

# RAPPORT DU PROJET CHP

## Equation de la chaleur 2D par différences finies

Victor Lederer

15/05/2019

## 1 Introduction

L'objectif de ce projet est la programmation parallèle via la librairie MPI d'un code de calcul 2D par différences finies pour résoudre l'équation de la chaleur sur un domaine cartésien. Le projet permet d'étudier l'influence de la répartition de la charge et l'envoi des messages entre processeurs, sur le speed up et l'efficacité. Une présentation du cadre de travail donnera suite à l'explication de la gestion du parallélisme mis en œuvre. Nous exposerons en suite l'analyse de nos courbes de speed up.

## 2 Cadre de travail théorique

On présente ici le domaine spatial d'étude, la discrétisation et la numération des points de calculs qui permettent d'écrire le schéma numérique.

### 2.1 Equation et conditions limites

La figure 1 représente le domaine  $\Omega = [0, L_x] \times [0, L_y]$  de  $R^2$ , de frontières  $\Gamma_0 = [0, L_x] \times \{\{0\} \cup \{L_y\}\}$ ,  $\Gamma_1 = \{\{0\} \cup \{L_x\}\} \times [0, L_y]$ . L'équation de la chaleur 2D munie des conditions de diriclets aux frontières s'écrit telle que :

$$\partial_t U(x, y, t) - D \cdot \Delta U(x, y, t) = f(x, y, t) \quad (1)$$

$$U|_{\Gamma_0} = g(x, y, t) \quad (2)$$

$$U|_{\Gamma_1} = h(x, y, t) \quad (3)$$

Le terme source  $f$  et les conditions de bords  $g, h$  sont les suivantes :

- CAS 1 :  $f = 2(y - y^2 + x - x^2)$ ,  $g = h = 0$ .
- CAS 2 :  $f = g = h = \sin(x) + \cos(y)$ .
- CAS 3 :  $f = \exp(-(x - \frac{L_x}{2})^2) \cdot \exp(-(y - \frac{L_y}{2})^2) \cdot \cos(\frac{\pi}{2} \cdot t)$ ,  
 $g = 0$ ,  $h = 1$ .

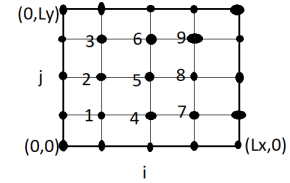


FIGURE 1 – Domaine d'étude, numérotation des noeuds de calculs.

### 2.2 Discrétisation et notation

On note  $N_x$  et  $N_y$  les nombres de noeuds dans les directions  $X$  et  $Y$ . Les points de calculs de la solution sont les points intérieurs au domaine, il y en a donc  $(N_x - 2) \times (N_y - 2)$ . On a le pas d'espace en  $X$ ,  $\Delta X = \frac{L_x}{N_x - 1}$  telle que  $\forall i \in [0, N_x - 1]$ ,  $X_i = i \cdot \Delta X$ . Idem en  $Y$  avec  $L_y$  et  $N_y$ . On peut donc repérer un point du domaine par le couple  $(i, j)$  que l'on nomme numérotation locale, ou bien par ses coordonnées  $(X_i, Y_j)$ . De plus on peut affecter une numérotation globale aux points intérieurs par  $P_{ij} = (i - 1) \cdot (N_x - 2) + j$  comme le montre la figure 1 ci-dessus. On peut donc approcher l'opérateur Laplacien de l'équation (1) via la formule de Taylor centrée d'ordre 2, telle que :

$$\partial_x^2(x, y, t) + O(\Delta X^3) = \frac{U(x + \Delta X) - 2U(X) + U(X - \Delta X)}{\Delta X^2} \quad (4)$$

$$\partial_y^2(x, y, t) + O(\Delta Y^3) = \frac{U(y + \Delta Y) - 2U(Y) + U(Y - \Delta Y)}{\Delta Y^2} \quad (5)$$

La dérivée en temps est approchée par une formule d'ordre un, avec  $\Delta T$  le pas temporel telle que :

$$\partial_t U(x, y, t) + o(\Delta T) = \frac{U(t + \Delta T) - U(t)}{\Delta T} \quad (6)$$

On a ainsi le schéma implicite centré d'ordre deux en espace tel que  $\forall (i, j) \in [[1, N_x - 2]] \times [[1, N_y - 2]]$  :

$$\sigma U_{ij}^{n+1} - \sigma_x (U_{i+1j}^{n+1} + U_{i-1j}^{n+1}) - \sigma_y (U_{ij+1}^{n+1} + U_{ij-1}^{n+1}) = U_{ij}^n + \Delta T F_{ij}^n \quad (7)$$

