extrusion Coupling Variables Projection Coupling Variables** Functions... Interpolation Functions... Coordinate Systems... Material Library... 函 Cross-Section Library... Visualization/Selection Settings... View Geometries... 0 Zoom 四日 Suppress HOWE Labels Preferences... Di



To validate the name of the parameter click « apply » to validate his value.

 $RHOAC = 7500 \text{ kg} / m^{3}$ CPAC = 520 J / kg / CKAC = 32 W / C / m

Troom = 20

emi = 0.6stefan = 5.67e - 8

$$Tsteel = 900$$

Name	Expression	Value	
RHOAC	7500	7500	
CPAC	520	520	
KAC	32	32	
Tsteel	900	900	
Troom	20	20	
HCV	15000	15000	
emi	0.6	0.6	
stefan	5.67e-8	5.67e-8	

•To exit: click OK

- Mesh step
- In the menu, select

 mesh » then « mesh
 parameters » to open
 the mesh parameters
 window



- Change the « meshgrowth rate »
- Refine the mesh near
 the quenched boundary
 2 : « maximum
 element size » = 5e-4
- Impose the size of the element in the domain 1 :
 « maximum element / size » = 2e-3
- And click the button « REMESH »



- Select « solve » in the bar menu,
- Then « solve parameters » and click
- The solver parameters window appears

🌃 FEMLAB - Geom1/Heat Transfer by Co	nduction (ht) : trempe.fl
File Edit Options Draw Physics Mesh	Solve Postprocessing Multiphysics Help
	= Solve Problem
	Solver Parameters
-0.5	
-0.52	
-0.54	

In the « general » menu, verify that is a 'time dependent' problem in « solver type »

In « solution form » select 'general'

Analysis:	General Time Stepping Advanc	ed	
Auto select solver	Time stepping Times: Relative tolerance:	0.0.1:60	
Stationary linear	Absolute tolerance:	0.0010	
Time dependent Eigenvalue Parametric linear	Linear system solver		
	Preconditioner: Incomple Drop tolerance: 0.01	te LU	
Adaption	Memory efficiency	Precond. quality	Settings
	Solution form:		
	Symmetric matrices		

The Levenberg-Marquardt method

- the inverse boundary problem formulation -

The inverse boundary problem formulation [3]: Find the parameter Z={H,E} which minimizes the quadratic criterion S(Z,T): $S(\overline{Z},T)=[Y_i-T_i(\overline{Z})]^T W[Y_i-T_i(\overline{Z})]$

With Yi is the measurements, Ti the calculated temperature, and W a diagonal matrix where the diagonal elements are given by the inverse of the standard deviation of the measurement errors, i is the total number of measurements.

The Levenberg-Marquardt Method says [4,5]:

$$\overline{Z^{k+1}} = \overline{Z^{k}} + [J^{T}WJ + \lambda^{k}\Omega^{k}]^{-1}J^{T}W[Y_{i} - T_{i}(\overline{Z_{k}})]$$

with J is the sensitivity matrix, λ is the damping parameter and Ω is a diagonal matrix equal to the identity matrix.

[3] A.N. Tikhonov & V.Y. Arsenin. Solutions of ill-posed problems. V.H. Wistom & Sons, Washington, DC (1977).
[4] K. Levenberg. A method for the solution of certain non linear problems in least squares. Quart. Appli. Math. 2 (1944) 4164-168.
[5] D.W. Marquardt. An algorithm for least squares estimation of non linear parameters. J. soc. Ind. Appli. Math. 11 (1963) 431-441.

