

La simulation numérique 2/2

Maintenant que nous avons expliqué dans le précédent podcast les besoins de simulation, les problématiques rencontrées avec différents champs de recherche et la première simulation numérique de l'histoire (finir par le hasard et les probabilités et toutes ces choses que je déteste personnellement a été dur pour moi) nous allons retourner sur des équations déterministes.

Rappel sur les types de simulation

Je ne l'avais pas forcément expliqué, mais il y a plusieurs types de simulation numériques possibles ¹ :

- La simulation discrète dans laquelle le système est soumis à une succession d'événements qui le modifient. Ces simulations ont vocation à appliquer des principes simples à des systèmes de grande taille. La simulation discrète se divise en deux grandes catégories :
 - asynchrone ou time-slicing : on simule à chaque fois le passage d'une unité de temps sur tout le système. Ce terme n'est généralement plus utilisé dans le domaine professionnel depuis l'apparition croissante des nouvelles technologies.
 - synchrone ou event-sequencing : on calcule l'arrivée du prochain événement, et on ne simule qu'événement par événement, ce qui permet souvent des simulations rapides, bien qu'un peu plus complexes à programmer.
- La simulation par agents, où la simulation est segmentée en différentes entités qui interagissent entre elles. Elle est surtout utilisée dans les simulations économiques et sociales, où chaque agent représente un individu ou un groupe d'individus. Par nature, son fonctionnement est asynchrone.
- La simulation continue, où le système se présente sous la forme d'équations différentielles à résoudre.

Dans la suite de ce podcast, je vais sûrement me pencher sur la simulation continue.

Il existe d'ailleurs différentes méthodes de simulation, avec par exemple les méthodes de monte-carlo comme nous l'avons vu dans le précédent podcast, mais aussi les simulation atomistiques (pour certaines appelées ab initio) dont un exemple serait l'étude de l'eau avec un travail au niveau des atomes, puis des molécules, puis de la masse globale ou encore celles dont je vais parler qui sont les méthodes de discrétisation.

¹http://fr.wikipedia.org/wiki/Simulation_informatique

Pourquoi la discrétisation ?

L'idée est en fait que la résolution analytique des équations de phénomènes physiques est très complexe et en dehors de cas particuliers ou alors avec des simplifications importantes impossible pour certaines équations comme par exemple celle décrivant des fluides : l'équation de Navier-Stokes qui possède de manière générale des termes non-linéaire.

Comme expliqué sur Wikipédia ² : On dit qu'un système de type entrée-sortie est linéaire ou relève du principe de superposition si:

- à la somme de deux entrées quelconques correspond la somme des deux sorties correspondantes,
- à un multiple d'une entrée quelconque correspond le même multiple de la sortie correspondante.

Dans le domaine des systèmes physiques et mécaniques, on appelle souvent l'entrée *excitation* et la sortie *réponse*. Plus précisément, si l'on note les excitations f (par référence aux forces en mécanique) et les réponses x (par référence aux mouvements générés par les forces) :

- Lorsque l'on sollicite le système par une entrée (excitation) f_1 , la réponse (déplacement) est x_1 ;
- Lorsque l'on sollicite le système par une entrée (excitation) f_2 , la réponse (déplacement) est x_2 ;

alors le système est dit linéaire si et seulement si pour λ_1 et λ_2 deux nombres quelconques, la réponse à l'excitation $\lambda_1.f_1 + \lambda_2.f_2$ est $\lambda_1.x_1 + \lambda_2.x_2$.

Je vais présenter un peu en détail la plus simple des trois grandes méthodes de discrétisation (Les différences finies) et j'expliquerais le principe des deux autres (Les éléments finis, Les volumes finis).

Chacune possède une origine spécifique et a pour rôle de permettre la transformation d'un problème basé sur des équations différentielles ou aux dérivés partielles donc plutôt de l'analyse vers un problème de résolution de système linéaire de type :

$$A.x = b$$

Où A est la matrice qui représente le problème, x la solution à trouver et b qui représente les conditions initiales/aux limites qui est un problème d'algèbre linéaire.