Avec  $\sigma = 1 + 2\sigma_x + 2\sigma_y$ ,  $\sigma_x = \frac{D \cdot \Delta T}{\Delta X^2}$ ,  $\sigma_y = \frac{D \cdot \Delta T}{\Delta Y^2}$ , le schéma préserve la positivité de la solution puisque  $\sigma + \sigma_x + \sigma_y = 1$  et il est inconditionnellement stable (Euler implicite). Les termes de bords étant connus, ils sont passés dans le membre de droite de l'équation (7), le schéma sous forme matricielle s'écrit alors :

$$A \cdot U = U_o + \Delta T \cdot F + C L \quad (8)$$

En notant  $N = (N_x - 2) \times (N_y - 2)$ , la matrice A de dimension  $N^2$  est symétrique définie positive, pentadiagonale et matrice block D et C de taille  $(N_x - 2) \times (N_y - 2)$  tel que :

$$A = \begin{Bmatrix} D & C & 0 & 0 & 0 \\ C & D & . & 0 & 0 \\ 0 & . & . & . & 0 \\ 0 & 0 & . & . & C \\ 0 & 0 & 0 & C & D \end{Bmatrix} \text{ avec } D = \begin{Bmatrix} \sigma & -\sigma_y & 0 & 0 \\ -\sigma_y & \sigma & -\sigma_y & 0 \\ 0 & -\sigma_y & \sigma & -\sigma_y \\ 0 & 0 & -\sigma_y & \sigma \end{Bmatrix}, C = -\sigma_x \cdot I_{N_x-2}$$

On montre que A est bien définie-positive :

$$\begin{aligned} \langle AX|X \rangle &= \sum_{i=1}^N \sigma X_i^2 - \sum_{i=1}^{N-1} \sigma_y X_i X_{i+1} - \sum_{i=2}^N \sigma_y X_i X_{i-1} - \sum_{i=1}^{N-(N_x-2)} \sigma_x X_i X_{i+N_x-2} - \sum_{i=N_x-1}^N \sigma_x X_i X_{i-N_x+2} \\ &= \sum_{i=1}^N \sigma X_i^2 - 2 \sum_{i=1}^{N-1} \sigma_y X_i X_{i+1} - 2 \sum_{i=1}^{N-(N_x-2)} \sigma_x X_i X_{i+N_x-2} \\ &= \sum_{i=1}^N X_i^2 + \sum_{i=1}^{N-1} \sigma_y (X_i - X_{i+1})^2 + \sum_{i=1}^{N-(N_x-2)} \sigma_x (X_i - X_{i+N_x-2})^2 + \sigma_y (X_N^2 + X_1^2) + \sum_{i=N-N_x+3}^N \sigma_x X_i^2 + \sum_{i=1}^{N_x-2} \sigma_x X_i^2. \end{aligned}$$

Comme  $\sigma$ ,  $\sigma_x$  et  $\sigma_y > 0$ , on a  $\langle AX|X \rangle > 0$  et est nul si et seulement si  $X = 0$ .  $U_o$ ,  $F$  et  $C$  sont respectivement les vecteurs solution, fonction source et conditions de bords au temps  $n - 1$ .  $U$  est le vecteur solution au temps  $n$ . Le codage spatial vectoriel utilise l'équivalence entre le système de coordonnées en base entière  $(i, j)$  et la numérotation  $P_{ij}$ . On fixe  $i$  puis on déroule  $j = 1, N_y - 2$ , pour  $i = 1, N_x - 2$ .

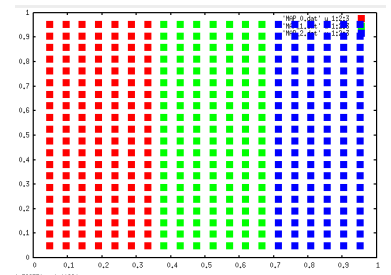
### 3 Gestion du parallélisme

Les durées d'exécution du code furent relevées pour un Intel Xeon(R) CPU E3-1245 v3 at 3.40GHz x 8 muni de 7,6 Gio de mémoire Ram. Le constructeur indique que ce CPU possède quatre coeurs physiques et huit threads (séquence d'ordonnances logicielles). Le fichier README du code indique les entrées pour la compilation, l'exécution ainsi que la commande pour visualiser l'erreur en norme 2. Les temps d'exécutions relevés comprennent aussi l'écriture en parallèle de la solution. La tolérance du Gradient conjugué a été fixé à 1E-16.

#### 3.1 Description de la répartition de la charge

On note  $N_p$  le nombre de processus actif et  $CHARGE\_TOT$  la charge totale correspondant au nombre de points de calcul de la solution sur la domaine intérieur. Cette charge doit être distribuer équitablement entre les  $N_p$  processus, pour cela on définit  $it_1$  et  $it_N$  les bornes de travail des processus sur la numérotation globale des noeuds de calcul. On souhaite aussi connaitre l'intervalle de travail induit par  $it_1$  et  $it_N$  des unités de calcul sur la discrétisation en  $X$  et en  $Y$ . En rappelant que  $it_1 = P_{i1j1}$  et  $it_N = P_{iNjN}$ , on a la charge suivante en  $X$  :  $[S1, S2]$  avec  $i1 = S1$  et  $iN = S2$  et la charge suivant  $Y$  :  $GAP(S1 : S2, 1 : 2)$  est une fonction de  $[S1, S2]$  et de  $[it_1, it_N]$ . On la stocke dans un tableau d'entier  $GAP$ . Deux mode de répartition sont implémentés.

