Etude comparative de 3 architectures spécialisées avec un code applicatif

Remy Dubois* and Philippe Parnaudeau†

* IDRIS, CNRS, UPS 851
e-mail: remy.dubois@idris.fr, web page: http://www.idris.fr

† Institut Pprime, CNRS, UPR 3346
e-mail: philippe.parnaudeau@cnrs.pprime.fr - Web page: https://www.pprime.fr

Jean Zay (IDRIS)

Caractéristiques principales :

- La puissance crête cumulée est de **36.76 PFs**$^{-1}$
- **2 712** Nvidia Tesla V100 et **440** Nvidia Tesla A100
- **89 976** cœurs (Intel Cascade Lake 6248 et AMD Milan 7543)
- **468 TB** mémoire
- Un réseau Intel Omni-PAth 100 Gbs^{-1}
- Peformance / Watt (théorique) : $\sim$ **16.76 GFs**$^{-1}$**/W**

Boreale (CRIANN)

Caractéristiques principales :

- La puissance crête cumulée est de **176.9 TFs**$^{-1}$
- **72** NEC SX-Aurora VE 20B soit **576** cœurs vectoriels
- **288** cœurs (Intel Cascade Lake 6248)
- **4.4 TB** mémoire
- Un réseau infiniband 200 Gbs^{-1}
- Performance / Watt (théorique): $\sim$ **5.9 GFs**$^{-1}$**/W**

Turpan (CALMIP)

Caractéristiques principales :

- La puissance crête cumulée est de **613.5 TFs**$^{-1}$
- **30** Nvidia Tesla A100
- **1200** cœurs (ARM V8.2)
- **7.68 TB** mémoire
- Un réseau infiniband 200 Gbs^{-1}
- Performance / Watt (théorique): $\sim$ **50 GFs**$^{-1}$**/W**

Les accélérateurs étudiés

Calculateur	Jean Zay	Boreale	Turpan
Accélérateur étudié	V100	VE-20B	Ampere Altra Q80-30
Puissance de calcul (DP)	**7.8 TF/s**	2.45 TF/s	1.9 TF/s
Mémoire, capacité	16 GB (HBM2)	48 GB (HBM2)	512 GB (DDR4)
Mémoire, débit (mesure prod.)	900 GB/s	**1530 GB/s**	175 GB/s
Performance / Watt (théorique)	**26.0 GFs^{-1}/W**	8.16 GFs^{-1}/W	9.0 GFs^{-1}/W
Performance / Watt (HPL)	**18.0 GFs^{-1}/W**	7.28 GFs^{-1}/W	5.25 GFs^{-1}/W
Performance / Watt (HPCG)	**0.47 GFs^{-1}/W**	0.39 GFs^{-1}/W	0.11 GFs^{-1}/W

- Rappel : le projet Exascale vise une performance / Watt (avec HPL) de $[30; 50]$ GFs^{-1}/W
- Le débit mémoire est très favorable à la solution NEC
- **Avec HPL : V100 offre le meilleur ratio Performance / Watt. HPL est-il un bench représentatif ?**
- **Avec HPPG : V100 et VE ont un ratio similaire, le Q80 s'effondre**
- **Objectif :** quid de la capacité d'une application académique à tirer des performances de ces architectures ?

Multiphasique, compressible, cartésien et géométries complexes

Principales difficultés liées à la physique

- Ecoulement avec deux phases (exemples : collapse de bulle, cavitation), compressible avec présence d'onde de choc
- Intensité et brièveté des phénomènes, prise en compte des discontinuités, variations importantes des conditions thermodynamiques, changement de phase

Principales difficultés numériques et informatiques

- Choix du modèle (avec un nombre d'équations allant de 3 à 7 équations) et des schémas
- Coût élevé des simulations numériques 3D ($> 1.10^9$ cellules)

Solutions retenues :

$\Rightarrow$ Modèle à 4 équations : bonne précision et coût de calcul modéré [(1)]

$\Rightarrow$ Approche hybride MPI-ACC ou MPI-OMP pour usage sur supercalculateur hétérogène [(2)]

(1) : Comparison of multiphase models for computing shock-induced bubble collapse, Goncalves & Parnaudeau, Int. J. Num. Meth. Heat Fluid Flow, 22, pp. 3845-3877, 2020.
(2) : High performance computing of stiff bubble collapse on CPU-GPU heterogeneous platform, Dubois, Goncalves & Parnaudeau, Comput. & Mathematics Appl., 99, pp.246-256, 2021.