Le travail que l'on fait finalement quand on part de zéro est donc le suivant :

²http://fr.wikipedia.org/wiki/Principe_de_superposition

1. On définit les équations du modèle qui représente le phénomène que l'on souhaite étudier avec ses conditions aux limites
2. On assure que la méthode de discrétisation est capable de fournir une solution unique (toute la branche mathématique de l'analyse numérique avec des gens comme Schwartz, Cauchy, Lipschitz ou encore Jacques-Louis Lions)
3. On ramène le problème décrit par un modèle basé sur des équations différentielles ou à dérivées partielles à un problème d'algèbre linéaire
4. On choisit un algorithme qui possède les bonnes propriétés pour la résolution de mon système linéaire
5. On implémente cet algorithme
6. On l'exécute (avec ou sans parallélisation)

Définition du domaine

On a déjà parlé dans le premier podcast de ce que l'on entend par modèle, un point qu'il est important de préciser c'est qu'il faut expliquer sur quel domaine ce modèle va s'appliquer (on peut voir en physique par exemple que les théories de la relativité générale et celle de la mécanique quantique par exemple n'ont pas le même domaine d'application) et avec quelles conditions aux limites/initiales.

Assez souvent on va donc donner les équations qui s'appliqueront sur le domaine et on parlera des conditions aux limites pour dire ce qu'il se passe sur les "bords" :

- une poutre est fixée à un mur
- un flux de liquide est présent sur l'entrée d'un tuyau, etc

Ces conditions aux limites vont être ainsi de plusieurs types ³ :

- *Les conditions aux limites en temps* : assez souvent on va imposer des conditions à $t = 0$ qui vont par exemple être les conditions initiales d'une simulation en météorologie. On peut aussi imposer des conditions à $t = +\infty$, mais je n'ai pas trouvé d'exemple assez parlant de l'usage de telles conditions aux limites.
- *Les conditions aux limites en espace* : il y en a de différents types, voici les plus classiques :
 - Condition aux limites de Dirichlet⁴ (nommée d'après Johan Dirichlet, 1805-1859, mathématicien allemand ayant notamment travaillé sur les séries de Fourier, l'arithmétique et on lui doit l'essentiel de la démonstration du dernier théorème de Fermat pour le cas où l'exposant

³http://fr.wikipedia.org/wiki/Condition_aux_limites

⁴http://fr.wikipedia.org/wiki/Condition_aux_limites_de_Dirichlet

est égal à 5⁵) : on spécifie la valeur que va prendre la solution des équations en certaines frontières ou limites du domaine étudié (pour un intervalle $[a, b]$ on aura donc $y(a) = \alpha$ et $y(b) = \beta$). Un exemple est la valeur de la température en a pour l'équation de la chaleur.

- Conditions aux limites de Neumann⁶ (nommée d'après Carl Neumann⁷ et pas John Van Neumann, 1832-1925 et il travailla notamment sur les équations intégrales dont l'une des indéterminées est une intégrale et dont les équations de Maxwell sont l'un des plus célèbres représentants⁸) : on spécifie la valeur que va prendre la dérivée de la solution des équations en certaines frontières du domaine (pour un intervalle $[a, b]$ on aura donc $y'(a) = \alpha$ et $y'(b) = \beta$). L'exemple est le flux de température, toujours pour l'équation de la chaleur.
- Condition aux limites de Robin⁹ (nommée d'après Victor Gustave Robin, 1855-1897, et non pas notre Robin national, qui a notamment travaillé sur des problèmes de thermodynamique) : ici il s'agit d'imposer aux limites du domaines une relation linéaire entre les valeurs de la fonction et celle de sa dérivée (pour un intervalle $[a, b]$, on aura donc $\alpha.y(a) - \beta.y'(a) = g(a)$ et $\alpha.y(b) - \beta.y'(b) = g(b)$ avec cette fois, α , β et g des fonctions). L'exemple de ce type de condition est un peu plus complexe mais si on reprend notre exemple d'équation de la chaleur, si un bord est en contact avec un autre milieu, les lois de Newton et de Fourier précise que le flux de température à cette interface est proportionnel à la différence de température, mais aussi à la valeur du gradient¹⁰.

