Parallélisme Régulé

Assala ASSELLALOU

21 mars 2025

1 Introduction

Ce TP a pour objectifs de gérer le parallélisme à gros grain, de paralléliser un algorithme par décomposition en sous-tâches et de connaître les services d'exécution de la plateforme Java

2 Calcul du maximum d'un tableau

Le but est de trouver l'élément maximal du tableau. La méthode '***LargeIntArray.max(int[] array, int start, int end)***' calcule le max en parcourant séquentiellement le tableau entre les indices ***'[start..end['***.

2.1 Méthode séquentielle

'MaxTabSequential.java' contient une version purement séquentielle.

2.2 Méthode Threads simples

Le fichier ***'MaxTabThread.java'*** a été complété en s'inspirant de l'exemple de la somme dans le fichier ***'SommeThreadPlus.java'***

La méthode de thread simple consiste à utiliser un seul thread pour exécuter une tâche à la fois. Cela signifie que toutes les opérations sont effectuées de manière séquentielle, sans parallélisme. On verra plus tard que cette méthode n'est pas efficace face au situations bloquantes ou très longues.

2.3 Méthode Pool de Threads

Le fichier ***'MaxTabPool.java'*** a été complété en s'inspirant de ***'exemples/SommePool.java'.***
La méthode du pool de threads consiste à utiliser un nombre fixe de threads réutilisables pour exécuter plusieurs tâches concurrentes.
Les applications peuvent mettre en file d'attente des éléments de travail, associer des travaux à des handles pouvant être mis en attente, mettre automatiquement en file d'attente en fonction d'un retardateur et établir une liaison avec des E/S.

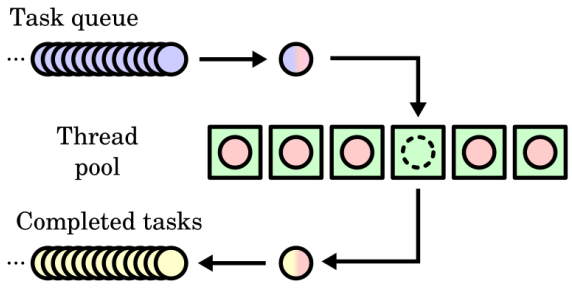

FIGURE 1 – Thread Pool

2.4 Méthode Fork Join

Le fichier ***'MaxTabForkJoin.java'*** a été complété en s'inspirant de ***'exemples/SchemaForkJoin.java'.***

Le ForkJoin divise un travail complexe en sous-tâches plus petites, qui peuvent être traitées en parallèle, puis fusionne les résultats pour obtenir le résultat final. Ce modèle est particulièrement adapté pour les tâches récursives où les sous-tâches sont indépendantes les unes des autres.

2.5 Comparaison des résultats

On a comparé les performances des solutions avec un gros tableau ***'foo'*** conformement au sujet.

Essai	MaxTabSequential	MaxTabThread	MaxTabPool	MaxTabForkJoin
0	3860 µs	14931 µs	23646 µs	22214 µs
1	669 µs	1213 µs	431 µs	724 µs
2	2168 µs	1037 µs	397 µs	737 µs
3	518 µs	718 µs	504 µs	688 µs
4	492 µs	1014 µs	325 µs	638 µs
5	445 µs	472 µs	424 µs	812 µs
6	574 µs	564 µs	407 µs	804 µs
7	504 µs	580 µs	260 µs	775 µs
8	511 µs	614 µs	332 µs	813 µs
9	490 µs	717 µs	380 µs	790 µs
10	506 µs	953 µs	407 µs	722 µs
11	532 µs	857 µs	347 µs	1025 µs
12	528 µs	673 µs	379 µs	1427 µs
13	452 µs	3062 µs	355 µs	1101 µs
14	443 µs	856 µs	243 µs	934 µs
15	478 µs	818 µs	468 µs	1509 µs
16	654 µs	711 µs	281 µs	915 µs
17	597 µs	843 µs	316 µs	1040 µs
18	604 µs	609 µs	390 µs	1193 µs
19	606 µs	602 µs	484 µs	1032 µs
Moyenne	**526 µs**	**872 µs**	**362 µs**	**971 µs**

Table 1 – Comparaison des performances des solutions

2.6 Variation du nombre de threads / taille du pool / seuil d'arrêt

2.6.1 Variation du nombre de threads

Nombre de Threads	Moyenne des Durées
1	1095 µs
2	762 µs
4	920 µs
8	939 µs

Table 2 – Moyenne des durées selon le nombre de threads

2.6.2 Variation de la taille du pool

Nombre de pool de threads	Moyenne des durées (µs)
2	460
4	408
8	378

TABLE 3 – Moyennes des durées pour différents nombres de pool de threads

2.6.3 Variation du seuil d'arrêt

Seuil d'arrêt	Moyenne des durées (µs)
100	2078
500	1343
1000	830

TABLE 4 – Moyenne des durées pour différents seuils d'arrêt.