Stratégie de programmation

Remarques :

- Coût des simulations 3D $\Rightarrow$ utilisation des supercalculateurs
- Supercalculateur hétérogène (CPU + accélérateur) $\Rightarrow$ programmation hybride : 2 paradigmes (MPI+xxx)
- GPU majoritaire $\Rightarrow$ OpenCL, CUDA, OpenMP, OpenACC, etc. : lequel choisir ?

Choix des paradigmes :

- MPI+OpenMP - Pour un usage CPU-CPU (toute solution, mais pas encore version GPU)
- MPI+OpenACC - Pour usage CPU-CPU et CPU-GPU (sur solution adaptée)

Objectifs :

- Performance
- Portabilité
- Développement et maintenance

Roofline

L'analyse sur un coeur Intel Xeon 6248 :

- Intensité arithmétique $\simeq 0.3$ F/B
- $\simeq 7\%$ du pic théorique d'un seul coeur
- $\simeq 20\%$ de la bande passante mémoire maximale

Remarques :

- Test réalisé avec Intel-Advisor et la librairie PAPI
- HPCG a une intensité arithmétique de $[0.20 : 0.25]^{1,2}$ F/B

Accélérateur étudié	V100	VE-20B	Ampere Altra Q80-30
Intensité arithmétique (min)	8.5 F/B	**1.60** F/B	19.0 F/B

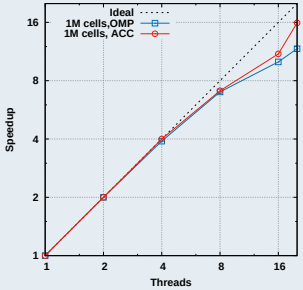
Hypothèse :

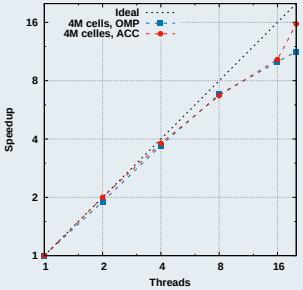
Boreale doit offrir les meilleurs performances avec SCB !

(1) Roofline Analysis with Cray Performance Analysis Tools (CrayPat) and Roofline- based Performance Projections for a Future Architecture, JaeHyuk Kwack, Galen Arnold, Celso Mendes, and Gregory H Bauer
(2) Supercomputer Benchmarks ! A comparison of HPL, HPCG, and HPGMG and their Utility for the TOP500, Erich Strohmaier, Jack Dongarra, Mike Heroux, Piotr Luszczek, Samuel Williams, and Mark Adams

Performances SCB : OpenMP vs OpenACC

Strong scaling : $[1 : 20]$ coeurs d'un processeur Intel Xeon 6248





- La programmation "grain fin" explique sans doute le manque de performance > 10 coeurs.
- OMP et ACC délivrent des performances similaires sur X86, le biais favorable à OMP semble limité (dans notre cas).
- Les temps de calcul sont identiques également avec l'ensemble des coeurs, un gap existe avec peu de coeurs.

Performances SCB : test sur un V100, VE et Q80-30

Strong scaling, problème 2D de 200K cellules, empreintu mémoire : 500 MB

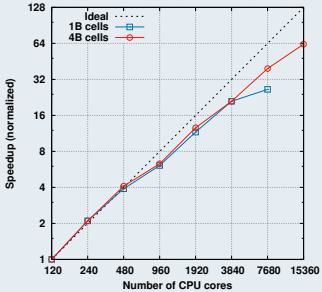
	Temps de calcul (s)				Energie (Kj)			
	Intel Xeon 6248 **Nvidia v23.1 (1) OMP**	**V100 Nvidia v23.1 (2) ACC**	**VE-20B Nec v3.5 (3) OMP**	**Q80-30 ARMFlang v22.1 (4) OMP**	**Intel Xeon** 6248	**V100**	**VE-20B**	**Q80-30**
% Threads/max	**20**	**1**	**8**	**80**	**20**	**1**	**8**	**80**
1.25 %				1965.03				584.86
5 %	1633.73				334.0			
6.25 %				505.27				164.68
12.5 %			118.19	320.84			32.84	110.85
25 %	437.05		69.83	234.74	98.98		20.30	85.91
50 %	312.75		47.84	206.75	87.71		14.47	80.00
100 %	247.96	82.81	38.86	205.72	74.84	45.90	12.13	91.19