The direct quenching problem

- the resolution with femlab -

- Solve the direct quenching problem by using the « solve » icon (symbol equal)
- We obtain the temperature field at the final time

0.1

		Assembling matric	es		
ogress Log Description	Progress	Convergence	Parameter	Value	
ïme-dependent solver	11 %		Time	6.37	Stop
Assembly	27 %				Stop
					Stop
Close automatically					Cancel



-0.06

The Levenberg-Marquardt method

- the inverse boundary problem formulation -

The inverse boundary problem formulation [3]: Find the parameter Z={H,E} which minimizes the quadratic criterion S(Z,T): $S(\overline{Z},T)=[Y_i-T_i(\overline{Z})]^T W[Y_i-T_i(\overline{Z})]$

With Yi is the measurements, Ti the calculated temperature, and W a diagonal matrix where the diagonal elements are given by the inverse of the standard deviation of the measurement errors, i is the total number of measurements.

The Levenberg-Marquardt Method says [4,5]:

$$\overline{Z^{k+1}} = \overline{Z^{k}} + [J^{T}WJ + \lambda^{k}\Omega^{k}]^{-1}J^{T}W[Y_{i} - T_{i}(\overline{Z_{k}})]$$

with J is the sensitivity matrix, λ is the damping parameter and Ω is a diagonal matrix equal to the identity matrix.

[3] A.N. Tikhonov & V.Y. Arsenin. Solutions of ill-posed problems. V.H. Wistom & Sons, Washington, DC (1977).
[4] K. Levenberg. A method for the solution of certain non linear problems in least squares. Quart. Appli. Math. 2 (1944) 4164-168.
[5] D.W. Marquardt. An algorithm for least squares estimation of non linear parameters. J. soc. Ind. Appli. Math. 11 (1963) 431-441.



The Levenberg-Marquardt method

- the sensitivity matrix -

• Sensitivity coefficients calculus [6]:

First method: $J_{\overline{Z}} = \frac{\partial T(\overline{Z})}{\partial \overline{Z}}$ Second method: $J_{Z} = \frac{T(Z + \varepsilon Z) - T(Z - \varepsilon Z)}{2\varepsilon Z}$

The expression of the sensitivity matrix becomes:

$$J[H,E] = \begin{bmatrix} \frac{\partial T_1}{\partial H} & \frac{\partial T_2}{\partial H} & \cdots & \frac{\partial T_1}{\partial H} \\ \frac{\partial T_1}{\partial E} & \frac{\partial T_2}{\partial E} & \cdots & \frac{\partial T_I}{\partial E} \end{bmatrix}$$

Stopping criterion:

 $S(\overline{Z^k},T) \leq \varepsilon$

[6] M.N. Osizik, H.R.B. Orlande, Inverse heat transfer: fundamentals and applications, Taylor and Francis, New York, 2000.

The Levenberg-Marquardt method - the algorithm -

- Levenberg-Marquardt Algorithm:
- 1-Solve the direct problem with for the unknown parameters Z^k to obtain the calculated temperatures $T(Z^k)$
- 2-Compute $S(Z^k, T(Z^k))$ 3-Compute the sensitivity matrix $J(Z^k)$ et the matrix Ω^k . 4-Calculate the new estimated Z^{k+1} .
- 5-Solve the direct problem with Z^{k+1} , Compute $S(Z^{k+1}, T(Z^{k+1}))$ 6-if $S(Z^{k+1}) > S(Z^k)$, replace λ^k by $10 * \lambda^k$ and go back to step 4 else if $S(Z^{k+1}) < S(Z^k)$, replace λ^k by $0, 1 * \lambda^k$ and continue. 7-Test if $S(Z^{k+1}) < \varepsilon$, Stop if it is true else do k = k + 1 and go back to step 3.

Let define the sensitivity problems under femlab ...



The sensitivity problems - the resolution with femlab -

First, in the bar menu, click on « Multiphysics » and select « Model Navigator », a window appears. Now add two new heat conduction systems : ht2 and ht3.

