**Calcul de l'état adjoint discret dans le cas du schéma de Crank-Nicholson.**

Les données sont des mesures placées dans des fichiers [1] Text.txt, Tint.txt, Qres.txt, Qs.txt de tailles (communes) par exemple $720 \times 2$ (720 lignes, 2 colonnes : la première contenant les temps en secondes, la deuxième contenant les températures extérieures ou intérieures (fichiers Text.txt et Tint.txt) ou des énergies ou puissances (fichiers Qres.txt et Qs.txt)) et une première estimation des paramètres cres, cs, ri, r0, rf (9e+6, 9e+7, 1e-2, 1e-2, 1e-2). Les fichiers contenant ces données peuvent être lues et placées dans des tableaux adaptés au moyen des instructions matlab :

Text=importdata('Text.txt','\t') ; Tint=importdata('Tint.txt','\t') ;
Qres=importdata('Qres.txt','\t') ; Qs=importdata('Qs.txt','\t') ;

Pour simplifier et avoir une formulation mathématique, on notera $t$ le temps (qui varie de 0 =Text(1,1) à $tmax$ =Text(end,1), le pas d'échantillonnage étant $\Delta t$ = Text(2,1)=Text(end,1)/(size(Text,1)-1)), $T_{ext}(t)$,$T_{int}(t)$ les températures extérieure et intérieure et $Q_{res}(t)$, $Q_s(t)$ les autres données. On notera également $\boldsymbol{m}$ les 5 paramètres $c_{res}$, $c_s$, $r_i$, $r_0$, $r_f$.

À partir de ces données on introduit la matrice $2 \times 2$ $\boldsymbol{A}$ et la fonction vectorielle à 2 composantes $\boldsymbol{G}(t)$ définies par :

$$\boldsymbol{A} = \boldsymbol{A}(\boldsymbol{m}) = \begin{pmatrix} -\frac{1}{c_{ref}}(\frac{1}{r_i} + \frac{1}{r_f}) & \frac{1}{c_{res}r_i} \\ \frac{1}{c_s r_i} & -\frac{1}{c_s}(\frac{1}{r_i} + \frac{1}{r_0}) \end{pmatrix},$$

$$\boldsymbol{G}(t) = \boldsymbol{G}(t, \boldsymbol{m}) = \begin{pmatrix} \frac{Q_{res}(t)}{c_{res}} + \frac{T_{ext}(t)}{c_{res}r_f} \\ \frac{Q_s(t)}{c_s} + \frac{T_{ext}(t)}{c_s r_0} \end{pmatrix} = \begin{pmatrix} \frac{Q_{res}(t)}{c_{res}} \\ \frac{Q_s(t)}{c_s} \end{pmatrix} + \begin{pmatrix} \frac{T_{ext}(t)}{c_{res}r_f} \\ \frac{T_{ext}(t)}{c_s r_0} \end{pmatrix}.$$

À noter qu'avec matlab il possible de "fabriquer" une fonction $\boldsymbol{G}(t, \boldsymbol{m})$ par interpolation en procédant par exemple comme suit (cela peut être intéressant pour faire des manipulations faisant abstraction du nombre de données ... bien sûr cela a des limites ...). En plus du tableau (de dimension $720 \times 1$) :

Tempsref=Text( :,1) ;

on définit des tableaux (de dimension $2 \times 720$ dans notre cas) :

Gtref=[Text( :,2)'/cres/rf ; Text( :,2)'/cs/r0] ; Gref=[Qres( :,2)'/cres ; Qs( :,2)'/cs]+Gtref ;

alors l'instruction g=(interp1(Tempsref,Gref',t,'pchip'))' ;

calcule la valeur en t de la fonction $(G)$ interpolation des données (en gros, l'option pchip est intéressante pour respecter les courbures) (et t peut être un vecteur). Attention ceci peut ralentir les calculs, donc il n'est pas sûr qu'il faille conserver ce détail.