A côté de ces conditions aux limites en espace, on en a encore qui sont plus siouuses avec par exemples des conditions aux limites dynamiques qui ressemblent aux conditions de Robin, mais qui évoluent au cours du temps et fonction des points de la frontière¹¹.

Différences finies

La méthode des différences finies apparaît comme la méthode la plus simple, il s'agit en effet de discrétiser les opérateurs de dérivation/différentiation grâce aux développements de Taylor-Young^{12 13}. Le mathématicien Brook Taylor établi en 1715 que l'on peut approximer des fonctions suffisamment dérivables

⁵<http://fr.wikipedia.org/wiki/Dirichlet>

⁶http://fr.wikipedia.org/wiki/Condition_aux_limites_de_Neumann

⁷http://fr.wikipedia.org/wiki/Carl_Neumann

⁸http://fr.wikipedia.org/wiki/%C3%89quation_int%C3%A9grale

⁹http://fr.wikipedia.org/wiki/Condition_aux_limites_de_Robin

¹⁰<http://www.cmi.univ-mrs.fr/~torresan/MathPhy/cours/node16.html#SECTION00333200000000000000000000000000>

¹¹http://fr.wikipedia.org/wiki/Condition_aux_limites_dynamique

¹²http://fr.wikipedia.org/wiki/Th%C3%A9or%C3%A8me_de_Taylor

¹³http://fr.wikipedia.org/wiki/Brook_Taylor

au voisinage d'un point par un polynôme dont les coefficients ne dépendent que des dérivées de la fonction en ce point. William Henry Young est arrivé plus tard (1863-1942) lors qu'il travailla sur la théorie de la mesure, les intégrales de Lebesgue etc.

Comme il est question de discrétiser, on peut par exemple regarder ce que cela donne sur un exemple simple :

- Prenons un segment de longueur 1
- On le découpe en $n + 1$ sous-segments dont le pas sera de $h = 1/n$

On a grosso modo trois types de différences :

- “en avant” : on prend les valeurs en x et $x + h$
- “en arrière” : on prend les valeurs en $x - h$ et x ,
- “centrées” : on prend les valeurs en $x - h/2$ et $x + h/2$

On aura alors comme approximation pour la dérivée de la fonction en x :

$$f'(x) = \frac{f(x+h) - f(x)}{h}$$

Ce qui ressemble à l'expression de la dérivée en terme de limite.

En appliquant ce principe à la dérivée première pour obtenir la dérivée second on pour obtenir quelque chose de similaire :

$$f''(x) = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2}$$

Tout ceci fonctionne bien en dimension 1, mais pour peut que les fonctions soient suffisamment dérivables, cela s'étend très bien aux dimensions supérieures.

On voit donc que l'on a besoin des valeurs en différents points qui vont former le maillage et que l'on obtient une relation assez simple pour l'expression des dérivées première, seconde, etc.

Finalement on se rend compte que si notre équation implique par exemple une combinaison linéaire de la dérivée seconde et de la fonction en tout point du domaine:

$$\alpha.f'' + \beta.f = \gamma$$

on se retrouve, grâce aux opérateurs discrétisés à n'avoir affaire qu'à un produit de valeurs de la fonction à trouver qui seront prises en des points définis du maillage.

Ainsi pour tous les points du domaine, on a une relation entre différentes valeurs de la fonction aux points du maillage. Si on représente par un vecteur f dont les différentes valeurs sont celles de la solution aux points du maillage, on peut représenter cela par le produit entre une matrice qui va exprimer cette relation et le vecteur qui correspond à la solution qui est ce que l'on cherche ensuite à résoudre avec des algorithmes.

Pour information : à partir du 18ème siècle des mathématiciens se sont mis à utiliser des développements de Taylor, et donc les différences finies pour mettre en place des abaques notamment pour les logarithmes et la trigonométrie qui étaient utilisés pour le cadastre, la navigation, l'artillerie, les statistiques, le calcul d'intérêts ou encore l'astronomie. Comme ceux-ci nécessitaient de grands nombres d'opérations de calcul, des mathématiciens et inventeurs se sont mis à tenter la mise en place de machines permettant le calcul "automatique" de ces différences finies. Le premier à presque y arriver fut Charles Babbage entre 1820 et 1843 (il n'y arriva pas complètement) et le Suédois George SCHEUTZ (1785-1873) y arriva en 1840. A savoir que ce type de machine a été utilisé jusque dans les années 1930.