Axial symmetry (2D) s and Conduction	+	Multiphysics Asto Remove Remove Reat Transfer by Conduction (ht Heat Transfer by Conduction (ht Heat Transfer by Conduction (ht
s and Conduction state analysis		Astrine Remove Seom1 (2D) Heat Transfer by Conduction (ht Heat Transfer by Conduction (ht Heat Transfer by Conduction (ht
nt analysis anics g Module idule	.	Dependent variables: T3 Application Mode Properties Add Geometry
	F	Ruling application mode:
2		Heat Transfer by Conduction (ht)
ırange - Quadratic	-	Multiphysics
	it analysis anics g Module dule dule range - Quadratic	it analysis anics g Module dule

The sensitivity problems

- the resolution with femlab -

Go to « multiphysics » and select the 2nd mode (ht2) In « physics » menu ... « boundary settings » to define the boundary conditions of the sensitivity problem for the convective coefficient J_{HCV} .

$$\rho C_{p} \frac{\partial J_{HCV}}{\partial t} - \nabla (k \cdot \nabla J_{HCV}) = 0 \in \Omega \times T$$

$$-k \frac{\partial J_{HCV}}{\partial \vec{n}} = 0 \text{ on } \Gamma_{2} \times T$$

$$-k \frac{\partial J_{HCV}}{\partial \vec{n}} = (T - T_{room}) + HCV * J_{HCV} \text{ on } \Gamma_{2} \times T$$

$$-k \frac{\partial J_{HCV}}{\partial \vec{n}} = 4 \varepsilon \sigma T^{3} J_{HCV} + \sigma (T^{4} - T_{room}^{4}) \text{ on } \Gamma_{3,4} \times T$$

$$J_{HCV}(r, z; t = 0 \text{ s}) = 0 \in \Omega$$



The sensitivity problems - the resolution with femlab -Boundary selection : 1 :'axial symmetry' $\rho C_p \frac{\partial J_{HCV}}{\partial t} - \nabla (k \cdot \nabla J_{HCV}) = 0 \in \Omega \times T$ 2 : 'heat flux' $-k \frac{\partial J_{HCV}}{\partial \vec{n}} = 0 \text{ on } \Gamma_2 x T$ 3 4 : 'heat flux' \Rightarrow h : '4*emi*stefan*T^3' $-k \frac{\partial J_{HCV}}{\partial \vec{n}} = (T - T_{room}) + HCV * J_{HCV} \text{ on } \Gamma_2 x T$ $-k \frac{\partial J_{HCV}}{\partial \vec{n}} = 4 \varepsilon \sigma T^3 J_{HCV} + \sigma (T^4 - T^4_{room}) on \Gamma_{3.4} x T$ $J_{HCV}(r,z;t=0s)=0\in\Omega$ Equation $\mathbf{n} \cdot (k \nabla T2) = q_0 + h(T_{inf} - T2) + Const(T_{amb} + T2^4)$ Boundary selection Boundary sources and constraints Boundary condition: Heat flux Value/Expression Description Quantity -(T-Troom) Inward heat flux q_o Heat transfer coefficient HCV. External temperature Tinf Const Problem-dependent constant Select by group Tamb Ambient temperature T₀ Temperature Interior boundaries OK Cancel Apply

The sensitivity problems

- the resolution with femlab -

In « physics » menu ... select « subdomain settings » to define the domain properties. And verify that the initial value is equal to zero.

quation	/		
i _{ts} pC _p ∂T2/∂t - ∇•(k∀T2) = G	2, T2= temperature		
	K		
Subdomain selection	Physics Init Elemen	nt]	
	Thermal properties a	nd heat sources/sinks	k
	Library material:	▼ Lo	ad
	Quantity	Value/Expression	Description
	ō _{ts}	1	Time-scaling coefficient
	🔍 k (isotropic)	KAC	Thermal conductivity
	C k (anisotropic)	400 0 0 400	Thermal conductivity
	ρ	RHOAC	Density
Select by group	Cp	CPAC	Heat capacity
Active in this domain	Q	0	Heat source
2		500 (100)	



The sensitivity problems - the resolution with femlab -

• Go to « multiphysics » and select the 3rd mode (ht3)