Alors, exprimé avec des données non discrètes, le problème consiste à estimer $\boldsymbol{m}$(c'est-à-dire $c_{res}$, $c_s$, $r_i$, $r_0$, $r_f$) en minimisant $J = \frac{1}{tmax}\int_0^{tmax}(T_i(t) - T_{int}(t))^2 dt$ où la fonction vectorielle $\boldsymbol{T}(t) = \begin{pmatrix} T_i(t) \\ T_s(t) \end{pmatrix}$ est solution du système différentiel linéaire :

$$\frac{d\boldsymbol{T}}{dt}(t) = \boldsymbol{A}\boldsymbol{T}(t) + \boldsymbol{G}(t, \boldsymbol{m}),\ 0 \leq t \leq tmax,\ T(0) = \begin{pmatrix} \mathsf{Tint}(1,2) \\ \mathsf{Tint}(1,2) \end{pmatrix},$$

J'ai fait les calculs avec ce choix de $T(0)$ mais je me demande s'il ne vaudrait pas mieux $T(0) = \begin{pmatrix} \mathsf{Tint}(1,2) \\ \mathsf{Text}(1,2) \end{pmatrix}$. Néanmoins je vais continuer avec ce choix.

Je n'expose pas ici la méthode de l'état adjoint en continu et on passe directement à la discrétisation du problème en utilisant le schéma de Crank-Nicholson sur le système différentiel. Il faut dire qu'il est important de calculer l'état adjoint sur le système discrétisé car en général la discrétisation de l'état adjoint continu n'est pas égal à l'état adjoint de système discrétisé.

Avec $N = \mathsf{size}(\mathsf{Tempsref}, 1) - 1$ ("nombre d'intervalles" ; il y a donc $N + 1$ valeurs de $\boldsymbol{T}$ à calculer (nombre d'extrémités")) et $\Delta t = \mathsf{Tempsref}(\mathsf{end})/N = \mathsf{Tempsref}(N+1)/N$, la discrétisation du système différentiel par le schéma de Crank-Nicholson conduit aux relations de récurrence (croissantes en $n$) :

$$\begin{aligned} \frac{\boldsymbol{T}^{(n+1)} - \boldsymbol{T}^{(n)}}{\Delta t} &= \boldsymbol{A}\frac{\boldsymbol{T}^{(n+1)} + \boldsymbol{T}^{(n)}}{2} + \frac{1}{2}(\boldsymbol{G}(n\Delta t, \boldsymbol{m}) + \boldsymbol{G}((n-1)\Delta t, \boldsymbol{m})),\ 1 \leq n \leq N, \\ \boldsymbol{T}^{(1)} &= \begin{pmatrix} \mathsf{Tint}(1,1) \\ \mathsf{Tint}(1,1) \end{pmatrix}. \end{aligned}$$

On peut réécrire cette relation sous la forme (il faut comprendre que lorsqu'on calcule $T^{(N+1)}$, la valeur en la dernière extrémité, "le" $t$ correspondant est $N\Delta t = \mathsf{Tempsref}(\mathsf{end})$, donc le dernier $G$ est bien $G(N\Delta t)$) :

$$T^{(n+1)} = (I - \frac{\Delta t}{2}A)^{-1}(I + \frac{\Delta t}{2}A)T^{(n)} + \frac{\Delta t}{2}(I - \frac{\Delta t}{2}A)^{-1}(\boldsymbol{G}(n\Delta t, \boldsymbol{m}) + \boldsymbol{G}((n-1)\Delta t, \boldsymbol{m})),\ 1 \leq n \leq N,$$

1. Je les ai fabriquées à partir des données brutes avec LibreOffice

bien sûr avec $\boldsymbol{T}^{(k)} = \begin{pmatrix} T_i^{(k)} \\ T_s^{(k)} \end{pmatrix}$, $1 \le k \le N+1$.

Il faut minimiser :

$$J = \frac{1}{2(N+1)} \sum_{n=1}^{n=N+1} (T_i^{(n)} - \mathsf{Tint(n,2)})^2.$$

Puisque le premier terme de la somme est 0, le code matlab suivant calcule bien les $T^{(k)}$, $1 \le k \le N+1$ et f valeur de $J$.

```
A=[-(1/ri+1/rf)/cres, 1/ri/cres ; 1/ri/cs, -(1/ri+1/r0)/cs] ;
N=size(Tempsref,1)-1 ; deltaT=Tempsref(720)/N ;
B=inv(eye(2)-deltaT/2*A) ; C=B*(eye(2)+deltaT/2*A) ; B=deltaT/2*B ;
T0=[Tint(1,2) ;Tint(1,2)] ; T=zeros(2,N+1) ; T=T0 ;
t0=0 ; t=zeros(1,N+1) ; t=t0 ;
G0=G(t0) ;
f=0 ;
for i=1 :N
    t1=t0+deltaT ; G1=G(t1) ;
    T1=C*T0+B*(G0+G1) ;
    f=f+(T1(1)-Tint(i+1,2))^2 ;
    t=[t,t1] ; T=[T,T1] ;
    t0=t1 ; G0=G1 ; T0=T1 ;
end
f=f/2/(N+1) ;
```

