TD 3 : FFT

19 Septembre 2024

Page web du cours : `https://people.irisa.fr/David.Baelde/algo/index.html`
Coordonnées des chargés de TD :
`malo.revel@ens-rennes.fr` `isseinie.calviac@irisa.fr`
Les TDs ont lieu les jeudis de 9h45 à 11h15, se fier à ADE pour la salle.

1 S'exercer avec la FFT

QUESTION 1 – Quelle est la transformée de Fourier discrète (DFT) de $\langle 1, 0, 0, 0\rangle$?

On calcule $FFT(\langle 1, 0, 0, 0\rangle, e^{i\pi/2})$ et on obtient $\langle 1, 1, 1, 1\rangle$.

QUESTION 2 – Quel vecteur admet $\langle 1, 0, 0, 0\rangle$ pour transformée de Fourier discrète ?

On calcule $\frac{1}{4}FFT(\langle 1, 0, 0, 0\rangle, e^{-i\pi/2})$ et on obtient $\langle \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\rangle$ (le calcul est le même qu'avant).

QUESTION 3 – Multiplier les polynômes $X + 1$ et $X^2 + 1$ en utilisant la FFT : choisir une bonne puissance de deux, calculer la DFT des deux séquences correspondant aux polynômes, multiplier les résultats point à point, puis appliquer la DFT inverse.

- On a besoin de vecteurs de taille 4 car le produit des polynômes est de degré 4. On prend $\omega = e^{i\pi/2}$.
- Les coefficients de $X + 1$ nous donnent le vecteur $a = \langle 0, 0, 1, 1\rangle$ et ceux de $X^2 + 1$ le vecteur $b = \langle 0, 1, 0, 1\rangle$.
- On calcule $FFT(a, \omega) \to \langle 2, -\omega - 1, 0, \omega - 1\rangle$.
- On calcule $FFT(b, \omega) \to \langle 2, 0, -2, 0\rangle$.
- On fait le produit point à point $\to c = \langle 4, 0, 0, 0\rangle$.
- On calcule $\frac{1}{4}FFT(c, \omega^{-1}) \to \langle 1, 1, 1, 1\rangle$ (on a déjà vu le calcul inverse)
- On obtient le polynôme $X^3 + X^2 + X + 1$ (on peut vérifier par le calcul).

2 FFT en place

Nous allons voir une nouvelle implémentation de la transformée de fourier rapide, qui modifie *en place* le vecteur en entrée sans en créer de nouveau.

Pour la solution, voir les notes de cours de François.

QUESTION 4 – Supposons que l'on calcule $FFT(\langle a_0, a_1, a_2, a_3, a_4, a_5, a_6, a_7 \rangle, \omega)$. Dessiner l'arbre des appels récursifs en indiquant dans chaque nœud l'argument de l'appel correspondant. L'appel principal est la racine et l'argument correspondant est $(\langle a_0, a_1, a_2, a_3, a_4, a_5, a_6, a_7 \rangle, \omega)$.

QUESTION 5 – On s'intéresse aux feuilles $\langle a_i \rangle$ de cet arbre en les parcourant de gauche à droite, et en particulier à la représentation binaire de chaque i.

Que dire de ces représentations binaires par rapport à celles de $0, 1, \ldots, 7$?

QUESTION 6 – Représenter le calcul $FFT(\langle a_0, a_1, a_2, a_3, a_4, a_5, a_6, a_7 \rangle, \omega)$ comme un circuit, construit avec un fil horizontal pour chaque a_i, et avec comme composants possibles :

— l'opération binaire "papillon" entre deux fils i et j, qui correspond au calcul suivant :

$$a_i := a_i + a_j$$
$$a_j := a_i - a_j$$

— l'opération unaire de multiplication par une puissance de ω.

QUESTION 7 – Supposons qu'un tel circuit soit implémenté. Est-ce que des opérations peuvent être réalisées en parallèle ? Avec un nombre arbitrairement grand d'unités de calcul, quel est le coût en temps asymptotique d'un appel à FFT sur un vecteur de taille n ?