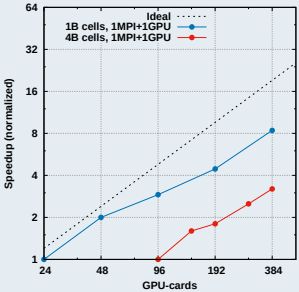
% Threads/max : le pourcentage du nombre de coeurs / au max

(1) : -Mpreprocess -fast -Mnoidiom -m64 -Mvect=nocond -acc -ta=multicore
(2) : -Mpreprocess -fast -Mnoidiom -m64 -Mvect=nocond -acc=gpu
(3) : -O4 -finline-functions
(4) : -O3 -fsimdmath -ffp-contract=fast -fvectorize -funroll-loops -ffast-math -finline-functions -mcpu=native

Performances SCB : MPI-seul et MPI-ACC sur plusieurs accélérateurs

- Strong scaling : $[120 : 15360]$ Intel Xeon Gold 6248 versus $[24 : 384]$ NVidia Tesla V100





Performances SCB : test sur un noeud avec plusieurs accélérateurs V100, VE et Q80-30

Strong scaling, problème 3D de 256 millions de cellules, empreinte mémoire : 62 GB

	Temps de calcul (s)				Energie (Kj)			
	Intel Xeon 6248 **Nvidia v23.1 (1) MPI(1)**	**V100 Nvidia v23.1 (2) MPI+ACC (2)**	**VE-20B Nec v3.5 (3) MPI (3)**	**Q80-30 ARMFlang v22.1 (4) MPI (4)**	**Intel Xeon** 6248	**V100**	**VE-20B**	**Q80-30**
% ressources/max	**80(80)**	**4(4)**	**8(64)**	**80(80)**	**80**	**4**	**8**	**80**
2.50 %	48310.86			12947	9820			4300
12.5 %	10479.30		4935.51	7286	2930		1360	2580
25.0 %	6092.39		2618.64	4638	1900		722	1800
50.0 %	4228.39		1184.58	3688	1610		334	1570
100.0 %	2175.19	306.58	612.75	3467	1160	?	185	1730

% des ressources disponibles pour un noeud de calcul

(1) : -Mpreprocess -fast -Mnoidiom -m64 -Mvect=nocond -acc -ta=multicore. Openmpi v4.1.1
(2) : -Mpreprocess -fast -Mnoidiom -m64 -Mvect=nocond -acc=gpu. Openmpi v4.1.1
(3) : -O4 -finline-functions. MPI-Nec V3.0.0
(4) : -O3 -fsimdmath -ffp-contract=fast -fvectorize -funroll-loops -ffast-math -finline-functions -mcpu=native. Openmpi v4.1.4

Conclusion

Remarques

- L'adaptation du code à GPU-Nvidia $\Longleftrightarrow$ 3 mois ETP (SCB $\approx$ 300K F90)
- La parallélisation (MPI+OMP) valable sur (X86 et Vec) $\Longleftrightarrow$ 2 mois ETP
- L'optimisation spécifique vectorielle $\Longleftrightarrow$ 1 semaine ETP

Résultats

- NEC offre une solution très performante pour un code Stencil comme SCB
- Il est difficile de tirer des performance de l'ARM
- Le GPU est une solution, mais l'enveloppe énergétique qui l'accompagne est élevée

Perspectives

- Le GPU, mais quel paradigme ?
- MPI-OpenMP en cours d'investigation... mais la maturité des compilateurs pour OMP-Target est un frein
- Souhait : une version commune entre OpenACC and OpenMP avant ma retraite

Remerciements

Merci aux collègues des équipes supports des 3 centres de calcul
A titre personnel, je tiens à remercier Laurent Gatineau