

Clément Hue

TP PVM

Pour paralléliser via PVM, on part sur le même principe que OpenMP, on parallélise le calcul des couleurs, car on ne peut paralléliser l'écriture dans les fichiers images, car l'ordre d'écriture est importante.

Pour ce faire on attribut à chaque worker la tâche de calculer une partie du tableau des couleurs. PVM ne permet pas d'envoyer directement des type spéciaux (struct par exemple) ou des tableaux de tableaux, mais uniquement des tableaux de type primitif.

Pour pouvoir transmettre les tableaux de couleurs calculés aux farmers, les workers calculent 3 tableaux : un tableau qui représente la couleur rouge, un qui représente la couleur vert et un autre qui représente la couleur bleu. Comme dit précédemment, ne pouvant envoyer des tableaux de tableaux, les couleurs sont calculées en un seul tableau, la première ligne occupe les x premières cases du tableau, la seconde les x suivantes et ainsi de suite.

Puis enfin le farmer récupère les tableaux de chaque workers et reconstitue le tableau de couleurs complet pour ensuite écrire dans les fichiers images les pixels.

Au niveaux des performances, on constate une diminution de 50 % du temps d'exécution par rapport au séquentiel, soit 8,3s, (17s pour la version séquentielle) mais reste supérieur à la version OpenMP qui descendait elle à 6s.

On remarque pendant l'exécution que tous les threads sont utilisés.

The screenshot shows a terminal window with two panes. The top pane displays PVM statistics for a Mandelbrot calculation. The bottom pane shows a list of processes running on the system.

PVM Statistics:

Worker	Progress (%)	Worker	Progress (%)	Worker	Progress (%)	Worker	Progress (%)
1	69.1%	4	87.5%	7	54.7%	10	51.2%
2	69.7%	5	59.6%	8	62.2%	11	58.6%
3	64.6%	6	66.9%	9	60.3%	12	46.7%

Mem: 5.44G/15.6G
Swp: 0K/2.00G
Tasks: 234, 1391 thr; 12 running
Load average: 0.82 0.65 0.71
Uptime: 02:00:34

Process List:

PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
24680	clement	20	0	1684M	1029M	12340	S	372.	6.4	2:06.25	/home/clement/.vscode/extension
24772	clement	20	0	1684M	1029M	12340	R	104.	6.4	0:05.61	/home/clement/.vscode/extension
2675	clement	20	0	4573M	429M	124M	S	13.4	2.7	3:51.51	/usr/bin/gnome-shell
24579	clement	20	0	4354M	70816	47896	S	32.5	0.4	0:01.72	/usr/share/code/code /usr/share
24769	clement	20	0	1684M	1029M	12340	S	19.5	6.4	0:01.08	/home/clement/.vscode/extension
24766	clement	20	0	1684M	1029M	12340	S	19.9	6.4	0:01.09	/home/clement/.vscode/extension
24771	clement	20	0	1684M	1029M	12340	S	19.5	6.4	0:01.06	/home/clement/.vscode/extension
24763	clement	20	0	1684M	1029M	12340	R	19.5	6.4	0:10.20	/home/clement/.vscode/extension
24765	clement	20	0	1684M	1029M	12340	S	19.1	6.4	0:01.07	/home/clement/.vscode/extension
24767	clement	20	0	1684M	1029M	12340	R	19.9	6.4	0:01.10	/home/clement/.vscode/extension
24770	clement	20	0	1684M	1029M	12340	R	20.3	6.4	0:22.15	/home/clement/.vscode/extension
24762	clement	20	0	1684M	1029M	12340	S	19.5	6.4	0:10.11	/home/clement/.vscode/extension
24761	clement	20	0	1684M	1029M	12340	S	20.3	6.4	0:01.74	/home/clement/.vscode/extension
24768	clement	20	0	1684M	1029M	12340	R	19.9	6.4	0:05.53	/home/clement/.vscode/extension
24764	clement	20	0	1684M	1029M	12340	R	19.5	6.4	0:01.03	/home/clement/.vscode/extension
5609	clement	20	0	7412	6744	2224	R	16.8	0.0	0:06.10	/usr/bin/pvmd
3331	clement	20	0	896M	250M	48932	S	0.0	1.6	0:06.26	/usr/bin/gnome-software --gappl

Terminal shortcuts: F1Help F2Setup F3Search F4Filter F5Tree F6SortBy F7Nice F8Nice F9Kill F10Quit

Chaque worker est en complexité $O(n^3)$ et le farmer en $O(n^2)$, on est donc en $O(n^3)$ (hauteur * largeur * n_{max} , 12), 12 car le PC possède 12 threads