Pour calculer le gradient de $J$ par rapport aux paramètres dont dépendent la matrice $\boldsymbol{A}$ et la fonction $\boldsymbol{G}$ on écrit la relation de récurrence satisfaite par la variation $\delta\boldsymbol{T}^{(n)}$ de $\boldsymbol{T}^{(n)}$ qui résulte de variations $\delta\boldsymbol{A}$ et $\delta\boldsymbol{G}(n\Delta t, \boldsymbol{m})$ de $\boldsymbol{A}$ et $\boldsymbol{G}$ :

$$\begin{aligned} \frac{\delta\boldsymbol{T}^{(n+1)} - \delta\boldsymbol{T}^{(n)}}{\Delta t} &= \boldsymbol{A}\frac{\delta\boldsymbol{T}^{(n+1)} + \delta\boldsymbol{T}^{(n)}}{2} \\ &+\delta\boldsymbol{A}\frac{\boldsymbol{T}^{(n+1)} + \boldsymbol{T}^{(n)}}{2} + \frac{1}{2}(\delta\boldsymbol{G}(n\Delta t, \boldsymbol{m}) + \delta\boldsymbol{G}((n-1)\Delta t, \boldsymbol{m})),\ 1 \le n \le N, \\ \delta\boldsymbol{T}^{(1)} &= 0, \end{aligned}$$

étant entendu que $J$ varie de :

$$\delta J = \frac{1}{N+1} \sum_{n=1}^{n=N+1} \delta T_i^{(n)} (T_i^{(n)} - \mathsf{Tint(n,2)}) = \frac{1}{N+1} \sum_{n=2}^{n=N+1} \delta\boldsymbol{T}^{(n)T} \begin{pmatrix} T_i^{(n)} - \mathsf{Tint(n,2)} \\ 0 \end{pmatrix},$$

et on fait le produit scalaire de ces relations avec une suite $\boldsymbol{S}^{(n)}$ de vecteurs $\mathbb{R}^2$ qui sera précisée plus loin (c'est là qu'interviendront les résidus $\begin{pmatrix} T_i^{(n)} - \mathsf{Tint(n,2)} \\ 0 \end{pmatrix}$ qui seront "rétropropagés"), et on somme :

$$\begin{aligned} \sum_{n=1}^{N} \boldsymbol{S}^{(n)T} \frac{\delta\boldsymbol{T}^{(n+1)} - \delta\boldsymbol{T}^{(n)}}{\Delta t} &= \sum_{n=1}^{N} \boldsymbol{S}^{(n)T} \boldsymbol{A} \frac{\delta\boldsymbol{T}^{(n+1)} + \delta\boldsymbol{T}^{(n)}}{2} \\ &+ \sum_{n=1}^{N} \boldsymbol{S}^{(n)^T} (\delta\boldsymbol{A}\frac{\boldsymbol{T}^{(n+1)} + \boldsymbol{T}^{(n)}}{2} + \frac{1}{2}(\delta\boldsymbol{G}(n\Delta t, \boldsymbol{m}) + \delta\boldsymbol{G}((n-1)\Delta t, \boldsymbol{m}))), \end{aligned}$$

mais, en convenant de $\boldsymbol{S}^{(N+1)} = 0$, et en tenant compte de $\delta\boldsymbol{T}^{(1)} = 0$ :

$$\sum_{n=1}^{N} \boldsymbol{S}^{(n)T} \frac{\delta\boldsymbol{T}^{(n+1)} - \delta\boldsymbol{T}^{(n)}}{\Delta t} = \sum_{n=2}^{N+1} \boldsymbol{S}^{(n-1)T} \frac{\delta\boldsymbol{T}^{(n)}}{\Delta t} - \sum_{n=1}^{N} \boldsymbol{S}^{(n)T} \frac{\delta\boldsymbol{T}^{(n)}}{\Delta t} = \sum_{n=2}^{N+1} \delta\boldsymbol{T}^{(n)T} \frac{\boldsymbol{S}^{(n-1)} - \boldsymbol{S}^{(n)}}{\Delta t}$$

et :