• In « physics» menu ... « boundary settings » to define the boundary conditions of the sensitivity problem for the emissivity coefficient.

$$\rho C_{p} \frac{\partial J_{emi}}{\partial t} - \nabla (k \cdot \nabla J_{emi}) = 0 \in \Omega \times T$$

$$-k \frac{\partial J_{emi}}{\partial \vec{n}} = 0 \text{ on } \Gamma_{2} \times T$$

$$-k \frac{\partial J_{emi}}{\partial \vec{n}} = HCV * J_{emi} \text{ on } \Gamma_{2} \times T$$

$$-k \frac{\partial J_{emi}}{\partial \vec{n}} = 4 \varepsilon \sigma T^{3} J_{emi} + \sigma (T^{4} - T_{room}^{4}) \text{ on } \Gamma_{3,4} \times T$$

$$J_{emi} (r, z; t = 0 \text{ s}) = 0 \in \Omega$$



The sensitivity problems

- the resolution with femlab -

In « physics » menu ... select « subdomain settings » to define the domain properties. And verify that the initial value is equal to zero.

_{is} pC _p ∂T3/∂t - ⊽•(k⊽T3) = G), T3= temperature		
ubdomain selection	Physics Init Elemer	t]	
	Thermal properties a	nd heat sources/sinks	
	Library material:	Lo	ad
	Quantity	Value/Expression	Description
	ō _{ts}	1	Time-scaling coefficient
	k (isotropic)	KAC	Thermal conductivity
	C k (anisotropic)	400 0 0 400	Thermal conductivity
-	ρ	RHOAC	Density
Select by group	C _p	CPAC	Heat capacity
Z Active in this domain	Q	0	Heat source



The sensitivity problems - the resolution with femlab -

In the bar menu, choose « file » then « reset M-file » before solving simultaneously the direct problem and the two sensitivity problems.

Solve these three problems by clicking on « solve (=) »

FEMLAB - Geom1/Heat Transfer by Conduction (ht3) : trempe.fl File Edit Options Draw Physics Mesh Solve Postprocessing Multiphysics Help New... Ctrl+N S D D D S Open Model Library... 🔁 Open... Ctrl+O Save Ctrl+S Save As.... Ctrl+P 🞒 Print... Generate Report Ctrl+G Model Properties... Save Model Image Reset Model... Import Export Movie Player... 1 C:\...\Aussois\trempe.fl Exit



The sensitivity problems

- the resolution with femlab -

- Save the data in a Mfile & .flb file (keep these two file is the same folder).
- Go to « file » in the menu, choose « Save As ». Save it as a matlab file (.m exetension)
- Name it 'quenching2p'
- The program generates a quenching2p.m file.

File Edit Options Draw	Physics	Mesh	Solve	Postpro	cessing
D New Open Model Library	Ctrl+N			-3	= 🗎
Dpen	Ctrl+O				
🔚 Save	Ctrl+S		1		- 1
Save As			80 80		12
🖨 Print	Ctrl+P				
Generate Report	Ctrl+G				
Model Properties					
Save Model Image					
Reset Model			10		8
Import	•				
Export	•	C			
Movie Player					
1 C:\\Aussois\trempe.f	I				12
2 C:\\Aussois\toto2.m					
3 C:\\Aussois\toto.m					
Exit					
0.04 -					



^{-0.04 -0.02 0 0.02 0.04 0.06} Min: -3.04

.0.06

Min: -3.044e-3

The sensitivity problems

- sensitivity results -



- Go to Matlab environments and open a new M-file in which we are going to program the inverse problem program.
- Name it « main_quenching2p.m »
- To do that : go to « file » menu then « save as »
- Give a title at the begin of the file (the sentence is preceeded of % symbol for comments).