Éléments finis

Pour les éléments finis, on n'utilise pas des développements de Taylor-Young comme pour les différences finies qui sont en fait des approximations des dérivées, mais plutôt des approximations des intégrales des équations aux dérivées partielles étudiées.

Dans ce type de discrétisation, il est plutôt question d'approximer la solution sur le maillage par des fonctions qui seront définies sur les éléments du domaine, et uniquement sur ceux-ci. Un peu comme une base de fonctions comme les bases dans \mathbb{R}^n .

Comme décrit dans¹⁴, à l'origine, la méthode des éléments finis était une généralisation de la méthode des déplacements pour les structures à barres, à la mécanique des milieux continus. Depuis cette technique a largement débordé ce premier cadre pour aboutir à une méthode numérique permettant de résoudre les problèmes d'équations différentielles "aux limites". C'est notamment pour cela que l'on retrouve souvent des histoires de "travail" pour exprimer certaines quantités dans les différentes formulations.

Comme je le disais, ici l'idée est de décomposer le problème sur des bases de fonctions qui sont définies sur les arêtes des "éléments" utilisés pour le maillage, et ensuite d'utiliser la décomposition de la fonction solution sur ce maillage, et par l'usage d'analyse numérique un peu poussée pour être explicité dans ce podcast, on arrive à trouver un système linéaire qui permet d'aboutir à un système linéaire de type $A.x = b$.

¹⁴docinsa

Au cours de la discrétisation, on fait ce que l'on appelle une réduction d'ordre de dérivation qui permet d'intégrer les conditions aux limites au sein du système linéaire.

Pour information, cette méthode des éléments finis est extrêmement répandue dans les logiciels de simulations pour des domaines variés allant de mécanique des milieux continus, la mécanique des fluides, la météorologie, en génie civil, en électromagnétique, etc.

Pour ceux que cela intéresse, vous pourrez trouver une liste assez longue de cours sur le sujet dans les références.

Volumes finis

De la même manière que pour les éléments finis, la méthode des volumes finis travaille sur les intégrales des EDP étudiées. A la différence des éléments finis où l'on travaille plutôt sur ce que l'on appelle la formulation variationnelle ou formulation faible (on a réduit le niveau de dérivation entre autres) on travaille ici directement sur la formulation forte ¹⁵.

En fait, cette méthode des volumes finis a d'abord été appliquée aux lois de conservation (conservation de la masse, de la quantité de mouvement, etc) qui mettent en jeu un opérateur différentiel nommé *divergence*.

Grâce à un théorème dit de flux-divergence, on transforme des équations sur des volumes (autour des points du maillage) en des équations sur des surfaces et comme les équations sont conservatives, le flux qui entre est égal au flux qui sort donc cela s'y prête bien.

A la différence des éléments finis, la méthode des volumes finis est simplement utilisable sur des maillages dit non-structurés (comme on ne se soucie pas du maillage, on peut mélanger des triangles avec des carrés, etc. Ceci est plus compliqué avec la méthode des éléments finis).

Résolution de systèmes linéaires

Une fois que ces méthodes de discrétisation nous ont permis d'obtenir des systèmes linéaires à résoudre, il est nécessaire de mettre en place des algorithmes de résolution du système obtenu.

Grosso modo, l'idée est d'inverser la matrice A pour que l'on puisse se retrouver avec $x = A^{-1}.b$. Sauf que cela n'est pas forcément évident quand on parle de matrices. Je ne reviendrais pas sur la question car elle a notamment été abordé dans de précédents podcasts je crois quand il était question de commutativité de la multiplication quand il est question de matrice.