**Répartition en mode 1** Elle effectue d'abord une répartition sur l'espace  $X$  en fonction de  $N_p$  et  $(N_x-2)$ . Elle en déduit la charge globale par processeurs  $[it_1, it_N]$  et fixe pour tout les processeur la charge en  $Y$  tel que  $GAP(:, 1) = 1$  et  $GAP(:, 2) = N_y - 2$ . Cela peut induire une différence maximale de charge globale de  $N_y - 2$  entre deux processus(cf figure 2,  $N=20^2$ ,  $N_p=3$ ) et elle ne peut être utilisée que pour  $N_p \leq N_x - 2$ .



**Répartition en mode 2** Elle effectue d'abord une répartition sur la numérotation globale  $[it_1, it_N]$  en fonction de  $N_p$  et  $N$ . Elle en déduit la charge en  $X$  par processeurs  $[S1, S2]$  et calcule par processeur la charge en  $Y$  tel que  $GAP(:, 1)$  et  $GAP(:, 2)$  soient variables suivant  $X$ . Cela induit une différence maximale de charge globale de 1 entre deux processus comme le montre la figure 3 ( $N=20^2$ ,  $N_p=3$ ) où la répartition du rang=0 est rouge, du rang=1 est verte et du rang=3 est bleu, pour  $N_x=N_y=22$  et elle peut être utilisée tant que  $N_p \leq N$ .

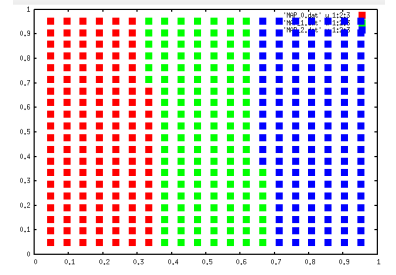


FIGURE 3 – Répartition mode 2

Avec ces répartitions, chaque processeur connaît  $U(it_1 : it_N)$ . Cependant, comme le montre l'équation 7 la solution calculée pour  $U_{ij}$  est une fonction de  $U_{i+1j}, U_{i-1j}, U_{ij+1}$  et  $U_{ij-1}$ . Autrement dit, un processeur doit connaître une partie de la solution de ses processus voisins. Par simplicité d'exécution, on prend la surcharge de taille  $N_y - 2$  tel que pour un rang donné :

- (Rang-1) envoie  $U(it_N - (N_y - 2) + 1 : it_N)$  à (Rang)
- (Rang+1) envoie  $U(it_1 : it_1 + (N_y - 2) - 1)$  à (Rang)

Le processus (Rang) connaît alors  $U(it_1 - (N_x - 2) : it_N + (N_x - 2))$ . Les sous parties suivantes détaillent ces communications.

### 3.2 Description du parallélisme

Le code parallèle permet de réduire l'allocation en mémoire, puisque chaque processus connaît les vecteurs par partie. De plus on ne stocke que les trois coefficients ( $\sigma, \sigma_x$  et  $\sigma_y$ ) non nuls de la matrice  $A$  des équations 7 et 8. Cela permet de ne pas allouer de l'espace mémoire en plus dans la stack des processus en vue de réaliser nos calculs plus rapidement dans les caches ne niveaux L1 à L3. Nous avons vu en 3.1 que le parallélisme induit des communications, notamment dans le gradient conjugué (GC) que nous précisons ici pour une itération en temps, en supposant que le GC converge en  $k$  itération :

- $(k+1)$  échange de partie de vecteur, une avant chaque produit matrice vecteur
  - $(2*k+1)$  réduction de réel après un produit scalaire, pour le pas de descente du GC et la norme du résidu
- Le code propose deux modes de communication de vecteur :

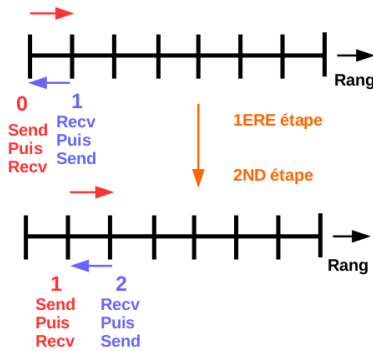


FIGURE 4 – Communication de vecteur en mode PtoP

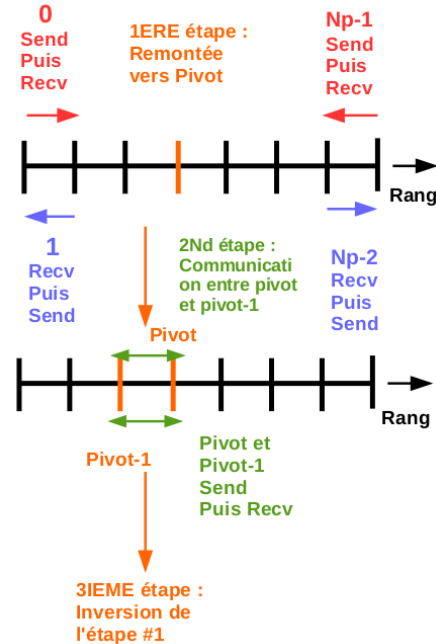


FIGURE 5 – Communication de vecteur en mode PtoP 2WAYS

**Le mode PtoP** ou Point à Point sur la figure 4 qui engage deux processus à la fois tel que :

- (Rang) envoie à (Rang+1)  $V(it_N - (N_y - 2) + 1 : it_N)$ , puis
- (Rang+1) reçoit dans  $V(it_1 - (N_y - 2) : it_1 - 1)$  et,