3 Tri bitonique

Définition 1 (Réseau de tri). *Un **réseau de tri** est un algorithme qui trie des séquences* de taille fixée n *(qu'on supposera être une puissance de 2) de valeurs, et que l'on représente par* n *fils horizontaux parallèles, et une séquence de **comparateurs** entre deux fils (cf figure 1).*

*Dans le cas général, on appelera un réseau de tri un **réseau** lorsqu'il n'a pas forcément comme fonction de trier.*

Le fonctionnement d'un tel réseau est le suivant :

— *Chaque élément de la liste en entrée est attribué à un fil différent (de haut en bas), et est transporté de gauche à droite par son fil.*

— *Lorsqu'une paire d'éléments* (x, y)*, transportés par deux fils* i *et* j *(avec* i *au-dessus de* j*) rencontrent un comparateur, les éléments transportés par les fils* i *et* j *en sortie du comparateur sont respectivement* $min(x, y)$ *et* $max(x, y)$*.*

— *Les éléments transportés par les fils en sortie du réseau de tri représentent la séquence de valeurs renvoyées par le réseau, de haut en bas.*

Dans les réseaux de tri, certaines comparaisons peuvent être réalisées simultanément car elles concernent des paires de fils disjointes (c'est le cas des deux comparateurs au centre de la figure 1).

Nous allons essayer de construire un réseau de tri qui profite le plus possible de ces comparaisons simultanées afin d'être exécutable rapidement sur une architecture parallèle.

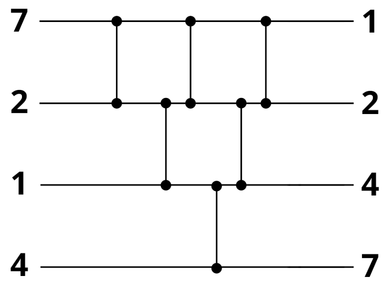

FIGURE 1 – Réseau du tri par insertion pour des séquences de taille $n = 4$, avec en entrée la séquence $(7, 2, 1, 4)$ (crédit Oskar Sigvardsson sur Wikipédia anglais)

QUESTION 8 – Dérouler l'exécution du réseau de tri de la figure 1 avec en entrée la séquence $(4, 2, 1, 0)$ (en indiquant après chaque comparateur les modifications des valeurs transportées par les fils).

Définition 2 (Séquence bitonique). *Une séquence $(x_1, \ldots, x_n)$ est dite **bitonique** s'il existe un indice k tel que $(x_1, \ldots, x_k)$ est croissante et $(x_k, \ldots, x_n)$ est décroissante, ou inversement, et toute permutation circulaire d'une suite bitonique est une suite bitonique.*

Par exemple, $(1, 2, 3, 2, 1)$, $(2, 1, 3, 4)$ et (1) sont bitoniques, mais pas $(0, 1, 0, 1)$.

Définition 3. *On appelle **séparateur** un réseau tel que celui de la figure 2. Il s'agit d'un réseau à n fils avec $n/2$ comparateurs, entre les fils i et $i + n/2$ pour tout $i \in [\![1, n/2 - 1]\!]$.*

Théorème 1. *(admis) Un séparateur vérifie la spécification suivante :*

- *<u>Entrée :</u> une séquence $(a_1, \ldots, a_n)$ bitonique*
- *<u>Sortie :</u> une séquence $(b_1, \ldots, b_{n/2}, c_1, \ldots, c_{n/2})$ telles que les séquences $(b_1, \ldots, b_{n/2})$ et $(c_1, \ldots, c_{n/2})$ sont bitoniques, et pour tout $i, j \in [\![1, n/2]\!]$, $b_i \leq c_j$.*

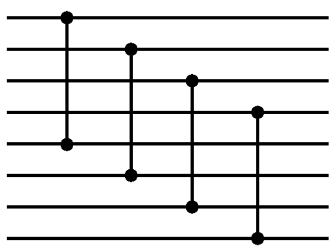

FIGURE 2 – Séparateur à 8 fils

QUESTION 9 – Construire un réseau à n fils qui vérifie la spécification suivante :

- <u>Entrée :</u> une séquence $a = (a_1, \ldots, a_n)$ bitonique

— Sortie : une séquence $(s_1, \ldots, s_n)$ qui est une permutation triée de a.