¹⁵http://fr.wikipedia.org/wiki/M%C3%A9thode_des_volumes_finis

Il existe ainsi différentes méthodes que l'on pourra classer dans deux grandes catégories :

- Les méthodes directes où avec la technique dite du pivot de Gauss ou d'élimination de Gauss-Jordan on se débrouille pour arriver à inverser la matrice (en gros). Le problème de ces méthodes, c'est qu'elles peuvent être longues et qu'elles peuvent amener des problèmes numériques pendant l'inversion (notamment quand on va devoir diviser par des nombres petits, des choses comme ça), même si à priori elles permettent d'obtenir la solution exacte.
- Les méthodes itératives. Celles-ci proposent de partir d'une solution proposée et d'ensuite minimiser la différence entre ce qui est obtenu et la réalité. Les algorithmes les plus connus vont être ceux nommés ADI, GMRES, Gradient conjugué, etc. À noter que la méthode ADI a été l'une des premières qui fut mise en place car elle se basait sur les différences finies (relativement moins complexes que les autres méthodes de discrétisation) et prenait peu de place en mémoire (la matrice avait beaucoup de zéros et seules des bandes le long de la diagonales étaient non-nulles). Ces méthodes peuvent diverger, et il est donc important d'avoir une solution initiale pas trop "mauvaise", mais aussi que la matrice aient de bonnes propriétés (conditionnement, etc).

Quand les systèmes linéaires à résoudre deviennent trop gros et que l'on a à disposition des serveurs informatiques avec de multiples processeurs, voire même plusieurs serveurs informatiques, on peut paralléliser ces algorithmes.

Pléthore de littérature existe sur la question, et on peut faire ce que l'on appelle de la décomposition de domaine. Si on dispose de quatre processeurs et que l'on veut simuler la modification de structure d'un avion en vol, on va par exemple faire calculer la solution sur chaque aile à l'un d'entre eux et on va couper le fuselage en deux pour le distribuer entre les deux processeurs restant.

Dans ces cas-là il devient important de bien découper ses problèmes pour qu'aux frontières tout se passe bien (je rappelle que l'on calcule les solutions aux points des maillages et que si ils ne coïncident pas on peut commencer à avoir des problèmes) avec un peu de recouvrement pour que les informations de la solution à chercher puissent se propager entre les "domaines".

Les solutions informatiques qui existent et les problèmes afférents

Il existe une grande quantité de bibliothèques logicielles qui existent pour réaliser ces différentes opérations, les plus connues se nomment BLAS (pour Basic Linear Algebra Solvers), Linpack ou encore LAPACK (pour Linear Algebra Package) qui fournissent des outils pour résoudre des parties des problèmes informatiques.

A savoir que ces bibliothèques ont été écrites en Fortran, l'un des tout premiers langages informatiques de haut-niveau créé dans les années 50 et encore toujours roi dans le monde de la simulation informatique.

Je l'ai survolé, mais l'informatique en terme de matériel et de logiciel a évolué de manière conjointe. Comme je l'expliquais, on est passé de discrétisation avec des différences finies et des méthodes de type ADI peu gourmande en mémoire dans les années 50-60, à des méthodes plus complexes comme les éléments finis par la suite. On a vu aussi grandir les maillages qui n'avait que de petites tailles pour des problèmes de taille mémoire et disque à des problèmes qui font maintenant plusieurs dizaines voire centaines de millions d'inconnues et qui prennent ainsi plusieurs giga-octets de RAM. On a aussi du paralléliser les algorithmes pour pouvoir tirer partie des super-calculateur et leur puissance répartie.

Petite anecdote marrante : en 2006 j'ai fait un stage dans une société qui faisait de la simulation et un cas marquant était celui de la simulation du décollage d'un hélicoptère à turbo-réacteur. Le calcul était tellement complexe qu'il fallait près de 24 heures pour que le logiciel simule quelques dixièmes de seconde avant d'exploser sur près de 50 serveurs !

C'est dire la complexité des modèles considérés et des contraintes informatiques (autant logicielles que matérielles) qui existent !

D'ailleurs un des problèmes qui est apparu est la question des données. J'ai parlé lors du précédent podcast de la simulation de l'univers, avec les données gigantesques générées (1,5 peta-octet utile). Ce qu'il faut savoir c'est qu'il y a eu près de 100x plus de données générées qu'il a fallu trier !!!

Les cartes graphiques pour aider dans la simulation

Quelque chose qui s'est développé ces dernières années à notamment été l'usage des cartes graphiques pour aider au calcul. En tant que solution de traitement parallèle massif, les GPUs de ces cartes peuvent avoir de vrais atouts.