- (Rang+1) envoie à (Rang)  $V(it_1 : it_1 + (Ny - 2) - 1)$ , enfin,
- (Rang) reçoit dans  $V(it_N + 1 : it_N + (Nx - 2))$

La communication se termine lorsque les rang (Np-2) et (Np-1) ont terminé leurs échanges, il ya (Np-1) étapes de 2 envois et 2 réceptions de taille (Ny-2). On voit quand attendant leur tour les processeurs ne réalise aucune tâche. On ne peut pas utiliser des communications non bloquantes car l'opération suivante est celle du produit matrice vecteur et l'on a vu qu'elle nécessite cette surcharge.

**Le mode PtoP 2WAYS** Pour palier à ce problème, la communication en mode PtoP 2WAYS se base sur le même principe et induit le même nombre de communication. Cependant comme le montre la figure 5, il est possible en choisissant un rang pivot, tel que :

$$pivot = \frac{Np - \text{mod}(Np, 2)}{2} \quad (9)$$

où "mod(Np,2)" correspond au reste de la division euclidienne de Np par 2, de diminuer en théorie le temps de communication par 2 puisque l'échange d'information en mode PtoP est initiée par les deux paires de processus limites (0,1) et (Np-2,Np-1). L'information se propage par paire dans deux sens : de (0,1) vers (Pivot-2,Pivot-1) et de (Np-2,Np-1) vers (Pivot,Pivot+1). L'étape deux correspond à l'échange entre (Pivot) et (Pivot-1) toujours selon une communication PtoP. La dernière étape consiste à inverser la première, d'une part de (Pivot-2,Pivot-1) vers (0,1) et d'autre part en simultanée de (Pivot,Pivot+1) vers (Np-2,Np-1).

## 4 Validation du code

Pour valider le code, nous utiliserons les solutions analytiques pour les cas 1 et 2. Pour le cas 3 on peut vérifier en visualisant la solution au cours du temps qu'elle est périodique. On précise ici l'erreur en norme infinie et en norme 2 :

$$ERR(N\infty) = \text{MAX}_{i=1,N} |U_{exa}(i) - U(i)| \quad (10)$$

$$ERR(N2) = \sqrt{\langle U_{exa} - U | U_{exa} - U \rangle} \quad (11)$$

**Cas 1** La solution analytique est la suivante :

$$U_{exa} = X * (X - 1) * Y(Y - 1) \quad (12)$$

On voit avec le tableau 1 que les erreurs sont de l'ordre de  $10^{16}$ , elles sont identiques quelque soit Np. Cela permet de valider l'implémentation du cas 1 et le GC. A noter que ITMAX correspond au nombre d'itération en temps.

TABLE 1 – Erreur pour le cas 1

Nx×Ny	$\Delta T$	ITMAX	ERR(N2)	ERR(N $\infty$ )
32×32	0,1	60	7,834E-16	6,245E-17
42×42	0,1	60	7,72E-16	9,02E-17

TABLE 2 – Erreur pour le cas 2

Nx×Ny	$\Delta T$	ITMAX	ERR(N2)	ERR(N $\infty$ )
102×102	0,1	60	4,52E-5	8,09E-7
202×202	0,1	60	2,27E-5	2,04E-7

**Cas 2** La solution analytique est :

$$U_{exa} = \text{Sin}(X) + \text{Cos}(Y) \quad (13)$$

Le tableau 2 donne l'ordre des erreurs pour ce cas. On en déduit que l'ordre numérique du schéma en espace (eq7) est de  $\text{LOG}_2(\frac{ERR_{\Delta x}}{ERR_{\Delta x/2}}) = 1$ . On précise aussi que même si le schéma en temps est implicite est donc inconditionnellement stable, il est au plus d'ordre 1 en temps. Il faut donc prendre un pas de temps assez petit pour être précis. la figure 6 montre la visualisation de l'erreur absolue pour le cas 2. On constate une valeur plus grande de l'erreur au centre du maillage. Nous avons aussi vérifié que l'on converge vers la solution exacte indépendamment de Np, du mode 1 ou 2, et du type de communication.

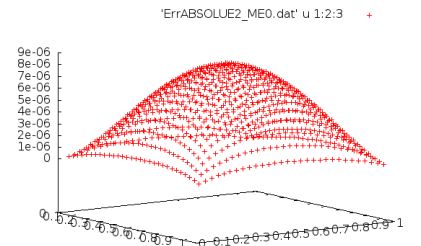


FIGURE 6 – Erreur absolue pour le cas 2

## 5 Analyses des temps d'exécution

Nous commenterons les courbes de Speed Up et d'efficacité. Nous regarderons aussi les temps induits par les communications afin de commenter le parallélisme mis en oeuvre.

