

Déplétion d'un réservoir pétrolier

Luc DORMIEUX et Alexandre ERN

October 10, 2017

Contents

1	problématique	1
2	Discrétisation en temps	2
3	Discrétisation en espace	3
4	Formulation du schéma numérique	3
5	Implémentation sous Scilab	4
6	Simulations numériques	5
7	Amélioration des performances numériques	6
7.1	Préfactorisation des inverses matricielles	6
7.2	Mise à profit des structures creuses des matrices	6

L'objet de ce TD est de simuler numériquement la déplétion d'un réservoir pétrolier. On se placera dans le cadre de l'hypothèse œdométrique ce qui permettra de modéliser les variations spatiales et temporelles du champ de pression dans le réservoir par une équation de la chaleur. De plus, on se placera en configuration axisymétrique. On réalisera un script Scilab fournissant une solution approchée de l'équation de la chaleur unidimensionnelle en coordonnées radiales. L'approximation spatiale sera réalisée avec un élément fini P_1 conforme et l'approximation temporelle par le schéma d'Euler explicite ou implicite.

1 problématique

Un réservoir pétrolier est une formation rocheuse perméable dont l'espace poreux est partiellement saturé par des hydrocarbures. Au terme d'un mouvement ascendant depuis la zone de formation des hydrocarbures, appelée roche mère, ceux-ci viennent se piéger dans le réservoir en raison de l'imperméabilité des couches limitant supérieurement ce dernier. L'accumulation progressive d'hydrocarbures conduit à l'existence d'une pression dans le

réservoir. On s'intéresse dans ce TD à la séquence de production primaire qui consiste à réaliser un puits dans le réservoir et à tirer parti de la pression initiale qui y règne pour récupérer les hydrocarbures. Il suffit pour cela que la pression de puits p_p soit plus basse que la pression de réservoir.

Pour simplifier, on suppose que le réservoir est un cylindre horizontal plat \mathcal{R} de rayon L entouré de formations imperméables. L'ordre de grandeur du rayon de puits r_p est décimétrique, tandis que celui de L est de l'ordre de plusieurs centaines de mètres. On suppose que les hydrocarbures saturant totalement l'espace poreux et que la pression initiale d'hydrocarbure est uniforme. Elle est notée p_0 . La roche constituant le réservoir est modélisée comme un matériau poroélastique linéaire (modules d'élasticité λ_o , μ_o coefficient de Biot b , perméabilité k).

Une fois le puits réalisé, on suppose que l'évolution du réservoir est régie par l'hypothèse oedométrique :

- σ_{11} uniforme et constant;
- $\underline{\xi} = \xi(x_1)\underline{e}_1$

où \underline{e}_1 désigne la direction verticale. Dans ces conditions, la pression ne dépend que du rayon r compté à partir de l'axe du puits et du temps t .

Le réservoir étant initialement à la pression uniforme p_0 , on impose à partir de l'instant $t = 0$ une pression de puits p_p nulle en $r = r_p \approx 0$.

Question 1 *Montrer qu'avec les hypothèses envisagées, le champ de pression $p(r, t)$ dans le réservoir est solution de*

$$\begin{cases} \partial_t p - c \frac{1}{r} \partial_r (r \partial_r p) = 0 & (r, t) \in]0, L[\times]0, +\infty[\\ p(0, t) = 0 & t \in]0, +\infty[\\ \partial_r p(L, t) = 0 & t \in]0, +\infty[\\ p(r, 0) = p_0 & r \in]0, L[\end{cases} \quad (1)$$

avec

$$c = k \left(\frac{b^2}{\lambda_o + 2\mu_o} + \frac{1}{M} \right)^{-1}.$$

On se propose de quantifier la séquence de production primaire à l'aide de l'indice de récupération $I(t)$ défini comme la quantité d'hydrocarbures (supposés ici incompressibles) produite entre l'instant où débute la production et l'instant t , rapportée à la quantité d'hydrocarbures initialement présente dans le réservoir. On a donc

$$I(t) = \frac{\int_{\mathcal{R}} (\phi_o - \phi(r, t)) dV}{\int_{\mathcal{R}} \phi_o dV}$$

où $\phi(r, t)$ et ϕ_o désignent respectivement les porosités lagrangiennes actuelle et initiale.

Question 2 Montrer que l'indice de récupération asymptotique est

$$I^\infty = \lim_{t \rightarrow \infty} I = \frac{p_0}{\phi_o} \left(\frac{b^2}{\lambda_o + 2\mu_o} + \frac{1}{M} \right)$$

On pose alors $Q(t) = I(t)/I^\infty$. Montrer que

$$Q(t) = 1 - \frac{2}{L^2} \int_0^L \frac{p(r, t)}{p_0} r dr.$$

2 Discrétisation en temps

On se donne un pas de temps δt et on pose $t^n = n\delta t$ pour n entier positif. La dérivée temporelle $\partial_t p$ dans (1) est discrétisée à l'aide de différences finies en temps. En notant $p^n(r)$ l'approximation du champ de pression à l'instant discret t^n , on aboutit au problème suivant : étant donné p^n , trouver p^{n+1} solution de

$$\begin{cases} p^{n+1} - p^n - c\delta t \frac{1}{r} \partial_r (r \partial_r p^{n+\alpha}) = 0 & r \in]0, L[\\ p^{n+1}(0) = 0 \\ \partial_r p^{n+1}(L) = 0 \end{cases} \quad (2)$$

avec la condition initiale $p^0 = p_0$. Ici, α vaut 0 pour le schéma d'Euler explicite et 1 pour le schéma d'Euler implicite.