Il y a quand même quelques inconvénients :

- Avant que qu'OpenCL n'arrive, voire même CUDA avant lui (deux "langages dédié à l'usage de GPU") il était nécessaire de manier les structures de données propres aux jeux vidéos pour en tirer partie. Ce n'était pas très évident et plus du domaine de la bidouille qu'autre chose. Maintenant cela est plus simple, et un certain nombre de code de calcul se mettent à en tirer partie.
- Cependant les limitations en terme de mémoire de ces cartes (si on a plus de données que la place disponible dans la carte, on va adresser la mémoire centrale de l'ordinateur et l'on perd tout l'intérêt) et de précision numérique (les cartes ne calcul qu'avec des entiers de base et pas des nombres réels) font que les performances mirobolantes annoncées par Nvidia notamment en font revenir plus d'un vers le calcul plus classique

Une des alternatives qui commence à arriver serait l'usage (comme il y a bien longtemps) de co-processeurs spécialisés à cette tâche comme les Xeon-Phi de chez Intel.

Linpack, le top500, le green500

Un effet collatéral étonnant a été que l'usage de la librairie Linpack pour évaluer la performance crête des super-calculateurs. L'idée était en effet de mesurer la performance des systèmes informatiques pour la résolution d'un système linéaire basé sur les fonctions fournies par la librairie.

Pendant longtemps ce logiciel a été à la base du Top500, le classement des 500 calculateurs les plus puissants du monde. Puis avec l'avènement des cartes graphiques qui en puissance brute sont intéressantes, mais en usage réel assez peu utilisables et les problématiques de grandes données qui ne sont pas très bien prises en compte par ce test, différents autres classements sont apparus avec le Green500 notamment qui estime plutôt la performance énergétique d'un système informatique.

Il est en effet devenu crucial de gérer correctement les problématiques d'énergie, car la puissance de calcul de ces moyens informatiques grandissant, la consommation énergétique va de pair et ceci sans parler de la consommation électrique des climatisations nécessaires pour refroidir les serveurs. Pour info, aujourd'hui, il faut quasiment autant d'énergie pour la climatisation que pour les serveurs.

Avec près de 17 000 coeurs de calcul au CC-IN2P3 (le centre de calcul qui possédait les données du LHC concernant le boson de Higgs) et les 20 Peta-octets de stockage sur disque et sur bande, il est nécessaire de disposer d'une alimentation de plusieurs mégawatts !

Notes

Différences entre la description lagrangienne et la description eulerienne

Il est utile de rappeler que deux points de vue peuvent être adoptés pour décrire le mouvement dans un milieu continu :

- La description lagrangienne suit chaque particule le long de sa trajectoire : la valeur d'une variable (température, pression, vitesse...) dépend de l'instant t et de la particule considérée (identifiée par sa position $\vec{\xi}$ à l'instant t_0 de référence).
- La description eulérienne est associée à un repère indépendant du mouvement du fluide, généralement fixe : la valeur des variables fluides dépend alors du temps t et de la position d'observation \vec{x} .

La description lagrangienne est peut-être plus intuitive mais revêt un inconvénient majeur pour décrire les fluides : contrairement aux solides, les particules peuvent se déplacer librement dans la totalité d'un domaine fluide. L'analyse d'un écoulement est alors une tâche très ardue (on ne peut même pas exprimer un gradient par exemple, car on ne connaît pas les particules voisines !). On préfère donc très largement utiliser un point de vue eulérien pour décrire le mouvement d'un fluide. La difficulté réside alors dans le fait que la conservation de la quantité de mouvement est physiquement vérifiée uniquement si l'on considère une particule fluide. Or la particule $\vec{\xi}$ (variable de Lagrange) coïncidant avec le point d'observation \vec{x} (variable d'Euler) change à chaque instant t . En termes plus imagés, le randonneur qui observe un point fixe de la rivière n'a jamais les mêmes particules fluides sous les yeux. En particulier, il n'y a aucune raison pour qu'au point \vec{x} il voie la même particule aux instants t et $t + \delta t$. Comment peut-il alors calculer la vitesse ou l'accélération de la particule fluide qu'il a instantanément sous les yeux ?