$$\sum_{n=1}^{N} \boldsymbol{S}^{(n)T} \boldsymbol{A} \frac{\delta\boldsymbol{T}^{(n+1)} + \delta\boldsymbol{T}^{(n)}}{2} = \sum_{n=2}^{N+1} \boldsymbol{S}^{(n-1)T} \boldsymbol{A} \frac{\delta\boldsymbol{T}^{(n)}}{2} + \sum_{n=1}^{N} \boldsymbol{S}^{(n)T} \boldsymbol{A} \frac{\delta\boldsymbol{T}^{(n)}}{2} = \sum_{n=2}^{N+1} \delta\boldsymbol{T}^{(n)T} \boldsymbol{A}^T \frac{\boldsymbol{S}^{(n-1)} + \boldsymbol{S}^{(n)}}{2},$$

on trouve :

$$\sum_{n=2}^{N+1} \delta \boldsymbol{T}^{(n)T} \frac{\boldsymbol{S}^{(n-1)} - \boldsymbol{S}^{(n)}}{\Delta t} = \sum_{n=2}^{N+1} \delta \boldsymbol{T}^{(n)T} \boldsymbol{A}^T \frac{\boldsymbol{S}^{(n-1)} + \boldsymbol{S}^{(n)}}{2}$$
$$+ \sum_{n=1}^{N} \boldsymbol{S}^{(n)^T} (\delta \boldsymbol{A} \frac{\boldsymbol{T}^{(n+1)} + \boldsymbol{T}^{(n)}}{2} + \frac{1}{2}(\delta \boldsymbol{G}(n\Delta t, \boldsymbol{m}) + \delta \boldsymbol{G}((n-1)\Delta t, \boldsymbol{m})))$$

Maintenant on voit apparaître l'équation (adjointe) que doit satisfaire $\boldsymbol{S}^{(n)}$. En effet si :

$$\frac{\boldsymbol{S}^{(n-1)} - \boldsymbol{S}^{(n)}}{\Delta t} - \boldsymbol{A}^T \frac{\boldsymbol{S}^{(n-1)} + \boldsymbol{S}^{(n)}}{2} = \frac{1}{N+1} \begin{pmatrix} T_i^{(n)} - \mathsf{Tint(n,2)} \\ 0 \end{pmatrix} \text{ pour } N+1 \geq n \geq 2 \text{ avec } \boldsymbol{S}^{N+1} = 0$$

il est clair que :

$$\delta J = \sum_{n=1}^{N} \boldsymbol{S}^{(n)^T} (\delta \boldsymbol{A} \frac{\boldsymbol{T}^{(n+1)} + \boldsymbol{T}^{(n)}}{2} + \frac{1}{2}(\delta \boldsymbol{G}(n\Delta t, \boldsymbol{m}) + \delta \boldsymbol{G}((n-1)\Delta t, \boldsymbol{m}))),$$

et comme par exemple $\frac{\partial \boldsymbol{A}}{\partial c_{res}} = \begin{pmatrix} \frac{1}{c_{res}^2}(\frac{1}{r_i} + \frac{1}{r_f}) & \frac{-1}{c_{res}^2 r_i} \\ 0 & 0 \end{pmatrix}$ et $\frac{\partial \boldsymbol{G}}{\partial c_{res}}(n\Delta t) = \begin{pmatrix} -\frac{Q_{res}(n\Delta t)}{c_{res}^2} - \frac{T_{ext}(n\Delta t)}{c_{res}^2 r_f} \\ 0 \end{pmatrix}$ (la première composante de $\frac{\partial \boldsymbol{G}}{\partial c_{res}}(n\Delta t)$ est simplement l'opposée de la première composante de $\boldsymbol{G}(n\Delta t, \boldsymbol{m})$ divisée par $c_{res}$) on a par exemple :

$$\frac{\partial J}{\partial c_{res}} = \sum_{n=1}^{N} \boldsymbol{S}^{(n)^T} (\frac{\partial \boldsymbol{A}}{\partial c_{res}} \frac{\boldsymbol{T}^{(n+1)} + \boldsymbol{T}^{(n)}}{2} + \frac{1}{2}(\frac{\partial \boldsymbol{G}}{\partial c_{res}}(n\Delta t, \boldsymbol{m}) + \frac{\partial \boldsymbol{G}}{\partial c_{res}}((n-1)\Delta t, \boldsymbol{m}))).$$

où $\boldsymbol{S}^{(n)}$ est solution de la relation de récurrence (décroissante en $n$) :

$$\frac{\boldsymbol{S}^{(n-1)} - \boldsymbol{S}^{(n)}}{\Delta t} = \boldsymbol{A}^T \frac{\boldsymbol{S}^{(n-1)} + \boldsymbol{S}^{(n)}}{2} + \frac{1}{N+1} \begin{pmatrix} T_i^{(n)} - \mathsf{Tint(n)} \\ 0 \end{pmatrix} \text{ pour } N+1 \geq n \geq 2 \text{ avec } \boldsymbol{S}^{(N+1)} = 0$$