3 Discrétisation en espace

On discrétise maintenant (2) en espace. On se donne un maillage non-uniforme de l'intervalle $[0, L]$ sous la forme

$$0 = r_1 < \dots < r_N = L.$$

Pour $1 \leq i \leq N-1$, on note $h_i = r_{i+1} - r_i$ et pour $1 \leq i \leq N$, on note φ_i la fonction chapeau associée au nœud r_i . On choisit de prendre en compte la condition de Dirichlet en $r = 0$ par une méthode de pénalisation : p^{n+1} est donné par la formulation variationnelle

$$\frac{1}{\varepsilon} p^{n+1}(0) \varphi_i(0) + \int_0^L (p^{n+1} - p^n) \varphi_i r dr + c\delta t \int_0^L (p^{n+\alpha})' \varphi_i' r dr = 0, \quad 1 \leq i \leq N, \quad (3)$$

où ε est le paramètre pour la pénalisation de la condition de Dirichlet. Par la suite, on prendra $\varepsilon = 10^{-12}$.

4 Formulation du schéma numérique

En notant P^n le vecteur dont les composantes sont les $p^n(r_i)$, $1 \leq i \leq N$, montrer que (3) s'écrit sous la forme matricielle

$$\frac{1}{\varepsilon}(P^{n+1}, e_1) + M(P^{n+1} - P^n) + AP^{n+\alpha} = 0, \quad (4)$$

où $e_1 = (1, 0, \dots, 0)$ et où M et A sont des matrices d'ordre N dont on précisera les coefficients.

Question 3 *Montrer que le schéma numérique s'écrit sous la forme matricielle*

$$Z_l P^{n+1} = Z_r P^n, \quad (5)$$

où Z_l et Z_r sont des matrices d'ordre N qu'on évaluera en fonction de M , A et ε pour les schémas explicite et implicite.

Question 4 *Evaluer $Q^n = Q(t^n)$ de façon analytique en fonction des coordonnées du vecteur P^n .*

5 Implémentation sous Scilab

Question 5 *Ecrire un script Scilab mettant en œuvre le schéma (5).*

On s'inspirera des conseils ci-dessous et de l'aide en ligne.

1. terminer toutes les lignes de commande par un point virgule ; (sauf celles pour lesquelles on souhaite afficher dans la fenêtre scilab le résultat numérique de la commande).
2. une ligne de commentaires doit commencer par `//`.
3. initialiser le script par la commande `clear all`; qui nettoie l'espace mémoire occupé.
4. initialisation du maillage : utiliser une fonction

```
function [u] = shapex(t)
    a = ...;
    u = atanh(tanh(a)*t) / a;
endfunction
x = L * shapex(linspace(0,1,nx));
```

Le paramètre a contrôle la finesse des mailles près de $r = 0$.

Question 6 *Pour quelle valeur de a construit-on un maillage uniforme ? Quel est l'effet d'une augmentation de la valeur de a ?*

5. on introduira une variable qu'on pourra appeler `euler`, qui prendra les valeurs 1 et 2 et qui codera pour le schéma explicite ou implicite. La syntaxe est la suivante

```
euler = ...; // 1=explicite, 2=implicite
select euler
  case 1
    ...; // commandes pour le cas explicite
  case 2
    ...; // commandes pour le cas implicite
end
```

6. programmation d'une boucle : par exemple

```
for i = 1:nx
  ...
end
```

7. mise en œuvre du schéma (5) :

```
unew = inv(ZL) * ZR * u;
```

8. animation graphique : avant la boucle en temps, il convient pour réaliser une animation d'initialiser la fenêtre graphique de la façon suivante :

```
driver("X11")
xselect()
xbasc()
xset("pixmap",1)
```

La commande `driver("X11")` permet d'accélérer l'affichage en ne stockant pas en mémoire toutes les données relatives au tracé de la courbe mais seulement les plus importantes (perte de la possibilité du zoom par exemple). La commande `xset("pixmap",1)` évitera le clignotement de la fenêtre lors de la boucle en temps. Au sein de la boucle en temps, la fenêtre graphique est rafraîchie de la façon suivante :

```
xset("wwpc")
plot2d(temps(1,1:l+1),qqq(1,1:l+1),rect=[0,0,Tmax,1]);
xset("wshow")
```

La commande `xset("wwpc")` efface le contenu courant de la fenêtre graphique en mémoire. La commande `xset("wshow")` affiche dans la fenêtre graphique le contenu courant de la mémoire. `l` est l'indice courant de la boucle en temps, `temps` est le vecteur contenant les valeurs des instants discrets t^m pour $0 \leq m \leq l$ et `qqq(m)` vaut Q^m .