On construit inductivement :
— si $n = 1$, rien à faire
— si $n \geq 2$, on met deux circuits A et B qui respectent la spec pour $n/2$ l'un au-dessus de l'autre (par hypothèse d'induction), et on ajoute à gauche un séparateur de taille n. Ce circuit total prend bien en entrée une séquence bitonique, puis le séparateur renvoie deux séquences bitoniques que A et B vont trier, et chaque valeur en sortie de A est inférieure à chaque valeur en sortie de B car A et B n'ont fait que permuter leurs éléments, donc la séquence finale est triée, et est une permutation de a.

Question 10 – En remarquant que si $(b_1, \ldots, b_{n/2})$ et $(c_1, \ldots, c_{n/2})$ sont triées alors $(b_1, \ldots, b_{n/2}, c_{n/2}, \ldots, c_2, c_1)$ est bitonique, expliquer comment construire un réseau de tri à n fils à partir de la construction précédente.

Dessiner ce réseau pour $n = 8$.

On appelle ce réseau le **tri bitonique**.

On le construit par induction :
— Si $n = 1$, rien à faire
— Si $n \geq 2$, par induction on place deux réseaux A et B qui trient des séquences de taille $n/2$ l'un au-dessus de l'autre, et en sortie de ces deux réseaux on place notre construction précédente avec les séparateurs, sauf qu'on modifie le premier séparateur de façon à inverser la sortie de B : plutôt que d'être entre les fils i et $i + n/2$, les comparateurs sont entre les fils i et $n - i + 1$.

Pour le dessin, cf wikipedia.

Définition 4. *La **profondeur** d'un réseau est le plus grand nombre de comparateurs qu'un élément en entrée peut rencontrer sur son chemin dans le réseau (ce chemin n'est pas forcément réalisable en pratique). Par exemple, le réseau de la figure 1 a une profondeur 5, et celui de la figure 2 une profondeur 1.*

La profondeur d'un réseau représente le coût en temps de l'exécution de ce réseau lorsqu'on dispose d'un nombre arbitrairement grand d'unités de calcul (i.e. toutes les comparaisons parallèles sont effectuées simultanément).

Question 11 – Quelle est la profondeur asymptotique du réseau du tri bitonique en fonction de son nombre de fils n ? Quel est son nombre de comparateurs ?

Profondeur : $O(log(n)^2)$.
Taille : $O(n.log(n)^2)$

Question 12 – (Bonus) Soit un réseau à n fils. Montrer que si ce réseau trie correctement toutes les séquences d'éléments de $\{0, 1\}$ de taille n, alors il trie correctement toutes les séquences d'éléments de $\mathbb{N}$ de taille n (indice : montrer que pour toute fonction f croissante, si le réseau transforme $(a_1, \dots a_n)$ en $(b_1, \dots, b_n)$ alors il transforme $(f(a_1), \dots, f(a_n))$ en $(f(b_1), \dots, f(b_n))$).

On appelle cette propriété des réseaux de tri le *théorème 0-1 de Knuth*.

On montre que pour toute fonction f croissante, si le réseau transforme $(a_1, \dots a_n)$ en $(b_1, \dots, b_n)$ alors il transforme $(f(a_1), \dots, f(a_n))$ en $(f(b_1), \dots, f(b_n))$ par induction sur le nombre de comparateurs du réseau (promis c'est facile).

Montrons la contraposée du théorème, en supposant qu'il existe deux éléments $a_i < a_j$ tels que a_i est après a_j dans $(b_1, \dots, b_n)$. Alors posons la fonction f telle que $f(x) = 0$ si $x \leq a_i$, et $f(x) = 1$ sinon.

f est bien monotone, donc le réseau transforme $(f(a_1), \dots, f(a_n))$ en $(f(b_1), \dots, f(b_n))$, et dans la séquence $(f(b_1), \dots, f(b_n))$ il y a un 0 après un 1.