New	0	M-file	Directory: C:WATLAB6p1\work	
Open	Ctrl+O	Figure		
Close Command Window	Ctrl+W	Model		×
Import Data		GOI	_	
Save Workspace As	Ctrl+S	Class		1
Set Path				
Preferences				
Print		1		
Print Selection				
1 C:\1\work\FE3D_test2.r	n			
2 C:\MATLAB6p1\work\FE3D).m			
3 C:\\work\fine_mesh3e.r	n			
4 C:\work\fine_mesh3e2.i	m			
Exit MATLAB	Ctrl+Q			
Launch Pad	Norkspace			
Incasconal The		0		

The inverse boundary problem resolution

- building of the inverse problem program -



The inverse boundary problem resolution

- building of the inverse problem program -

• First let solve the direct problem with the true parameters to obtain the theoretical temperatures Yi.

• Write the followings lines.

:\[)0(cuments and Settings\sébastien\Bureau\Aussois\main_quenching2p.m					
E	dit	View Text Debug Breakpoints Web Window Help					
	3	🖬 🕺 🛍 🛍 🗠 🖙 🎒 👭 🐔 🗧 😫 👘 🗊 📭 🏭 Stack: Base					
1		%Identification of the emissivity and the convective heat transfer coeffici					
2		%in the quenching process. Used optimization method : Levenberg-Marquardt					
3							
4 -	-	clear					
5							
6 -		global coefH coefE					
7 -	1	global ITEMPS					
8							
9 -	•	ITEMPS=[0:0.01:1,1.2:0.2:60];%time interval					
0							
1		%Resolution of the direct problem to obtain the exact temperatures (Yi)					
2 -		coefH=15000;%initialization of the convective heat transfer coefficient					
3 -	-	coefE=0.6;%initialization of the emissivity coefficient					
4 -	1	toto;%call of the direct problem generates with femlab					
5							
5		%interpolation of the exact temperatures (Y1) at point ncl and save it					
		nci=[U.UI2;U.UU2];%r=Um and z=2mm					
0 -		[Yi]=postinterp(remquench, 'l', ncl, 'soinum', l: length(rem.sol.tlist));					
9 -		Save Iquench.txt 11 -ASUI					
1		< m=road('iquench.cxc');					
2		> for noised terrereture uncompant these following lines .					
2 -	•	<pre>for (it-l:length(Vi))</pre>					
4 -		Vi(it)=Vi(it)+0.01*Vi(it)*cos(3.14*rand(1)). $\&$ the theoritical temperat					
5 -		end * cos(3.14*rand(1)) * cos(3.14*rand(1)) *					
6		· · · · · · · · · · · · · · · · · · ·					

Initial set of Solve the direct problem Parameters Z⁰ & Y Compute the calculated temperature and the Quadratic criterion S(Z^k) Yes Verify If $S(Z^k) < \mathbb{I}$ End No Solve the sensitivity problems, Calculate the sensitivity matrix $J(Z^k)$ Estimate the new set of parameters Z^{k+1} Solve the direct problem for the new set of parameters Compute the Divide I by 10 Multiply I by 10 If $S(Z^k) > S(Z^k)$ If $S(Z^k) \leq S(Z^k)$ quadratic criterion S(Z^k

The inverse boundary problem resolution

- building of the inverse problem program -

Initialize the parameters of the levenberg-Marquardt method : unknown parameters, damping parameter, weight matrix, first criterion, ... (for estimation just the first measurements are useful! because the values of the sensitivity coefficient ...)