### 5.1 Résultats pour le cas 1

les figures 7 et 8 présentent les speed up suivant le mode de répartition 1 ou 2. On constate que les speed up tant

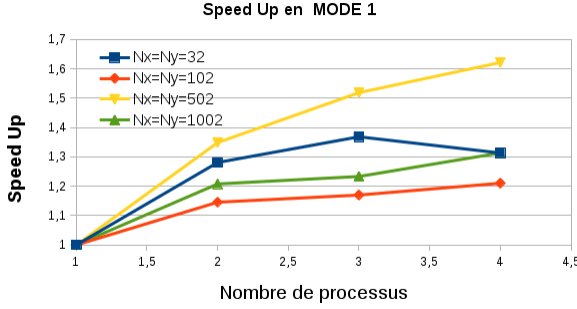


FIGURE 7 – Speed Up en mode 1

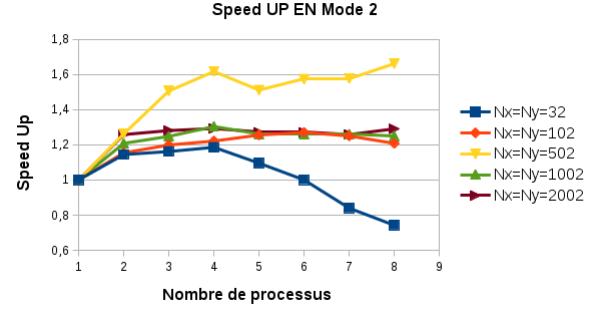


FIGURE 8 – Speed Up en mode 2

en mode 1 que 2 sont très inférieurs à la valeur idéale  $S(Np) = Np$ . La meilleur tendance semble être pour  $N=502$ . On constate aussi sur les figures 9 et 10 que l'efficacité du parallélisme est quasi-absente, idéalement on doit avoir  $E(rNp) = S(Np)/Np = 1$  Pour expliquer cela, on trace sur la figure 11 la somme des temps de communication, de

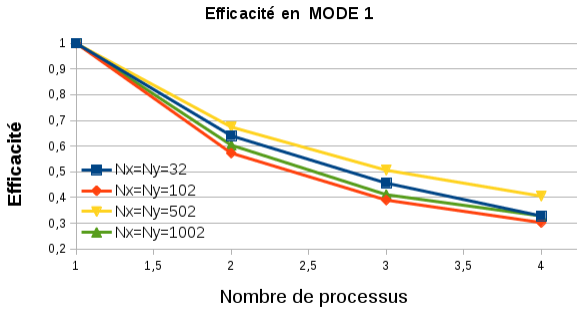


FIGURE 9 – Efficacité en mode 1

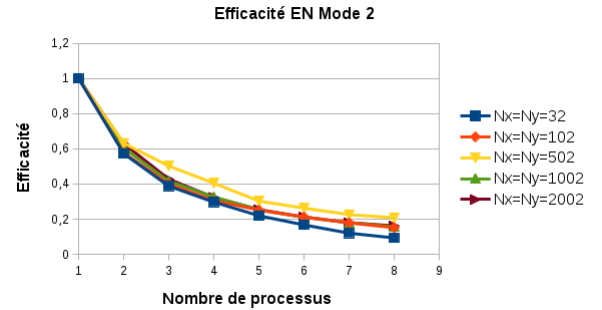


FIGURE 10 – Efficacité en mode 2

produit matrice vecteur et allreduce en pourcentage du temps total. On voit pour  $N=502$  que la somme des temps en parallèle est supérieur à 10% du temps total, en retirant de la somme des temps du produit matrice vecteur cela ne diminue pas assez ce pourcentage. En effet comme le montre l'Annexe A le temps du produit matrice vecteur décroît d'un facteur quasiment  $1/Np$ . Autrement dit le parallélisme apporte un gain sur le produit matrice vecteur (PMV) mais engendre des communication trop importante par rapport au séquentiel. Paradoxalement, pour  $N=2002$  si l'on retire le temps du PMV le pourcentage tombe entre 0% et 5%. Cela peut indiquer que le parallélisme doit être appliqué pour  $Np$  très grand. Les annexes B C D E donnent les tables de valeur des figure 7 à 10.

### 5.2 Résultats pour le cas 2 et 3

Ces deux cas impliquent des conditions de bords non nuls qu'il faut rajouter à chaque itération en temps. On souhaite voir l'influence de ces calculs sur les speed up. Les figure 12 et 13 présentent les courbes de speed up pour ces cas. Là encore les speed up sont loin des courbes théoriques. Néanmoins avec la figure 14 on montre que le parallélisme permet d'avoir sur le PMV un speed up superlinéaire pour  $N=102$ , linéaire pour  $N=502$  pour le cas 2. Les courbes d'efficacité (figure 15) nous indiquent que le pour  $N=102$  et 502 le parallélisme est efficace.

