

## **Acceler'8 : Maximum Subarray Problem**

### **I – Approche algorithmique du problème**

Après une première réflexion sur des algorithmes de résolution du problème de Maximum Subarray Problem (appelé MSP par la suite), nous avons dans un premier temps décidé de concevoir différents algorithmes suivant les cas particuliers que nous pouvions rencontrer. C'est ainsi que nous avons rédigés des algorithmes différents selon que la matrice soit totalement positive, totalement négative, à une dimension ou finalement à deux dimensions. Cette sélection d'algorithme a lieu lors de la lecture du fichier contenant la matrice.

Nous nous sommes rapidement rabattu sur l'algorithme de Kadane qui constitue une réponse à notre problème et qui a une complexité de  $O(N)$  pour une matrice à une dimension et  $O(N^3)$  pour une matrice à deux dimensions. Cet algorithme consiste dans le parcours de sous zones de la matrice et l'addition de la nouvelle zone si celle-ci permet de créer une somme plus importante.

### **II – Problématique de lecture des fichiers**

Grace à l'utilisation de Intel Vtune, nous avons identifié un goulet d'etranglement lors de la lecture des fichiers. Malgré nos efforts, nous ne sommes pas parvenu à paralléliser l'analyse de ceux ci.

Nous avons tenté de charger le fichier en memoire et d'interpreter le contenu sur plusieurs threads. Malgré nos efforts, cette approche offrait de mauvais resultats.

### **III – Mise en parallèle**

Afin de réaliser la mise en parallèle de notre algorithme, nous nous sommes tournés vers la librairie C++ Intel TBB. La librairie nous permettait de controler comme nous le souhaitions la parallelisation. Nous avons utilisé une reduction pour optimiser la premiere boucle de Kadane 2D et avons ecrit notre propre classe d'exploration BlockKadan2d.