27 % ****************************** beginning of the inverse problem resolution procedure ******* 28 % by the levenberg-Marguardt method : Pk+l = Pk + inv{[Jh'*WJ*Jh+lambda(k)*OMEG(k)]}*{Jh 29 % where Pk is the unknown parameter at iteration k, Jh the sensitivity coefficients, lam 30 % WJ a matrix, OMEG an other matrix, Yi the measured data and Tcal the calculated data 31 32 %Initializating of the unknown parameter coefH 33 coefH=0: 34 coefHprec=coefH;%necessary for the levenberg-marguardt method, value of the unknown para 35 coefE=0; 36 coefEprec=coefE; 37 38 %Initialization of the lambda coefficient used in the levenberg-marguardt method 39 lambda=0.001; 40 41 Screation of the matrix WJ (equal to the identity matrix when measurements are not noise 42 WJ=eye(length(Yi(1:21))); 43 44 %Creation of the matrix OMEG equal to a two dimension identity matrix for two parameters 45 OMEG=eye(2); 46 47 SOM=Yi-20; 48 somme=(norm(SOM)*norm(SOM));%initialization of the quadratic criterion (exact data minus 49 niter=1;%initialization of the iteration number

50

Initial set of Solve the direct problem Parameters Z⁰ & Y Compute the calculated temperature and the Quadratic criterion S(Z^k) Yes Verify If S(Z^k)< End No Solve the sensitivity problems, Calculate the sensitivity matrix $J(Z^k)$ Estimate the new set of parameters Z^{k+1} Solve the direct problem for the new set of parameters Compute the Divide I by 10 Multiply I by 10 If $S(Z^k) > S(Z^k)$ If $S(Z^k) \leq S(Z^k)$ quadratic criterion S(Z^k



- Open the beginning of the optimization loop
- Solve the direct problem for the set of initial unknown parameters
- Interpolate the calculated temperatures
- Compute the quadratic criterion (called « somme » here)

```
51
       %beginning of the optimization loop
52
      while ((somme>le-3)&(niter<25))</pre>
53
54
       %resolution of the direct problem (and the sensitivity problem) at iteration k=0
55
       toto;
56
57
       %Mesure de la température calculée à la première itération k=0
      [Tcal]=postinterp(femquench, 'T',ncl, 'solnum',l:length(fem.sol.tlist));
58
59
60
       %Calculation of quadratic criterion S(k)=(Tcal(t)-Yi(t))*(Tcal(t)-Yi(t))
61
       SOM=Yi-Tcal;
62
       somme=norm(SOM) *norm(SOM);
63
      sprintf('critere %18.8f à 1 iteration %3f',somme,niter)
64
65
```

Initial set of Solve the direct problem Parameters Z⁰ & Y Compute the calculated temperature and the Quadratic criterion $S(Z^k)$ Yes Verify If S(Z^k)< End No Solve the sensitivity problems, Calculate the sensitivity matrix $J(Z^k)$ Estimate the new set of parameters Z^{k+1} Solve the direct problem for the new set of parameters Compute the Divide I by 10 Multiply I by 10 If $S(Z^k) > S(Z^k)$ If $S(Z^k) \leq S(Z^k)$ quadratic criterion S(Z^k



- building of the inverse problem program -

```
66
      if (niter==1)
67
               %case niter=1 (for the first iteration)
68
               somprec=somme;
69
              EVOLH(niter)=coefH;
70
               EVOLE(niter)=coefE;
71
              EVOLSOM(niter)=somme;
72
73
               %interpolation of the sensitivities coefficients
74
              [Jh]=postinterp(femquench, 'T2',ncl, 'solnum', l:length(fem.sol.tlist));
75
              [Je]=postinterp(femquench, 'T3',ncl, 'solpum', 1:length(fem.sol.tlist));
76
               Jhe=[Jh(1:21) Je(1:21)];
                                                        We use the 21 first measurements
77
78
               SEstimation of the parameter coeff with Pror the estimation lambda(k)
              dcoef = inv((Jhe'*WJ*Jhe+lambda*OMEG))*(Jhe'*WJ*SOM(1:21)).
79
80
              coefH = coefHprec + dcoef(1);
81
82
               coefE2 = coefEprec + dcoef(2);
83
84
              if ((coefE2>0)&(coefE2<1)) coefE=coefE2;</pre>
85
               end
86
87
               sprintf('New estimated values for coefH %8.1f and coefE %6.3f', coefH, coefE)
88
89
               niter=niter+1;
90
```