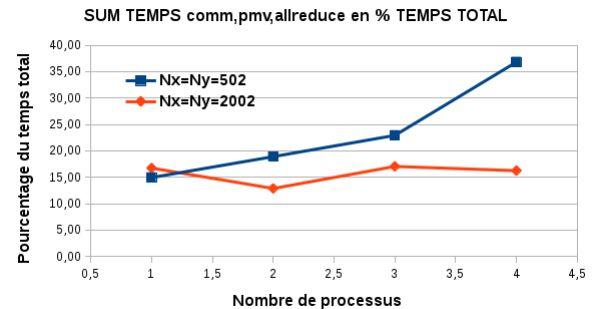


FIGURE 11 – Pourcentage du temps total en mode 2

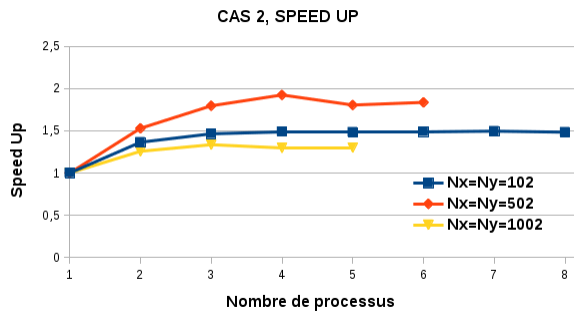


FIGURE 12 – Speed Up cas 2

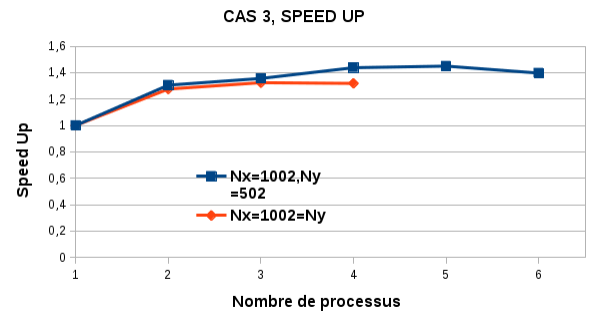


FIGURE 13 – Speed up cas 3

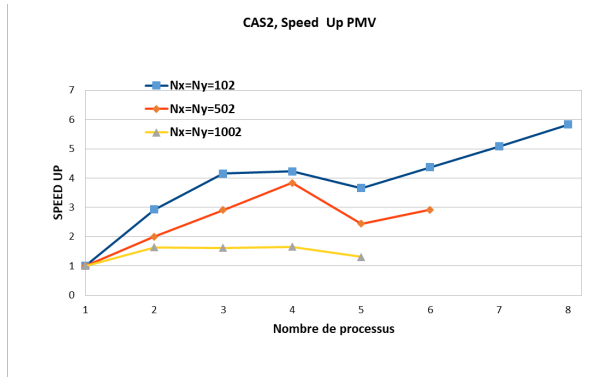


FIGURE 14 – Speed Up du PMV, cas 2

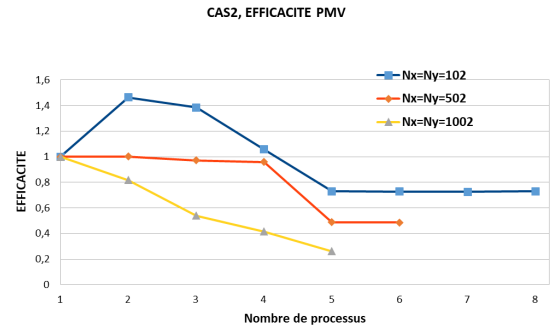


FIGURE 15 – EFFICACITE SUR LE PMV cas 2

Pour  $N=1002$  et pour  $N_p > 4$  l'on ne constate pas de bonne valeur de speed up ni d'efficacité, on reporte en Annexe F la table de valeur pour le cas 2 où l'on voit aussi que la communication PtoP 2WAYS ne diminue les temps de communication que 0,1 à 5 secondes par rapport à la PtoP. Enfin, Sur la figure 16, on reporte la somme des temps (communication et allreduce) en pourcentage du temps total. On constate que les ces opérations sont inférieures à 10% du temps total pour  $N_p \leq 4$  ce qui conforte leur implémentation. Ainsi, l'utilisation des treads n'optimise pas les communications et n'est que peu efficace pour l'opération PMV (cf figure 15). On met en Annexe G la table de valeur pour le cas 3.

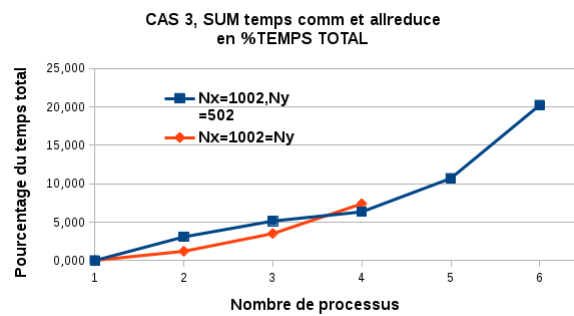


FIGURE 16 – Opération de communication en Pourcentage du temps total , cas 3

## 6 Conclusion

Nous avons vu que l'utilisation des treads n'est que peu efficace, et qu'augmenter le nombre de processus conduit à une augmentation des temps de communication (PtoP et allreduce). Nous n'avons pas obtenu des courbes de speed up proche d'une courbe idéale pour les temps totaux de même pour l'efficacité. Nous expliquons cela par le coût des communication et du PMV dans certains cas. Dans d'autrse cas, nous avons constaté de bon speed up pour le PMV et des faibles temps de communication. On constate aussi dans les tables de valeurs en annexe que l'écriture de la solution en parallèle ne représente un gain de temps très petit.