9. afin de mesurer le temps d'exécution du script, on insérera la commande

```
timer();
```

avant le démarrage de la boucle en temps et la commande

```
timer()
```

à la fin de celle-ci. Le temps d'exécution écoulé entre les 2 appels de la fonction `timer` sera affiché dans la fenêtre Scilab.

10. on affichera également le temps t^m au bout duquel l'indice de récupération atteint 90% de sa valeur asymptotique.

6 Simulations numériques

Question 7 *Montrer que la dépendance en temps de la pression et de l'indice de récupération est contrôlée par un temps adimensionnel.*

Evaluer l'échelle de temps pour les valeurs numériques suivantes (appropriées pour un réservoir de grès) : viscosité du fluide 10^{-3} Pa.s, perméabilité intrinsèque 10^{-14} m², $\lambda_o = 1.5$ GPa, $\mu_o = 0.75$ GPa, $M = 50$ GPa, $b = 0.9$ et L est de l'ordre de 100 à 500m.

Notre objectif est de réaliser des simulations numériques afin de déterminer la valeur du temps adimensionnel pour laquelle on atteint 90% de l'huile susceptible d'être produite. On notera $t_{90\%}$ ce temps. Pour les simulations numériques, on prendra $L = 1$ et $c = 1$, la dimensionnalisation de $t_{90\%}$ se faisant via l'échelle de temps calculée ci-dessus.

Considérer d'abord un maillage uniforme avec $N = 10$ et un temps de simulation de $T = 10$.

Question 8 *Vérifier que le schéma explicite n'est stable que si le pas de temps est suffisamment petit. Estimer numériquement le seuil de stabilité. Qu'en est-il pour le schéma implicite ? Commenter l'efficacité relative des deux schémas.*

Par la suite, on ne considérera que le schéma implicite.

Question 9 *Faire un premier calcul de $t_{90\%}$ sur maillage uniforme ($N = 10$) et avec $\delta t = 0.1$. Comparer par rapport à la valeur obtenue avec un calcul très précis ($N = 1000$ et $\delta t = 0.01$) qui est de $t_{90\%} \simeq 9.7$. Etudier les améliorations obtenues en augmentant N et en travaillant sur maillage non-uniforme (on pourra par exemple réaliser une simulation avec $N = 100$ et $a = 1$ puis $a = 4$). Quelle est approximativement la valeur maximale de N que l'on peut considérer sans saturer complètement l'espace mémoire Scilab ? Commenter par rapport à l'estimation de $t_{90\%}$.*

7 Amélioration des performances numériques

Deux pistes seront explorées.

7.1 Préfactorisation des inverses matricielles

On pose

$$Z = Z_l^{-1} Z_r.$$

Question 10 Expliquer pourquoi il est intéressant de précalculer la matrice Z avant de lancer la boucle en temps. Faire quelques essais numériques afin de quantifier les gains en temps de calcul.

7.2 Mise à profit des structures creuses des matrices

L'assemblage d'une matrice A tenant compte de sa structure creuse (ici tridiagonale) se fait de la manière suivante

```
ij = [];  
v = [];  
for i = 1:nx-1  
    ij = [ij; i,i]; v = [v, \alpha_{i,i}];  
    ij = [ij; i+1,i]; v = [v, \alpha_{i+1,i}];  
    ij = [ij; i,i+1]; v = [v, \alpha_{i,i+1}];  
    ij = [ij; i+1,i+1]; v = [v, \alpha_{i+1,i+1}];  
end
```

où α_{ij} est la contribution de la maille $[r_i, r_{i+1}]$ au terme d'indice (i, j) dans la matrice A . Le résultat de cette opération sont deux vecteurs, ij qui contient la liste de tous les couples d'indices (i, j) pour lesquels le coefficient A_{ij} est non nul et v qui contient les valeurs correspondantes de A_{ij} . Noter que si un couple (i, j) apparaît deux fois dans ij , les valeurs correspondantes dans v seront cumulées.

L'assemblage de la matrice A se fait simplement en utilisant la commande

```
A = sparse(ij,v);
```

L'inversion d'une matrice creuse se fait en deux étapes :

- $H = \text{lufact}(A)$; qui produit une matrice H contenant les facteurs de la décomposition LU de A ;
- $y = \text{lusolve}(A,x)$; qui calcule $y = A^{-1}x$.

Question 11 Créer un nouveau script Scilab utilisant les fonctions décrites ci-dessus. Comparer les performances numériques par rapport à l'approche ne tirant pas profit des structures creuses, notamment dans le cas de maillages fins en espace.