(autres détails plus tard)

À noter que ce calcul de deux suites $\boldsymbol{T}^{(n)}$ et $\boldsymbol{S}^{(n)}$ fonctionne aussi pour un modèle plus compliqué avec plus de R et C.

Il faut noter aussi que la relation de récurrence rétrograde satisfaite par $\boldsymbol{S}^{(n)}$ est donc, $\boldsymbol{I}$ étant la matrice identité $2 \times 2$ :

$$\boldsymbol{S}^{(n-1)} = (I - \frac{\Delta t}{2}\boldsymbol{A}^T)^{-1}(I + \frac{\Delta t}{2}\boldsymbol{A}^T)\boldsymbol{S}^{(n)} + (I - \frac{\Delta t}{2}\boldsymbol{A}^T)^{-1} \frac{\Delta t}{N+1} \begin{pmatrix} T_i^{(n)} - \mathsf{Tint(n)} \\ 0 \end{pmatrix}, \; N+1 \geq n \geq 2, \; \boldsymbol{S}^{(N+1)} = 0,$$

ou encore en transposant et en tenant compte du fait que la transposée de l'inverse est égale à l'inverse de la transposée et la transposée d'un produit est égale au produit des transposés dans l'ordre inverse :

$$\boldsymbol{S}^{T(n-1)} = \boldsymbol{S}^{T(n)}(\boldsymbol{I} + \frac{\Delta t}{2}\boldsymbol{A})(\boldsymbol{I} - \frac{\Delta t}{2}\boldsymbol{A})^{-1} + \frac{\Delta t}{N+1} \begin{pmatrix} T_i^{(n)} - \mathsf{Tint(n)} & 0 \end{pmatrix} (\boldsymbol{I} - \frac{\Delta t}{2}\boldsymbol{A})^{-1}, \; N+1 \geq n \geq 2, \; \boldsymbol{S}^{T(N+1)} = 0.$$

Se reportant à la relation de récurrence satisfaite par les $\boldsymbol{T}^{(n)}$ on voit qu'on retrouve la mêmes matrices (à un facteur 2 prés pour le second terme) car :

$$(\boldsymbol{I} + \frac{\Delta t}{2}\boldsymbol{A})(\boldsymbol{I} - \frac{\Delta t}{2}\boldsymbol{A})^{-1} = (\boldsymbol{I} - \frac{\Delta t}{2}A)^{-1}(\boldsymbol{I} + \frac{\Delta t}{2}A)$$

(vu que pour toute matrice $D$ telle que les inverses suivants existent, $(I + \boldsymbol{D})(\boldsymbol{I} - \boldsymbol{D})^{-1} = (\boldsymbol{I} - \boldsymbol{D})^{-1}(\boldsymbol{I} + \boldsymbol{D})$ équivaut à $(\boldsymbol{I} - \boldsymbol{D})(\boldsymbol{I} + \boldsymbol{D})(\boldsymbol{I} - \boldsymbol{D})^{-1} = (\boldsymbol{I} + \boldsymbol{D})$ ou encore à $(\boldsymbol{I} - \boldsymbol{D})(\boldsymbol{I} + \boldsymbol{D}) = (\boldsymbol{I} + \boldsymbol{D})(\boldsymbol{I} - \boldsymbol{D})$ ce qui est vrai et vaut $\boldsymbol{I} - \boldsymbol{D}^2$... je note ceci car en général le produit matriciel n'est pas commutatif ...). Donc le code pour calculer l'état adjoint peut être :

```
S1=[0,0] ; S=zeros(size(Tempsref,1),2) ; S=S1 ;
for i=N+1 :-1 :2
    S0=S1*C+[T(1,i)-Tint(i,2), 0]*B ;
    S=[S0 ;S] ; S1=S0 ;
end
S=2*S/(N+1) ;
```