## 7 Annexe

ANNEXE A : Table de valeur en mode 2 cas 1

MODE 2				SOMME SUR ITMAX				%
Nx=Ny=502	DT=0,5 sec	ITMAX=15		DES TEMPS MAX en seconde				TEMPS
Np	TEMPS (s)	SPEED UP	EFFICACITE	COMM VECT	PMV	ALLREDUCE	ECRITURE	TOTALE
1	71,88	1	1	0,00151	10,722	0,0223	34,401	14,950
2	53,012	1,356	0,678	4,401	5,394	0,233	29,54	18,916
3	47,321	1,519	0,506	6,43	3,693	0,729	27,369	22,933
4	44,827	1,603	0,401	12,891	2,946	0,672	28,256	36,828
MODE 2				SOMME SUR ITMAX				%
Nx=Ny=2002	DT=5,5 sec	ITMAX=1		DES TEMPS MAX en seconde				TEMPS
Np	TEMPS (min)	SPEED UP	EFFICACITE	COMM VECT	PMV	ALLREDUCE	ECRITURE	TOTALE
1	8,766	1	1	0,0022	87,969	0,0513	67,01	16,74
2	6,753	1,298	0,649	0,331	51,403	0,423		12,87
3	6,661	1,316	0,439	5,674	55,655	6,813	56,188	17,05
4	6,569	1,334	0,334	3,735	54,306	5,999	55,283	16,25

FIGURE 17 – TABLE de VALEUR POUR LE MODE 2 CAS 1

ANNEXE B à E : Table de valeur pour les figures 7 à 10

Nx=Ny=32		DT= 0,5 sec	ITMAX=15	
Mode 1				
Np	TEMPS (s)	SPEED UP	EFFICACITE	
1	0,26	1	1	
2	0,203	1,281	0,640	
3	0,190	1,368	0,456	
4	0,198	1,313	0,328	
Nx=Ny=102		DT=0,5 sec	ITMAX=15	
Mode 1				
Np	TEMPS (s)	SPEED UP	EFFICACITE	
1	1,975	1	1	
2	1,724	1,146	0,573	
3	1,688	1,170	0,390	
4	1,632	1,210	0,303	
Nx=Ny=502		DT=0,5 sec	ITMAX=15	
Mode 1				
Np	TEMPS (s)	SPEED UP	EFFICACITE	
1	72,744	1	1	
2	53,94	1,349	0,674	
3	47,904	1,519	0,506	
4	44,871	1,621	0,405	
Nx=Ny=1002		DT=2,5 sec	ITMAX=2	
Mode 1				
Np	TEMPS (s)	SPEED UP	EFFICACITE	
1	124,642	1	1	
2	103,204	1,208	0,604	
3	101,07	1,233	0,411	
4	94,9	1,313	0,328	

FIGURE 18 – Annexe B



Nx=Ny=32	DT=0,5 sec	ITMAX=15	
Mode 2			
Np	TEMPS (s)	SPEED UP	EFFICACITE
1	0,23	1	1
2	0,201	1,144	0,572
3	0,198	1,162	0,387
4	0,194	1,186	0,296
5	0,21	1,095	0,219
6	0,23	1	0,167
7	0,274	0,839	0,120
8	0,31	0,742	0,093
Nx=Ny=102	DT=0,5 sec	ITMAX=15	
Mode 2			
Np	TEMPS (s)	SPEED UP	EFFICACITE
1	2,015	1	1
2	1,746	1,154	0,577
3	1,681	1,199	0,400
4	1,65	1,221	0,305
5	1,605	1,255	0,251
6	1,586	1,270	0,212
7	1,612	1,250	0,179
8	1,668	1,208	0,151

FIGURE 19 – Annexe C

Nx=Ny=502	DT=0,5 sec	ITMAX=15	
Mode 2			
Np	TEMPS (s)	SPEED UP	EFFICACITE
1	72,164	1	1
2	57,186	1,262	0,631
3	47,922	1,506	0,502
4	44,646	1,616	0,404
5	47,759	1,511	0,302
6	45,881	1,573	0,262
7	45,758	1,577	0,225
8	43,439	1,661	0,208
Nx=Ny=1002	DT=2,5 sec	ITMAX=2	
Mode 2			
Np	TEMPS (s)	SPEED UP	EFFICACITE
1	124,507	1	1
2	103,061	1,208	0,604
3	99,8	1,248	0,416
4	95,487	1,304	0,326
5	98,747	1,261	0,252
6	98,791	1,260	0,210