• Case of the first iteration k=1 : Computation of the new estimated, and read of the sensitivity coefficients values ('Jh' and 'Je')

Initial set of Solve the direct problem Parameters Z⁰ & Y Compute the calculated temperature and the Quadratic criterion $S(Z^k)$ Yes Verify If $S(Z^k) < \mathbb{I}$ End No Solve the sensitivity problems, Calculate the sensitivity matrix $J(Z^k)$ Estimate the new set of parameters Z^{k+1} Solve the direct problem for the new set of parameters Compute the Divide I by 10 Multiply I by 10 If $S(Z^k) \leq S(Z^k)$ If $S(Z^k) > S(Z^k)$ quadratic criterion S(Z¹

91	-	else	
92	-	if	(somme <somprec)< td=""></somprec)<>
93			%case somme≺somprec (when the new calculated quadratic criterion is smaller t
94	-		lambda = 0.1*lambda;%divide by 10 the damping parameter
95	-		somprec=somme;
96	-		EVOLH(niter)=coefH;
97	-		EVOLE(niter)=coefE;
98	-		EVOLSOM(niter)=somme;
99			
00			%interpolation of the sensitivities coefficients
01	-		[Jh]=postinterp(femquench, 'T2',ncl, 'solnum',l:length(fem.sol.tlist));
02	-		[Je]=postinterp(femquench, 'T3',ncl, 'solnum',l:length(fem.sol.tlist));
03	-		Jhe=[Jh(1:21) Je(1:21)];
04			
05			<pre>%Estimation of the parameter coefH with Pk+1 = Pk + inv{[Jhe'*WJ*Jhe+lambda(k</pre>
06	-		dcoef = inv((Jhe'*WJ*Jhe+lambda*OMEG))*(Jhe'*WJ*SOM(1:21));
07	-		coefHprec = coefH;
08	-		coefEprec = coefE;
09	-		<pre>coefH = coefHprec + dcoef(1);</pre>
10	-		<pre>coefE2 = coefEprec + dcoef(2);</pre>
11			
12	-		if ((coefE2>0)&(coefE2<1)) coefE=coefE2;
13	-		end
14			
15	-		sprintf('New estimated values for coefH %8.1f and coefE %6.3f',coefH,coefE)
16			
17	-		niter=niter+1;
18			

• Case $S(Z^{k+1}) \leq S(Z^k)$

Initial set of Solve the direct problem Parameters Z⁰ & Y Compute the calculated temperature and the Quadratic criterion $S(Z^k)$ Yes Verify If $S(Z^k) < \mathbb{I}$ End No Solve the sensitivity problems, Calculate the sensitivity matrix $J(Z^k)$ Estimate the new set of parameters Z^{k+1} Solve the direct problem for the new set of parameters Compute the Divide I by 10 Multiply I by 10 If $S(Z^k) > S(Z^k)$ If $S(Z^k) \leq S(Z^k)$ quadratic criterion S(Z^k



119 -	else
120	%case somme>somprec (when the new calculated quadratic criterion is bigger t)
121 -	lambda=10*lambda;%multiply by 10 the damping parameter
122	
123	%Estimation of the parameter coefH with Pk+l = Pk + inv{[Jhe'*WJ*Jhe+lambda(]
124 -	<pre>dcoef = inv((Jhe'*WJ*Jhe+lambda*OMEG))*(Jhe'*WJ*SOM(1:21));</pre>
125 -	<pre>coefH = coefHprec + dcoef(1);</pre>
126 -	<pre>coefE2 = coefEprec + dcoef(2);</pre>
127	
128 -	<pre>if ((coefE2>0)&(coefE2<1)) coefE=coefE2;</pre>
129 -	end
130	
131 -	sprintf('New estimated values for coefH %8.1f and coefE %6.3f',coefH,coefE)
132	
133 -	end
134	
135 -	end
136	
137 -	end%end of the optimization loop
138	