FIGURE 20 – Annexe D

7	98,794	1,260	0,180
8	99,626	1,250	0,156
Nx=Ny=2002	DT=0,5 sec	ITMAX=1	
Mode 2			
Np	TEMPS (min)	SPEED UP	EFFICACITE
1	8,148	1	1
2	6,477	1,258	0,629
3	6,361	1,281	0,427
4	6,303	1,293	0,323
5	6,413	1,270	0,254
6	6,414	1,270	0,212
7	6,475	1,258	0,180
8	6,313	1,291	0,161

FIGURE 21 – Annexe E

ANNEXE F : Table de valeur pour le cas 2

CAS2				
Nx=Ny=102	DT=0,1 sec	ITMAX=60		Comm en
Mode 2				mode pivot
Np	TEMPS (s)	SPEED UP	EFFICACITE	Comm
1	8,315	1	1	0,001491
2	6,095	1,364	0,682	1,727
3	5,68	1,464	0,488	2,556
4	5,581	1,490	0,372	2,906
5	5,608	1,483	0,297	2,929
6	5,585	1,489	0,248	3,128
7	5,56	1,496	0,214	3,602
8	5,607	1,483	0,185	3,484
Nx=Ny=502	DT=0,1 sec	ITMAX=60		Comm en
Mode 2				mode pivot
Np	TEMPS (s)	SPEED UP	EFFICACITE	Comm
1	358,088	1	1	0,0126
2	234,218	1,529	0,764	25,385
3	199,482	1,795	0,598	12,882
4	186,18	1,923	0,481	57,268
5	198,483	1,804	0,361	51,562
6	195,086	1,836	0,306	80,081
Nx=Ny=1002	DT=2,5 sec	ITMAX=2		
Mode 2				
Np	TEMPS (s)	SPEED UP	EFFICACITE	Comm
1	154,882	1	1	0,00266
2	123,214	1,257	0,629	1,052
3	116	1,335	0,445	1,263
4	119,4	1,297	0,324	2,244
5	119,128	1,300	0,260	8,068

FIGURE 22 – Annexe F cas 2

Nx=Ny=102				comm		
		% TEMPS		de base	ERR	ERR
pmv	allreduce	TOTAL	ecriture	comm	N2	Ninfinity
0,939	0,0387	11,776	5,917		4,52E-05	8,09E-07
0,321	0,04188	34,288	4,233			
0,226	0,31	54,437	3,726			
0,222	1,502	82,960	3,363			
0,257	2,115	94,526	3,553			
0,215	1,822	92,480	3,199	3,358		
0,185	1,495	95,000	3,855			
0,161	2,173	103,763	3,103	3,803		
Nx=Ny=502				comm		
		% TEMPS		de base	ERR	ERR
pmv	allreduce	TOTAL	ecriture	comm	N2	Ninfinity
80,198	0,249	22,469	126,223		9,12E-06	3,28E-08
40,029	2,84	29,141	104,297			
27,534	4,888	22,711	104,868			
20,934	5,948	45,198	108,89	58,378		
32,824	32,171	58,724	99,564			
27,483	28,57	69,782	98,505	85,01		
Nx=Ny=1002						
		% TEMPS				
pmv	allreduce	TOTAL	ecriture			
26,375	0,0613	17,070	24,913			
16,14	1,059	14,812	22,016			
16,317	1,87	16,767	20,21			
15,893	1,92	16,798	19,758			
20,112	8,781	31,026	20,909			

FIGURE 23 – Annexe F cas 2 suite

ANNEXE G : Table de valeur pour le cas 3

CAS3				
Nx=1002,Ny=502	DT=0,5 sec	ITMAX=2		Comm en
Mode 2				mode pivot
Np	TEMPS (s)	SPEED UP	EFFICACITE	Comm
1	48,966	1	1	0,00122
2	37,508	1,305	0,653	0,848
3	36,105	1,356	0,452	1,402
4	34,066	1,437	0,359	0,978
5	33,785	1,449	0,290	1,046
6	35,059	1,397	0,233	3,619
Nx=1002=Ny	DT=0,5 sec	ITMAX=2		
Mode 2				
Np	TEMPS (s)	SPEED UP	EFFICACITE	Comm
1	114,74	1	1	0,00142
2	89,935	1,276	0,638	0,476
3	86,661	1,324	0,441	1,368
4	87,018	1,319	0,330	4,186

FIGURE 24 – Annexe G table de valeur cas 3

Nx=1002,Ny=502			
		% TEMPS	
pmv	allreduce	TOTAL	ecriture
8,27	0,00299	0,009	12,518
4,671	0,307	3,079	11,5453
4,519	0,453	5,138	10,471
3,802	1,178	6,329	10,551
3,797	2,564	10,685	10,41
4,534	3,469	20,217	10,873
Nx=1002=Ny			
		% TEMPS	
pmv	allreduce	TOTAL	ecriture
19,3	0,0322	0,029	24,85
11,942	0,609	1,206	22,067
12,593	1,67	3,506	20,13
12,243	2,236	7,380	20,443

FIGURE 25 – Annexe G table de valeur cas 3, suite