- Case $S(Z^{k+1}) > S(Z^k)$
- And close all the loop

Initial set of Solve the direct problem Parameters Z⁰ & Y Compute the calculated temperature and the Quadratic criterion S(Z^k) Yes End Verify If $S(Z^k) < \mathbb{I}$ Post processing No Solve the sensitivity problems, Calculate the sensitivity matrix $J(Z^k)$ Estimate the new set of parameters Z^{k+1} Solve the direct problem for the new set of parameters Compute the Divide I by 10 Multiply I by 10 If $S(Z^k) > S(Z^k)$ If $S(Z^k) \leq S(Z^k)$ quadratic criterion S(Z^k

The inverse boundary problem resolution

- building of the inverse problem program -

139	%post-processing step
140	
141 -	<pre>it=1:1:niter-1;</pre>
142 -	EVOLHEXA(1:1:niter-1)=15000;
143 -	<pre>EVOLEEXA(1:1:niter-1)=0.6;</pre>
144	
145 -	figure(1)
146 -	plot(ITEMPS,SOM)
147 -	grid
148 -	xlabel('time (s)')
149 -	ylabel('Temperature difference')
150 -	Legend('Evolution of difference between theoritical and calculated temperature',1)
151	
152 -	figure(3)
153 -	plot(ITEMPS,Tcal, <mark>'s'</mark> ,ITEMPS,Yi,'')
154 -	grid
155 -	<pre>ylabel('Temperature (°C)')</pre>
156 -	xlabel('time (s)')
157 -	legend('Calculated temperature','Theoritical temperature',1)
158	
159 -	figure(4)
160 -	semilogy(it,EVOLSOM)
161 -	grid
162 -	ylabel('Quadratic criterion')
163 -	xlabel('Iteration number')
164 -	legend('Evolution of the quadratic criterion in function of iteration number',3)
165	

•To visualize the results of the numerical identification (1)



The inverse boundary problem resolution

- building of the inverse problem program -

```
166 -
        figure(5)
167
        plot(it,EVOLHEXA,it,EVOLH,'-.s')
168
        grid
169
       ylabel('H parameter')
170
       xlabel('Iteration number')
171
        title ('Evolution of the H parameter in function of iteration number')
        Legend('Theoritical Value', 'Calculated Value', 4)
172
173
174
        figure(6)
       plot(it,EVOLEEXA,it,EVOLE,'-.s')
175
176
        grid
177
        ylabel('Emissivity parameter')
178 -
       xlabel('Iteration number')
179
        title('Evolution of the emissivity parameter in function of iteration number')
        Legend('Theoritical Value', 'Calculated Value', 4)
180 -
```

•To visualize the results of the numerical identification (2)





- building of the inverse problem program -

At the end of the file:

In the solver, replace time vector by the vector named 'ITEMPS'Comment the postplot process

```
20
        % Solve problem
 21
        fem.sol=femtime(fem, ...
                         'solcomp', {'T3', 'T', 'T2
                         'outcomp', {'T3', 'T', 'T2'}, ...
                         'tlist', ITEMPS, ...
                         'tout','tlist', ...
125
126
                         'linsolver', 'cg');
127
128
        % Save current fem structure for restart purposes
129
        femguench=fem;
130
131
        % Plot solution
      % postplot(fem, ...
132
133
                    'tridata',{'T','cont','internal'}, ...
        $
134
                    'trimap','jet(1024)', ...
        *
135
        *
                    'solnum',601, ...
                    'title','Time=60
                                         Surface: Temperature', ...
136
        *
137
        $
                    'refine',3, ...
                    'axis',[-0.088128699538072,0.100628699724337,-0.0050000000745058
138
        $
139
```

You can run the program









Thanks for your attention

And a special thanks to my colleagues for their councils : Philippe Le MASSON and Tahar LOULOU

Thanks to Comsol support for their help today

Have a nice Eurotherm Winter School

